19TH INTERNATIONAL CONFERENCE

ON

INTELLIGENT GAMES AND SIMULATION

GAME-ON® 2018

EDITED BY

David King

September 18-20, 2018

Abertay University

Dundee, Scotland

UNITED KINGDOM

A Publication of EUROSIS-ETI

Cover Art by Sarah Martin, Abertay University

19TH International Conference

on

Intelligent Games and Simulation

DUNDEE, SCOTLAND

SEPTEMBER 18 - 20, 2018

Organised by

ETI

Sponsored by

EUROSIS

Co-Sponsored by

Ghent University

University of Skövde

University of Žilina

GODAN

Abertay University

Dundee, Scotland

EXECUTIVE EDITOR

PHILIPPE GERIL
(BELGIUM)

EDITOR

General Conference Chair

David King
SFHEA
Division of Computing & Mathematics, School of Design and Informatics
University of Abertay, Dundee, United Kingdom

Local Programme Committee

Christopher Acornley
SFHEA
Division of Computing & Mathematics, School of Design and Informatics
University of Abertay, Dundee, United Kingdom
Grant Clarke
SFHEA
Division of Computing & Mathematics, School of Design and Informatics
University of Abertay, Dundee, United Kingdom
Karen Meyer
SFHEA
Division of Computing & Mathematics, School of Design and Informatics
University of Abertay, Dundee, United Kingdom

Past Conference Chair

Joseph Kehoe, Institute of Technology Carlow, Carlow, Ireland

INTERNATIONAL PROGRAMME COMMITTEE

Game Development Methodology and Gamification

Óscar Mealha, University of Aveiro, Portugal

Physics and Simulation

Graphics Simulation and Techniques
Ian Marshall, Coventry University, Coventry, United Kingdom
Marco Roccetti, University of Bologna, Bologna, Italy

Facial, Avatar, NPC, 3D in Game Animation
Yoshihiro Okada, Kyushu University, Kasuga, Fukuoka, Japan
Marcos Rodrigues, Sheffield Hallam University, Sheffield, United Kingdom

Artificial Intelligence

Artificial Intelligence and Simulation Tools for Game Design
Stephane Assadourian, UBISOFT, Montreal, Canada
Antonio J. Fernandez, Universidad de Malaga, Malaga, Spain
Marc-Philippe Huget, University of Savoie, Le-Bourget-du-Lac, France
Joseph Kehoe, Institute of Technology Carlow, Carlow, Ireland
David Moffat, Glasgow Caledonian University, Glasgow, United Kingdom
Miguel Tsai, Ling Tung University, Taichung, Taiwan

INTERNATIONAL PROGRAMME COMMITTEE

Learning & Adaptation
Christian Bauckage, University of Bonn, Sankt Augustin, Germany
Christos Bouras, University of Patras, Patras, Greece
Adriano Joaquim de Oliveira Cruz, Univ. Federal de Rio de Janeiro, Rio de Janeiro, Brazil
Andrzej Dzielinski, Warsaw University of Technology, Warsaw, Poland

Intelligent/Knowledgeable Agents
Wenji Mao, Chinese Academy of Sciences, Beijing, China P.R.

Collaboration & Multi-agent Systems
Sophie Chabridon, Groupe des Ecoles de Telecommunications, Paris, France

Opponent Modelling
Ingo Steinhauser, Binary Illusions, Braunschweig, Germany

Peripheral

Psychology, Affective Computing and Emotional Gaming

Eva Hudlicka, Psychometrix Associates, Blacksburg, USA

Artistic input to game and character design

Anton Eliens, Vrije Universiteit Amsterdam, Amsterdam, The Netherlands
Richard Wages, Nomads Lab, Koln, Germany

Storytelling and Natural Language Processing

Jenny Brusk, Gotland University College, Gotland, Sweden
Ruck Thawonmas, Ritsumeikan University, Kusatsu, Shiga, Japan
Clark Verbrugge, McGill University, Montreal, Canada

Online Gaming and Security Issues in Online Gaming
Marco Furini, University of Modena and Reggio Emiliano, Modena, Italy
Michel Simatic, Telecom Sud-Paris, Evry Cedex, France

MMOG's

Michael J. Katchabaw, The University of Western Ontario, London, Canada
Jens Mueller-Iden, University of Munster, Munster, Germany

Serious Gaming

Wargaming Aerospace Simulations, Board Games etc....
Roberto Beauclair, Institute for Pure and Applied Maths., Rio de Janeiro, Brazil
Joseph M. Saur, Georgia Tech Research Institute, Atlanta, USA
Jaap van den Herik, Tilburg University, Tilburg, The Netherlands

Games for Training
Michael J. Katchabaw, The University of Western Ontario, London, Canada

Games Applications in Education, Government, Health, Corporate, First Responders and Science
Laura Lima, Universidade Federal de Juiz de Fora, Minas Gerais, Brazil

All articles published in these Proceedings have been peer reviewed

EUROSIS-ETI Publications are ISI-Thomson and IET referenced

GAMEON proceedings are indexed on SCOPUS and Elsevier Engineering Village

A CIP Catalogue record for this book is available from the Royal Library of Belgium under nr.12620

For permission to publish a complete paper write EUROSIS, c/o Philippe Geril, ETI Executive Director, Greenbridge Science Park, Ghent University, Ostend Campus, Wetenschapspark 1, Plassendale 1, B-8400 Ostend, Belgium.

EUROSIS is a Division of ETI Bvba, The European Technology Institute, Torhoutsesteenweg 162, Box 4.02, B-8400 Ostend, Belgium

Printed in Belgium by Reproduct NV, Ghent, Belgium

EUROSIS-ETI Publication

ISBN: 978-9492859-04-4
EAN: 978-9492859-04-4

GAME ON®
2018

Preface

Dear Participants,

We would like to extend a warm welcome to everyone attending the 19th GAME-ON conference in Dundee (Officially, the sunniest city in Scotland!). This year's event is being hosted by the School of Design and Informatics at Abertay University, Dundee, Scotland from the 18th of September until the 20th of September 2018.

We are delighted to be chosen to host this year's event, particularly as we have just last year celebrated 20 years involvement in teaching games technology, and look forward to hearing about the latest approaches to all things related to simulation and AI in Computer Games.

We are particularly pleased with the keynote speakers who have agreed to talk at this conference. Michael Cook is a Senior Research Fellow at the University of Falmouth's Metamakers Institute and is an expert in procedural generation of game content and is probably best known as the creator of ANGELINA, an AI that designs games. Alan Hinchcliffe is the Lead Engineer with Spirit AI and has vast experience in the technical challenges of video games. All three speakers bring fresh perspectives alongside a wealth of experience and their talks will give attendees new insights into the future of Gaming. We are also happy to introduce you to our own Joseph DeLappe who is Professor of Games and Tactical Media in the Division of Games and Art at Abertay and has been invited to give a talk on his involvement in online gaming performance and conceptual art and how computer games can be used as a means of creating art intervention pieces exploring contemporary issues in politics.

This conference would not be possible without the input and effort of many people. This includes the participants who have submitted and will present papers over the course of the conference and the programme committee who have reviewed papers and helped organize the event. We would particularly like to thank, Karen Meyer, Grant Clarke and Christopher Acornley in the School of Design and Informatics, and Vivien Collie and Elizabeth Wilson in Events for all their work helping to organise this conference, and especially Sarah Martin, who is studying for an MProf in Games Development at Abertay, for designing the cover for the conference proceedings. But most of all we must thank Philippe Geril, the behind the scenes driving force, without whom this conference would not be possible.

I hope that you enjoy your stay in Scotland and find the conference interesting and inspiring.

Dundee, September 2018

Dave King
Abertay University, Dundee, Scotland
GAME-ON'2018 General Conference Chair

CONTENTS

CONTENTS

SCIENTIFIC PROGRAMME

GAME
AI

Monte Carlo Tree Search for Quoridor

Victor Massagué Respall, Joseph Alexander Brown and Hamna Aslam
Artificial Intelligence in Games Development Lab, Innopolis University
Innopolis, Republic of Tatarstan, Russia, 420500
Email: v.respall@innopolis.ru, j.brown@innopolis.ru and h.aslam@innopolis.ru

KEYWORDS
Monte Carlo Tree Search, Quoridor, Genetic Algorithm, Agent

ABSTRACT

This paper presents a preliminary study using Monte Carlo Tree Search (MCTS) upon the board game of Quoridor. Quoridor is an interesting game for expansion of player agents in MCTS due to having a mechanically simple rule set, however, Quoridor has a state-space complexity similar to Chess with a higher game-tree complexity. The system is shown to perform well against current existing methods, defeating a set of player agents drawn from an existing digital implementation as well as a previous method using a GA.

INTRODUCTION

Monte Carlo Tree Search (MCTS) (Coulom 2006) (Kocsis and Szepesvári 2006) is a technique well-known these days (Browne, C. B. et al. 2012) due to the efficient results obtained in the board game *Go* (Silver D. et al. 2016). This game, produces difficulty for an AI expert to create an agent, due to its space-complexity, branching factor and difficulties to evaluate the state of the game in the middle. To deal with these, Monte Carlo tree search was used because of its following properties. It uses *UCT* (Upper Confidence Bound applied to Trees) (Gelly et al. 2006) for evaluating the final states of the game. Also, it consists of Monte Carlo rollouts, explained later in Section V, to estimate the value of each state in the search tree. As the tree grows larger more accurate values are generated. The average of these rollouts can provide an effective position evaluation achieving accurate performance in games such as Backgammon (Tesauro and Galperin 1997) and Scrabble (Sheppard 2002). This paper presents the research on the board game Quoridor to develop an artificial player agent using the Monte Carlo tree search algorithm and compares it with current existing agents.

QUORIDOR

Quoridor (Marchesi 1997) is played in a nine by nine board. We focus only on the two-player version. Each player is represented by a pawn which begins at the center space in opposite edges of the board, the baselines. The goal is to be the first player to move their pawn from its side to the opposite side of the board, the opposite baseline.
The main feature that makes this game interesting and tactical is its fences. Fences are flat two-space-wide pieces which can be placed in the groove between the squares of the board. Fences have the ability to facilitate the player's progress or block the path of the pawns, which must go around them. Each player has ten fences at the start of the game, and once placed, cannot be moved or removed.
Each player at his turn can choose to move his pawn or to place one of his fences. Once the player runs out of fences, its pawn must be moved. Pawns are moved one square at a time, horizontally or vertically, forwards or backwards. When two pawns face each other on neighboring squares which are not separated by a fence, the player whose turn is it can jump over the opponent's pawn and place himself behind the opponent's pawn, thus advancing an extra square. If there is a fence behind the pawn, the player can place his pawn to the left or the right of the opponent's pawn. Fences may not be jumped, including when moving laterally due to a fence being behind a jumped pawn, see Figure 1.

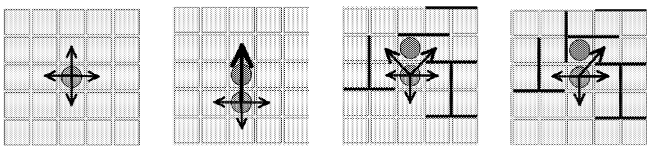


Figure 1: Allowed movements for the lower red pawn

Fences can be placed directly between two spaces, in any groove not already occupied by a fence. However, a fence may not be placed which cuts off the only remaining path of any pawn to the goal. The first player who reaches one of the 9 squares of his opponent's base line is the winner.
There is no official notation for Quoridor, so for this project, we use a notation from a community of players (Quoridor Strats 2014). The notation proposed is similar to algebraic Chess notation. Each square gets a unique letter-number designation. Each column is given a letter *A-I* and each row is given a number *1-9*. A move

is recorded as the column followed by the row. The first player always starts on *E1* and the second player always starts on *E9*. This marks the top and the bottom of the board, which are needed for recording fence placement. Each fence move is defined by the lower left square they touch along with their direction, horizontal or vertical. Every fence touches four squares, so we denote the position of the fence by the square closest to the *A1* corner of the board.

PREVIOUS AGENTS FOR QUORIDOR

Mastering Quoridor (Glendenning et al. 2005)
Agent development has been done by implementing a search algorithm, using iterative-deepening alpha-beta negamax search algorithm with few modifications. For the evaluation function, the ten features chosen are summarized as: the Shortest path to the goal, Markov Chains, Goal Side, and Manhattan Distance for both players, and Pawn Distance and Number of fences of the current player. For the learning algorithm, a variant of a GA is using as a fitness evaluation function, the number of games that a chromosome wins against the others inside the population. This agent was shown to be easily beaten by a human.

A Quoridor-playing Agent (Mertens 2006)
MiniMax algorithm is used in this case with Alpha-Beta pruning, but the game tree is to large to perform MiniMax search all the way down to the leaves of the tree. Therefore, the solution is limiting the depth of the MiniMax search. Further, an evaluation function is applied to determine the value of a position in order to allow for a quicker return. The result obtained is a weak Quoridor agent as it is unable to see depth in the game. MCTS balances the want for a fast evaluation mixed with the ability to see beyond a set horizon depth.

GAME COMPLEXITY

Quoridor has the number of possible ways to determine a fast upper bound on complexity, such as pawns moves multiplied by the number of possible ways to place the fences. Since the board has eighty one squares, we can place the first pawn in any of them, and the second one in eighty, due to the first pawn already placed on the board. Hence, the total number of positions, S_p, with two pawns is given by the following equation:

$$S_p = 81 * 80 = 6480 \tag{1}$$

Further, for the fences, since each fence occupies 2 squares, there are eight ways to place a fence in one row. Given that there are eight rows, there are sixty four possible places to put a fence horizontally. Since the board is a square, we have the same number of rows and columns, one fence can be put in one hundred and twenty eight places. We have to take into account that one fence occupies four fence positions, except for the squares on the border. So the total number of positions of the twenty fences, S_f, and the upper bound of the size of the state space, *S*, are given by the following equations(Mertens 2006):

$$S_f = \sum_{i=0}^{20} \prod_{j=0}^{i} (128 - 4i) = 6.1582 * 10^{38} \tag{2}$$

$$S = S_p * S_f = 6480 * 6.1582 * 10^{38} = 3.9905 * 10^{42} \tag{3}$$

Quoridor has a state-space complexity similar to Chess and a higher game-tree complexity, shown to be 10^{162} in (Mertens 2006).

MONTE CARLO TREE SEARCH

The Monte Carlo Tree Search algorithm is used for building the AI agent for Quoridor, as it appears to be an efficient algorithm for this type of board game and game tree size. It is a probabilistic search algorithm with a unique decision-making ability because of its efficiency in open-ended environments with an enormous amount of possibilities. To deal with the size of the game tree, it applies Monte Carlo method (Metropolis and Ulam 1949). As it is based on random sampling of game states, it does not need to use brute force.
We have built a game tree with a root node, then it is expanded with random simulations. In the process, we maintain the number of times we have visited a specific node and a win score, used to evaluate the state of the board. In the end, we select the node with best results and higher win scores.
This algorithm consists of four phases:

1. *Selection*: In this initial phase, the algorithm starts with a root node and selects a child node such that it picks the node with maximum win rate. In order to make sure that each node is given a fair chance to be selected and to balance the situation between exploration and exploitation, we use *UCT* (Gelly et al. 2006).

$$\frac{w_i}{n_i} + c * \sqrt{\frac{ln(t)}{n_i}} \tag{4}$$

   Where

   - w_i = number of wins after the i-th move
   - n_i = number of simulations after the i-th move
   - c = exploration parameter (theoretically equal to $\sqrt{2}$ (Kocsis and Szepesvári 2006))
   - t = total number of simulations for the parent node

This formula ensures that agent will play promising branches more often than their counterparts, but will also sometimes explore new options to find a better node, if exists.

It selects the best node of the entire generated tree, traverses down the tree and selects a leaf node.

2. *Expansion*: When it can no longer apply UCT to find the successor node, it expands the game tree by generating all possible states from the leaf node.

3. *Simulation*: After Expansion, the algorithm picks a node randomly and it simulates the game until the very end, randomly for both players.

4. *Backpropagation*: This last phase consists of updating the nodes according to the result of the simulation. It evaluates the state to figure out which player has won and traverses upwards to the root incrementing visit counts and win scores, i.e. a count of if the player of that position has won, of each node visited.

   The algorithm keeps looping these four phases until some fixed number of iterations. Higher the number of iterations, more reliable the estimate becomes.

IMPLEMENTATION

This project is implemented in the Java programming language, using standard libraries. This section explains main classes and data structures used.
The class *Board* has all the information related to the board game, squares, fences and both players. All squares are stored in an *ArrayList*, and to navigate through the board, we use a *Map* that each key is a square and it returns a *Set* of all adjacent squares to the key.
For checking if a certain fence can be placed, we initially start the game with a *Set* containing all possible places that a fence can be placed on the board. Then, each time a fence is placed, we update the *Set* removing "forbidden" positions. After checking that the position of that fence is inside the *Set*, we must check that both players are not completely blocked from reaching the opposing baseline. For this, a *BFS* (Breadth First Search) is applied which finds the shortest path to the goal for each player and considers it an illegal move if the distance is undefined.
The system explained before for storing fences is also useful for returning all possible movements available to the player.
Another important class is *Monte Carlo Tree Search*, that contains the tree and all the algorithm explained in the section before. The first attempt to implement the tree was including in each node the board of the game. The problem was since it contains a huge amount of information and data structures, the program ran out of memory just with 5000 simulations, because MCTS was generating too many nodes. The solution is, instead of saving the board in each node generated, we store the move that we should perform and the scores of the node. Then each time when we visit a node we need to compute the move in a temporal board. The score added to each winning node is 10.
Finally, instead of using random decisions in the simulation phase of Monte Carlo Tree Search, we improved our system by adding a heuristic. The heuristic helps us to balance the placement of fences and the moves of the pawn. Running MCTS with random simulation shows that our agent spends all fences at the beginning of the game and on average is better to save them until the middle of the game. Basically this heuristic consists of calculating shortest path for each player, and if the player at the turn, has a shorter path than the opponent, then it gives more chances to move the pawn rather than placing another fence.

EXPERIMENTAL SETTINGS

Due to limited research on the game of Quoridor and no human Elo rankings, we cannot measure the level of the agent globally. However, we have tried to evaluate our player against self play with more simulation steps and against other player agent types to have an idea of the agent's ability:

120k Simulations Agent:
The default agent for running the experiments uses MCTS with the heuristic described before. It performs 120000 simulations of the game per decision.

60k Simulations Agent:
This agent was created to see the influence in the number of simulations of the game per decision. The only difference between this agent and the one stated in subsection *A* is that this agent is doing 60000 simulations per decision.

Alternative Agent:
To further evaluate the MCTS method, an alternative Quoridor agent base was found with four different agent levels *(Brain1, Brain2, Brain3 and Brain4)* (van Steenbergen 2006). *Brain1* simply moves the pawn at every turn without placing any fence. *Brain2*, places all fences at the beginning of the game, wasting all resources. *Brain3* places fences more strategically, but still the problem of placing all fences at the beginning of the game. That gives a lot of advantage to the opponent then. *Brain4* is the smartest agent, it focuses on reaching the goal and it uses fences during the middle of the game. It is sadly impossible to describe the algorithm that is using, because of the lack of information about the API from the developers.

Genetic Algorithm Agent:
This agent is based on a previous work, mentioned in Section III, Mastering Quoridor of Lisa Glendenning (Glendenning et al. 2005). For developing such an agent we used Minimax algorithm (Stockman 1979) with some modifications to improve its performance. It is possible to modify the game tree values to use just maximization operations, negating the returned values from the recursion. This approach is called *Negamax algorithm* (Campbell and Marsland 1983). However, the problem with Minimax search (Stockman 1979) is that the number of states it has to examine is exponential in the number of moves. For reducing this amount of moves it is applied *alpha-beta pruning* (Knuth and Moore 1975) (Pearl 1982) technique, that basically prunes away branches of the tree that cannot influence the final decision. Last modification applied was iterative-deepening due to the agent plays with time limit decision for every move. This allows us to return the best value computed until that point (Nilsson 1996).
After each execution of alpha beta, it is required an evaluation function for selecting the best state of the board. Eight features are proposed for Quoridor:

- Shortest Path Player (SPP), length of Breadth First Search path for the player
- Shortest Path Opponent (SPO),length of Breadth First Search path for the opponent
- Manhattan Distance Player (MDP), Manhattan distance for the player (straight distance from the player to the goal)
- Manhattan Distance Opponent (MDO), Manhattan distance for the opponent (straight distance from the opponent to the goal)
- Pawn Distance (PD), the distance between pawns using Breadth First Search
- Goal Side Player (GSP), boolean that tells if the player is between the midpoint of the board and the goal
- Goal Side Opponent (GSO), boolean that tells if the opponent is between the midpoint of the board and the goal
- Number Fences Player (NFP), number of fences of the player

Finally a genetic algorithm (GA) is used for weighting each feature described before. A chromosome is represented as a vector of weights and the fitness of each one is determined by the number of games that a chromosome wins against the other chromosome of the population. To create a population it is initialized with random float point values. It is established with a probability of 0.3 that a chromosome has a non-zero weight in some feature.

RESULTS

For selecting the Genetic Algorithm between all chromosomes created inside a population, we performed a tournament consisting about creating a population of 10 chromosomes and each one plays against each other. This will give us the fitness of the chromosomes. The maximum number of moves to avoid infinite loops was set to 120, and the decision time to 10 seconds per move. For playing against our 120k agent we took the two best chromosomes of the tournament.
In all experiments performed, our player was the default, 120k Agent. The experiment with 60k Agent was done automatically for 250 games since it was easy to adapt the code to play against itself. However, as we do not have the source code for the alternative agents, the experiments with all of them were done manually, taking the moves from our agent and playing them against the four brains. In the evaluative process, first ten games were played against each of the four brain types, when there was no an obvious mercy situation, i.e. in the case of *Brain4*, then we extended this to 100 games in order to allow for a statistical evaluation.
Table 1 shows that reducing the amount of simulations to 60000, our agent performs a little bit better than 120k agent, and much faster to decide each movement. Though this result is not significant at $p < 0.05$. Note that in playing these games, in 40 instances the MCTS players began to alternate moving a pawn back and forth, not seeing an obvious solution to winning the game. We therefore can infer there is a situation in the game which such a delaying tactic has some amount of value, or for which there is not an obvious good strategy. This only happened in self play, and more analysis is required to understand what developed this situation, as there is no obvious tie state.
The first three brains of the alternative agent and Chromosome 2 and 5 of the GA agent were easy to beat, with significant results over the ten evaluations each the MCTS would win 100% of the time. However, with *Brain4*, it is not possible for our agent to win all the time, but still performed better that the opponent, significant at $p < 0.05$. There is no information about which method is used by this agent so we are unable to make a deep evaluation as to the play method which is able to at least give some challenge to MCTS. Finally, using (Glendenning et al. 2005) best chromosome achieved - Psi1, the MCTS was able to defeat it in all of the ten games played.

CONCLUSIONS

We have created an MCTS agent for the board game Quoridor and compared it to a number of previous agent types, including reimplementation of a GA. This research work completed thus far focuses only in the two players version, Quoridor can be played with four play-

Table 1: Comparison of the 120k Agent to Other Agent Types (Significant at 95% Confidence in Bold using a Binomial exact test)

Opponent	Number of Games Played	Percentage (Count) of Wins for 120k	Lower Confidence Bound (95%)	Upper Confidence Bound (95%)
60k Agent	210*	46% (97 games)	39.3%	53.2%
Brain1	10	100%	69.2%	100%
Brain2	10	100%	69.2%	100%
Brain3	10	100%	69.2%	100%
Brain4	100	66% (66 games)	55.9%	75.2%
Chromosome2	10	100%	69.2%	100%
Chromosome5	10	100%	69.2%	100%
Psi1 (Glendenning et al. 2005)	10	100%	69.2%	100%

*250 games in total were played with 40 being undecided, we only examine the difference of wins in decided games.

ers, each with the goal of taking their pawn the opposite end of the board, the two other players take their pawns horizontally from the side baselines.

We have used Monte Carlo tree search as the main algorithm. It is a probabilistic search algorithm, and a unique decision making because of its efficiency in open-ended environments with an enormous amount of possibilities. Also, we have added a heuristic to balance the placement of fences and the moves of the pawn, in the simulation phase of the MCTS algorithm. The results obtained from the experiments are not sufficient to determine precisely the level of our agent, but it gives us an estimation of how it will perform against humans. The 60k agent as shown in Table 1 appears to perform better than the 120k agent and has a shorter runtime. Though this study will need to extend in order to prove this trend to hold in larger cases.

Future work should take into account the improvement in the heuristic, to decide a better quality movement and consider more features of the game such as a number of fences of each player. This will build a solid strategy for an agent. Moreover, as used in AlphaGo, deep learning can also be used (Silver D. et al. 2016). Finally, it is the goal of the authors to show that this system is competitive against the ranked human play. There are a number of human play strategies which are used commonly in competitive play, much along the same lines as chess openings, and perhaps it would be best to add an evaluated game tree as an initialization step to ensure competitive play.

REFERENCES

Browne, C. B. et al., 2012. *A survey of monte carlo tree search methods. IEEE Transactions on Computational Intelligence and AI in games*, 4, no. 1, 1–43.

Campbell M.S. and Marsland T.A., 1983. *A comparison of minimax tree search algorithms. Artificial Intelligence*, 20, no. 4, 347–367.

Coulom R., 2006. *Efficient selectivity and backup operators in Monte-Carlo tree search.* In *International conference on computers and games*. Springer, 72–83.

Gelly S.; Wang Y.; Teytaud O.; Patterns M.U.; and Tao P., 2006. *Modification of UCT with patterns in Monte-Carlo Go.*

Glendenning L. et al., 2005. *Mastering quoridor. Bachelor Thesis, Department of Computer Science, The University of New Mexico.*

Knuth D.E. and Moore R.W., 1975. *An analysis of alpha-beta pruning. Artificial intelligence*, 6, no. 4, 293–326.

Kocsis L. and Szepesvári C., 2006. *Bandit based monte-carlo planning.* In *European conference on machine learning*. Springer, 282–293.

Marchesi M., 1997. *Quoridor.* Family Games, Inc.

Mertens P.J., 2006. *A Quoridor-playing agent. Bachelor Thesis, Department of Knowledge Engineering, Maastricht University.*

Metropolis N. and Ulam S., 1949. *The monte carlo method. Journal of the American statistical association*, 44, no. 247, 335–341.

Nilsson N.J., 1996. *Artificial intelligence: A modern approach: Stuart Russell and Peter Norvig,(Prentice Hall, Englewood Cliffs, NJ, 1995); xxviii+ 932 pages.*

Pearl J., 1982. *The solution for the branching factor of the alpha-beta pruning algorithm and its optimality. Communications of the ACM*, 25, no. 8, 559–564.

Quoridor Strats, 2014. *Notation.* Https://quoridorstrats.wordpress.com/notation/.

Sheppard B., 2002. *World-championship-caliber Scrabble. Artificial Intelligence*, 134, no. 1-2, 241–275.

Silver D. et al., 2016. *Mastering the game of Go with deep neural networks and tree search. Nature*, 529, 484–503. URL `http://www.nature.com/nature/journal/v529/n7587/full/nature16961.html`.

Stockman G.C., 1979. *A minimax algorithm better than alpha-beta? Artificial Intelligence*, 12, no. 2, 179–196.

Tesauro G. and Galperin G.R., 1997. *On-line policy improvement using Monte-Carlo search.* In *Advances in Neural Information Processing Systems*. 1068–1074.

van Steenbergen M., 2006. *Quoridor.* online. Http://martijn.van.steenbergen.nl/projects/quoridor/.

Monte Carlo Tree Search for Love Letter

Tamirlan Omarov, Hamna Aslam, Joseph Alexander Brown, Elizabeth Reading
Artificial Intelligence in Games Development Lab, Innopolis University,
Innopolis, Republic of Tatarstan, Russia, 420500
email: t.omarov@innopolis.ru, h.aslam@innopolis.ru, j.brown@innopolis.ru, beth_reading@hotmail.com

KEYWORDS
monte carlo tree search, imperfect information games, determinization, minimax

ABSTRACT

Love Letter is a card game for two to four players. Players maintain one card in their hands at all times. For a series of rounds, a player draws a card and then plays one of the two cards in their hand with a goal of knocking out other players from the game by a card effect or by having the highest card at the end of the deck when all the cards have been played. We examined the use of Single Observer Information Set Monte Carlo Tree Search (SO-ISMCTS) in a player agent for a two-player Love Letter, compared against a knowledge based agent, Perfect Information Monte Carlo (PIMC) and Determinized Minimax and found that none of the algorithm could show the stable results due to game randomness.

Introduction

The research work presented in this paper, compares some of the existing MCTS methods (Browne et al. 2012) for imperfect information games(such as games with hidden information and uncertainty property) and traditional approaches such as knowledge based agent, Perfect Information Monte Carlo (PIMC) algorithm, Single Observer-Information Set Monte Carlo Tree Search (SO-ISMCTS) and Determinized Minimax. These algorithms have been applied to develop an AI player for the Love Letter game. Love Letter is an *imperfect information game* and players can not observe their opponents' cards and do not know the particular state of other players. The SO-ISMCTS algorithm is the main AI agent for which the rest of the three algorithms are playing as opponents. We have selected SO-ISMCTS as the main algorithm for comparison for its ability to search a single tree for games of imperfect information, it can be adopted for multi-player games, it has shown good results for more complex card games, it makes 4000 iterations during 2 seconds and can be used in games against human opponent. Also, SO-ISMCTS searches more deeply than PIMC with the same computational budget (Cowling et al. 2012), as Love Letter game has $\frac{16!}{5!2!2!2!2!} = 21794572800$ different initial states. These properties are essential to implement AI agent for the Love Letter game.

Love Letter

Love Letter (Kanai 2012) is a card game for two to four players. The game includes sixteen cards and thirteen red cube tokens. Each of the cards have a rank and text which explains its effect, see table 1. The game play starts by shuffling the cards and forming a face down draw deck. Players have a hand size of one and draw one card during their turn. The player must play one of his two cards and follow the effect of the card which is written over the card. Some of the cards have effects which remove a player from the round.

There are four win conditions in two player Love Letter. A player discards the Princess, so the other player wins. A player correctly guesses their opponent's hand using the Guard. A player has the higher card when the Baron is played, or when there is no deck remaining. In each round of the game, winner gets a token. In the event of a tie, the winner is the one who played the highest value of cards. The player with the most tokens of the thirteen rounds wins the game.

In the beginning of every round one card is removed from a deck that can be used if players run out of cards in the deck. We refer to it as an *out card* in this paper. In 2 player games, three more cards are removed from the deck and placed all face up.

In this paper we assume that playing the first move does not give any advantages. The maximum length of the game for 2 player version consists of 10 moves. So the depth of the game tree for all search tree based algorithms used in this work is 10 too.

Implementation

We have implemented four AI agents for the game Love Letter based on Single Observer Information Set Monte Carlo Tree Search (SO-ISMCTS) Algorithm, Perfect Information Monte Carlo (PIMC), Knowledge based agent and Determinized Minimax. The vertex of the game tree built, corresponds to a player's current hand and edges correspond to moves. Players are assumed to play rationally, i.e if a player can make a move and win the game, he has to make this move. For SO-ISMCTS and

Table 1: List of Cards - with some edits of rules text for clarity based off Kanai (2012)

Value	Name	Amount	Text
8	Princess	1	Player loses if this card is discarded or played
7	Countess	1	Player Must Discard this Card if the other card in hand is a King or Prince
6	King	1	Players trade hands
5	Prince	2	One player at the table *must* discard his hand
4	Handmaid	2	Player is protected from all card text until next turn
3	Baron	2	Player compares hands with another player; The player with the card of lower value loses
2	Priest	2	Player must look at another players hand
1	Guard	5	Player guesses another player's hand, if that guess is correct then that player loses

Player A cards: Guard, Countess

Player B card: Priest

Non played deck: Handmaid, Prince

Out card: Princess

Figure 1: Original Game State (assuming that players have already made some turns)

Player A cards: Guard, Countess

Player B card: Priest

Non played deck: Handmaid, Princess

Out card: Prince

Figure 2: Game State after Determinization

PIMC algorithms, the first step is the same. All of them are sampling the *determinization* randomly from *information set*. Where ***informations sets*** are collections of states, which appear in the game when some players have information about the state that others do not. For example, in Poker each player hides card from his opponents. Here information set consists of all possible permutations of states which corresponds to opponent's cards. A player knows his own information set, but he does not know particular state within this set. The ***determinization*** involves selecting the current state from player's information set and then all future events are fixed and known. In other words, it is the process of converting an imperfect information game to an instance of perfect information game(such as, a game in which all information about the state of the game is observable). For example, in case of a card game, determinization is the instance when all players' cards and deck are visible to all players.

In the next paragraphs algorithms for two player version will be described. The Player A has two cards in hand, one is a Guard and the other is the Countess. The player B has Priest card. The non-played deck consists of two cards, Handmaid and the Prince card. The out card is the Princess. This example will be used to describe steps of the algorithms. The state is shown in the figure 1.

Knowledge based agent

Knowledge based agent enforces game moves in accordance to the actual game rules. Secondly the algorithm checks for the game move that can lead to an immediate win in single turn. For instance, in our example (see Fig.2), player 'A' could win the round using the Guard's ability, provided player 'A' can correctly guess their opponent's card (see Table.1 for card's abilities). The knowledge based agent handles the selection of the guess card as: the card which is encountered the minimum number of times in the played deck is chosen. In Love Letter, there are predefined number of each type of card therefore the algorithm can deduce the number of non-played cards for each character in the game. The rule based approach prefers to make a move with the smallest valued card in the hand in order to save the highest one (each character card has a unique value). These set of rules and checks is called *domain knowledge*. The domain knowledge is reused in other three algorithms described further in the paper. For our example, the Guard has been selected by the knowledge based agent. The knowledge based agent algorithm does not need any iterations and it runs faster than the rest of the algorithms with the time complexity O(1). The full set of checks and conditions can be found in the code repository [1].

Perfect Information Monte Carlo (PIMC) algorithm

The first step in PIMC involve finding moves that can lead to an immediate win (as mentioned in the knowledge based algorithm). Then determinization (defined in Implementation section) is done and algorithm removes played deck from the full deck. After that, it deals card to the opponent randomly and shuffle the remaining cards. Thus, it has sampled one state from information set. One of the possible new states after determinization is shown in figure 2. Currently, the game tree consists of single vertex which is a root node only. Then Monte Carlo Tree Search (MCTS) steps are

[1] https://github.com/tamirOK/ISMCTS-for-love-letter-game

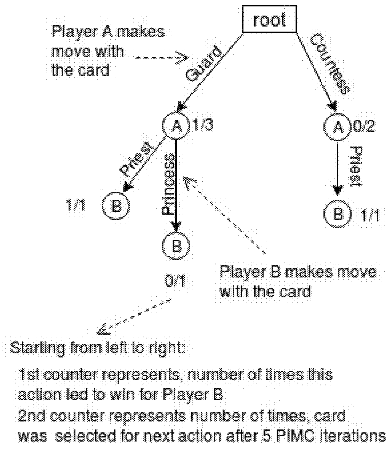


Figure 3: Game Tree after 5 MCTS iterations on Determinized game state

applied on this tree.

The first step is selection in which the algorithm chooses move according to UCB1 formula (Auer et al. 2002):

$$\bar{x}_i + c\sqrt{\frac{\ln n}{n_i}} \tag{1}$$

where $\bar{\boldsymbol{x}}_{\boldsymbol{i}}$ is average reward passing through the node $\boldsymbol{i}$. It is calculated as ratio of number of won simulations passed through the node to number of all simulations passed through the node. $\boldsymbol{n_i}$ is a number of times the node $\boldsymbol{i}$ was selected from its parent, $\boldsymbol{c}$ is a constant, that controls the rate of exploitation and exploration while traversing the tree nodes and $\boldsymbol{n}$ is number of times parent node was visited. When using UCB1, it is important to select optimal $\boldsymbol{c}$ because it affects playing strength. The value depends on the applied domain and the MCTS algorithm used. (Cowling et al. 2012) conducted an experiment where the PIMC and SO-ISMCTS players with exploration constant {0.25, 0.5, 0.75, ..., 1.75, 2} played repeatedly against PIMC with exploration constant 0,7. Performance of the algorithms decreased outside the range [0.5, 1], but inside the range none of the algorithms were particularly sensitive to the coefficient value. Thus, value of 0.7 has been used for all the algorithms in this work.
After **five** MCTS iterations on determinized game state, the built game tree is shown in the figure 3. MCTS steps will be explained using this game tree.
The MCTS consists of 4 stages, selection, expansion, simulation, backpropagation. The selection happens until algorithm encounters a node in the game tree which has unexplored moves. In our example algorithm will select a move with the Guard for the player A, then a move with the Priest for the player B and stop in that state. At the expansion state a child node which is not a part of the game tree yet is added to the tree. After that, at simulation step, game is played till the end and no new node is appended to the game tree. In our example player B won the simulated game. Then, at backpropagation step, starting from the leaf node of the game tree, win counter is updated for every node on the path to root where player B made move. Also, the visit count for every node in the path is updated.
At this point single MCTS iteration is completed. After all MCTS iterations, a counter for the move which was visited most is increased. Then determinization step is completed.
After all determinizations are completed, the root's child which was selected highest number of times is returned by the PIMC algorithm. If this move is Guard, algorithm also selects the guess card as in knowledge based agent (explained in previous section) .
It is important to select balanced number of determinization and MCTS iterations for single game tree. (Cowling et al. 2012) and (Powley et al. 2011) applied the PIMC for the Dou Di Zhu card game and found that as long as both parameters are sufficiently large, their precise values do not have a significant impact on playing strength. They used 40 determinizations with 250 MCTS iterations. In our implementation we have used 50 determinizations with 160 MCTS iterations.
This method has several disadvantages. Russel and Norvig (Russell et al. 2003) point out that determinization will never decide to make information gathering play (i.e. move which causes opponent to reveal his cards) or information hiding play (i.e. move that avoids revealing current player's hidden information). (Frank and Basin 1998) found two main problems of determinization:

- *Strategy fusion*: An AI agent incorrectly assumes that it can make different decisions from different determinizations in the same information set. For example, consider a game where card is taken from the Love Letter deck and put face down on the table. The player has two options: get 0.4 points without guessing the taken card or get 1 point if his guess was correct or 0 otherwise. The first choice's expected reward is 0.4. The second choice's reward is 0.125. So the first option is better. However, if PIMC is applied to this game, then in each determinization the card will be known and expected reward will be 1 point.

- *Non-locality*: Some states are very unlikely because other players will play away from the corresponding states. However, PIMC considers them too. For example, if player plays rationally, knows the opponent's card and could have played the Guard in order to win, he will play the Guard. However, in PIMC determinizations in which player holds the Guard will be considered. This case contradicts assumption of rationality.

Despite these problems many researchers successfully applied it in different domains. Ginsberg in (Ginsberg 2001) applied determinization to create AI for Bridge game which plays at human expert level. Bjarnson (Bjarnason et al. 2009) used this approach for the single-player card game Klondike Solitaire.

Single Observer-Information Set Monte Carlo Tree Search (SO-ISMCTS) algorithm

In SO-ISMCTS set of available actions from some node varies between visits to that node. For example, in our implementation a new deck is generated on every determinization and set of the available moves depends on the deck. Thus, a node in the game tree has branches to every legal move that was available from this state at some moment, but availability of a branch depends on current determinization. It is a multiarmed bandit (Auer et al. 2002), in which only subset of the arms are available on each trial. In the work of Cowling et al. (2012), it is called a *subset-armed bandit.* They replace n in formula 1 with the number of iterations in which the parent was visited and *node i was available.* This modification prevents over exploration of rare actions, i.e. actions which are available in few states of the information set. If every state in an information set has rare action, search will perform almost no exploitation and almost all exploration.
In SO-ISMCTS states are grouped together into information sets. It is improvement upon previous approach, designed not to suffer from strategy fusion (Whitehouse 2014). To solve the problem of learning in the context of specific states, SO-ISMTCS learns values in the context of information set instead. Thus, it is a tree where every node is information set and knowledge is shared during the learning between different determinizations. This approach does not need to find correct values for determinizations and simulations as in PIMC. The algorithm uses a new determinization on every iteration in order to take average value in final decision step. However, usually there is large number of states in the information set. For example, on the first move of the game there are 14! determinizations in the information set. Thus, it is infeasible to iterate over all determinizations in the information sets.
This algorithm is very similar to the PIMC approach except that is uses modified UCB1 formula in selection step and shares learned information during iterations. Therefore, SO-ISMCTS algorithm performs more iterations and greater playing strength is expected from it.
We will consider the determinization on the figure 2 example. After ten SO-ISMCTS iterations, Guard card has been selected nine times to be played by player A and it led to win six times. The Countess was played one time and Player A lost. This game tree is shown on the figure 4.

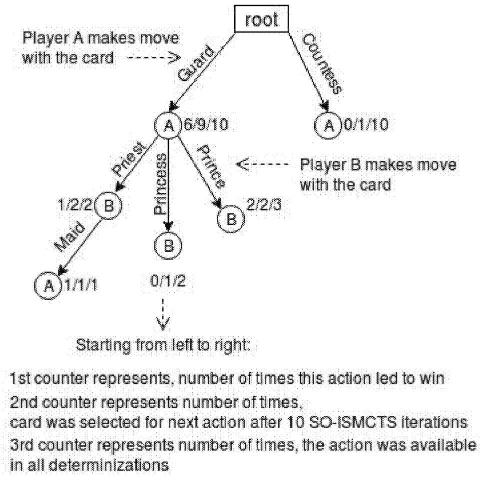


Figure 4: SO-ISMCTS Game Tree after 10 Iterations

Determinized Minimax

We have applied the Determinized Minimax approach as in Ginsberg (2001). Particularly, after each determinization, the vanilla minimax algorithm (Russell et al. 2003) is applied. After all determinizations, the move which was returned the most number of times by minimax is selected. There are 100 determinizations applied in author's implementation.
After determinization, the game state is fixed. That is, the order of cards in the deck and opponent's card will be the same. It means that algorithm can build the whole game tree. Because at each move, there are only 2 possible moves at most and 10 turns in the worst cases, the game tree will contain $O(2^{10})$ nodes and it is feasible to use full search here.

Minimax
Considering game state after determinization in the figure 2. In the minimax algorithm, there are two players, Max and Min, who make alternating moves. The Max players starts the game and authors assume that both players play optimally, that is they prefer the best possible result. The Max player tries to maximize own reward and Min player tries to minimize the Max player's result, that is to maximize his own gain. In other words, the Max player tries both moves and selects one with maximum value and Min player selects one with minimal value.
The algorithm starts from the terminal nodes where minimax values are known and then goes upwards and calculates a node's minimax value depending on the current player. After the whole tree is built, the algorithm selects the move which corresponds to the root's child node with maximum minimax value. In our case both moves have the same value, so algorithm will return any of them. The minimax algorithm performs complete ex-

ploration of the game tree. Let m be the maximal depth of the tree and b be a branching factor of each node, then the time complexity of the algorithm is $O(b^m)$ and space complexity is $O(bm)$. As it was mentioned above, the time complexity is reasonable in our case and a version of the algorithm without any optimizations can be applied here.

Experiments

In this section we will compare SO-ISMCTS with other techniques. Sometimes random deals have a significant effect on the outcome of the game. For example, at dealing stage, a player can get Princess and Baron cards and win immediately in 2-player game. To make a fair comparison between algorithms and reduce the variance of the results, we have extracted a set of 36 decks which does not give any advantages in the beginning of the round and used them in all experiments. The practice of specifying deck ordering in advance is common in Bridge tournaments between human players, to minimize the effect of luck when comparing players (Powley et al. 2011).
In comparison of two player version binomial distribution is assumed and error bars show 95% binomial proportion confidence interval. In all experiment there were 200 games played for each number of iterations. The first move is done by SO-ISMCTS algorithm. First moves on the successive rounds are done by the algorithm which won the previous one. All algorithms are implemented using Python 2 programming language. All implementations can be found in code repository [2].
First of all, we need to determine a number of iterations for SO-ISMCTS. To do this we have run a series of games against the same technique where first version played using 1000 iterations and number of iterations for the second version is varied. The results of the experiment is shown in the figure 5.

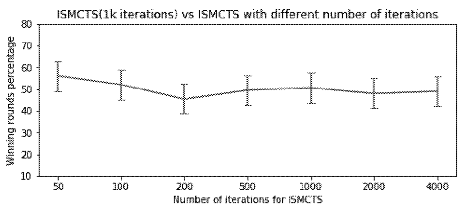


Figure 5: Winning Percentage for SO-ISMCTS with Varied Number of Iterations

It is shown that SO-ISMCTS with 1000 iterations shows almost the same performance as SO-ISMCTS with any other number of iterations . We will use 1000 iterations for SO-ISMCTS in all experiments for convenience.

[2]https://github.com/tamirOK/ISMCTS-for-love-letter-game

SO-ISMCTS vs PIMC

This section presents a comparison of SO-ISMCTS and PIMC. The number of iterations for PIMC will be varied. As it was mentioned in the PIMC section, we will use 50 determinizations. It means that if there are 1000 iterations, there will be 50 game trees built and for each tree there will be 20 MCTS iterations. Results of the experiment are shown in the figure 6.

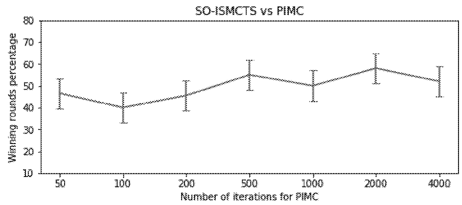


Figure 6: Winning Percentage for PIMC with Varied Number of Iterations

It is shown that initially SO-ISMCTS performs better. However, at 500 iterations PIMC wins approximately in 55% of cases, continues to win till the end of experiment and shows maximal performance of 58% with 2000 iterations.

SO-ISMCTS vs Determinized Minimax

This section presents a comparison of SO-ISMCTS and Determinized Minimax. It was mentioned that for each determinization, minimax builds a whole game tree and then traverse it. For this reason its performance is slower compared to the previous algorithms, so we decreased the number of iterations for the Determinized Minimax in our experiments. The results are shown below in figure 7.

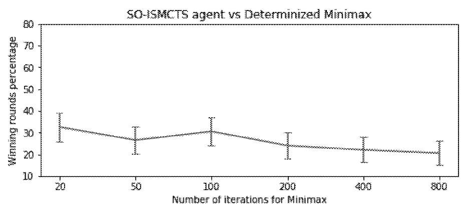


Figure 7: Winning Percentage for Determinized Minimax

It is observed that Determinized Minimax performance is low. The results provide interesting evidence that the classical approach for the perfect information game does not show good results for the Love Letter game.

SO-ISMCTS vs Knowledge based agent

In this section SO-ISMCTS is compared to knowledge based agent. Knowledge based agent does not have any simulations and always plays the same move for the fixed game state. For this reason we changed number of iterations for the SO-ISMCTS approach. Results of the experiment are shown in the figure 8.

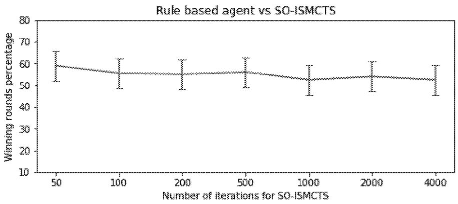


Figure 8: Winning Percentage for SO-ISMCTS with Varied Number of Iterations

It is seen that SO-ISMCTS shows better performance than knowledge based agent. However, its playing strength does not significantly outperform the knowledge based agent. The experiment shows that the game is stochastic and even simple deterministic approaches like knowledge based can win in half of simulations.

Conclusions

This paper presents an AI player for the game Love Letter. The AI player has been implemented using four algorithms among which SO-ISMCTS AI player's performance has been compared to knowledge based, PIMC and Determinized Minimax player. We have found that the playing strength is not statistically effected by the number of iterations played in Love Letter game. The results of the experiments have revealed SO-ISMCTS played much better than Determinized Minimax, slightly outperformed the knowledge based agent and plays a little weaker than PIMC. We can also conclude that the game is random and none of the algorithms could manage to take advantage of the randomness for the improved performance. We found out that PIMC and SO-ISMCTS agents cannot find the best move for every possible determinizations in the information set. This problem can possibly be minimized by increasing number of iterations for SO-ISMCTS, however it will increase running time of the algorithm.

Implementing the algorithm which will handle the specifics of the game, applying classical AI approaches like expectimax and use inferencing technique is a subject for future work. After solving this issue, the next step is to extend the algorithm to the multiplayer version and compare its performance to human players via human competitive testing. The implementation of the algorithm can support multi player version, the issue is opponent selection. Player selection can be done randomly or selecting the player who won maximum rounds so far or choosing the immediate player on your right or left etc. Another approach to improve the playing strength is to vary exploration-exploitation constant, number of iterations and adding more rules to the knowledge based agent.

REFERENCES

Auer P.; Cesa-Bianchi N.; and Fischer P., 2002. *Finite-time analysis of the multiarmed bandit problem. Machine learning*, 47, no. 2-3, 235–256.

Bjarnason R.; Fern A.; and Tadepalli P., 2009. *Lower Bounding Klondike Solitaire with Monte-Carlo Planning.* In *ICAPS*.

Browne C.B.; Powley E.; Whitehouse D.; Lucas S.M.; Cowling P.I.; Rohlfshagen P.; Tavener S.; Perez D.; Samothrakis S.; and Colton S., 2012. *A survey of monte carlo tree search methods. IEEE Transactions on Computational Intelligence and AI in games*, 4, no. 1, 1–43.

Cowling P.I.; Powley E.J.; and Whitehouse D., 2012. *Information set monte carlo tree search. IEEE Transactions on Computational Intelligence and AI in Games*, 4, no. 2, 120–143.

Frank I. and Basin D., 1998. *Search in games with incomplete information: A case study using bridge card play. Artificial Intelligence*, 100, no. 1-2, 87–123.

Ginsberg M.L., 2001. *GIB: Imperfect information in a computationally challenging game. Journal of Artificial Intelligence Research*, 14, 303–358.

Kanai S., 2012. *Love Letter : Tempest Edition.* Alderac Entertainment Group.

Powley E.J.; Whitehouse D.; and Cowling P.I., 2011. *Determinization in Monte-Carlo tree search for the card game Dou Di Zhu. Proc Artif Intell Simul Behav*, 17–24.

Russell S.J.; Norvig P.; Canny J.F.; Malik J.M.; and Edwards D.D., 2003. *Artificial intelligence: a modern approach*, vol. 2. Prentice hall Upper Saddle River.

Whitehouse D., 2014. *Monte Carlo Tree Search for games with Hidden Information and Uncertainty.* Ph.D. thesis, University of York.

GAME DEVELOPMENT METHODOLOGY

AFFECTIVE GAMES: A MULTIMODAL CLASSIFICATION SYSTEM

Salma Hamdy and David King
Division of Computing and Mathematics
SDI, Abertay University
DD1 1HG, Dundee,
Scotland
E-mail: {s.elsayed, d.king}@abertay.ac.uk

KEYWORDS
Affective games, physiological signals, emotion recognition.

ABSTRACT

Affective gaming is a relatively new field of research that exploits human emotions to influence gameplay for an enhanced player experience. Changes in player's psychology reflect on their behaviour and physiology, hence recognition of such variation is a core element in affective games. Complementary sources of affect offer more reliable recognition, especially in contexts where one modality is partial or unavailable. As a multimodal recognition system, affect-aware games are subject to the practical difficulties met by traditional trained classifiers. In addition, inherited game-related challenges in terms of data collection and performance arise while attempting to sustain an acceptable level of immersion. Most existing scenarios employ sensors that offer limited freedom of movement resulting in less realistic experiences. Recent advances now offer technology that allows players to communicate more freely and naturally with the game, and furthermore, control it without the use of input devices. However, the affective game industry is still in its infancy and definitely needs to catch up with the current life-like level of adaptation provided by graphics and animation.

1. INTRODUCTION

Affective computing (AC) (Picard 1997) is the science that aims to design and develop emotionally intelligent machines. Such automated systems should process and interpret human emotions via the analysis of sensory data. An affective model cannot be generic as applications vary in emotion models, available information, input devices and user requirements. For example, health care systems may require intrusive sensors to collect very reliable data, while e-learning and games may not demand such optimality and may or may not require additional controllers (Szwoch and Szwoch 2015). Overall, an affect recognition system is typically a trained classifier and, regardless of the application or input, includes components of traditional supervised classification (Fairclough, 2009). Human-Computer Interaction (HCI) applications further require system adaptation according to the predicted user emotion. This extends to affective games (AGs) where the goal is to increase engagement by explicitly or implicitly altering the game in response to players' emotions.

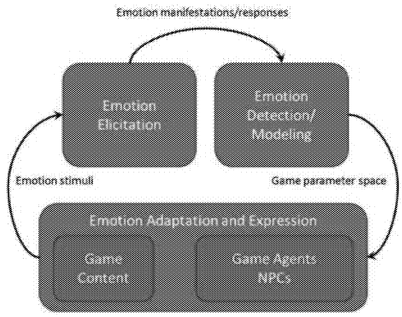


Figure 1. The realisation of the affective loop in games (Yannakakis and Paiva 2014).

Though a typical diagram of the affective loop in games (Fig. 1) does not reflect how the system infers the emotions, it implies the need to classify or estimate the response received from users (Novak et al. 2012). Very few affect-ware games truly reflect the concept of the full circle and are rather developed for academic research purposes. Commercial affective loops engage players' emotions through gameplay and other content in the development stage based on a representative player model (Adams 2014). This is problematic since individual players often differ from the average model, in addition to the rich spectrum of emotions experienced by players, which could change from sessions to session even for the same player, making it almost impossible to predict. However, this is likely to change with the advances in affect recognition techniques and input devices, allowing the capture of various information channels and more reliable predictions.

Fortunately, there are a variety of traditional classifiers that fit the task of emotion recognition and a large number of software libraries that make these classifiers available. Also, several emotion models and databases have been developed and standardised to an extent. Hence, the issue to consider often is what affective channels to acquire information from, and how to properly process them. Most attempts address the face as the main source of affect, while others involve speech, bodily and physiological signals. The latter has recently gained attention in the gaming context with the growth of affordable wearable technology and the well-established psychophysiological correlation (Christy and Kuncheva 2014). As observed by (Picard et al. 2001), physiological responses are translated to discrete psychological (emotional) states by a supervised classification pipeline. Furthermore, it is believed that commercial game publishers will start considering "psychophysiological hardware" in their next generation of game consoles (Valve Steam Box 2013).

A multimodal architecture was presented in (Hamdy 2016) as a generic model for affect-aware machines. It suggest a more reliable prediction by fusing different types of input information. This can naturally be extended to games and hence, a typical closed AG loop would include: multimodal emotion acquisition, modelling and identification of the collected signals via machine learning or statistical methods, and reflecting the decision back into the game engine to subsequently alter the game, ultimately taking into account the strength and type of the recognised affect (Christy and Kuncheva 2014). Variations of game adaptation includes dynamic difficulty adjustment (DDA), audiovisual content alterations, and affect-aware NPCs.

This paper discusses the external part of the AG loop as a multimodal recognition system, and reviews the different sources and methods of collecting affect information from players. In section 2, a number of modalities used as input to affective systems are discussed along with examples from the literature that employ these in games. Section 3 analyses these sources of affect in gaming context, and highlights relevant design issues of AGs as multimodal classifiers. Conclusions are presented in section 4.

2. AFFECTIVE INFORMATION

Attempting to improve classification tasks, it is recommended that multiple types of input from different modalities or different features from the same modality be combined (Gunes and Piccardi 2005). Hence, identifying psychological states from user biometrics requires that different types of measurements be provided simultaneously to allow one to verify the others (Drachen et al. 2010). The commonly used modelling approach for categorising emotions from mono- or multi-modal input is based on the arousal (high-low) and valence (positive-negative) dimensions in terms of the collected information. In addition to the apparent facial expressions and body movement, AGs often use monitoring modalities (Giakoumis et al. 2009) produced by the autonomic nervous system reflecting cardiovascular, electrodermal, or electrical activity in the human brain.

2.1 Behavioural

Vision channels hold the most informative data as humans tend to convey their feelings in a visual sense. The non-intrusive properties of cameras make vision-based systems more practical especially with the rapid advances in hardware and computer vision technology (Szwoch and Szwoch 2015). It is well-established that facial expressions and emotion mutually influence each other, hence the majority of affect recognition systems focus on face. Several features like Face Action Units or facial landmarks have been studied and benchmarked to model primary and secondary emotions in terms of selected dimensions. However, this is the least explored category in the literature with a limited number of studies addressing facial expression recognition in the context of games.

NovaEmötions (Mourão and Magalhães 2013) is a multiplayer game where players score by acting an emotion through facial expressions. The captured emotions are labelled using a multiclass Support Vector Machine (SVM) and the player with the closest expression wins a round. Authors claim the face images were captured in a novel and realistic setting despite the purpose being "act out an emotion" rather than spontaneously reacting to a stimulus. However, the experiment released a novel facial expression dataset of several emotions. Three AGs with linearly increasing difficulty were developed in (Bevilacqua et al. 2018) to investigate the relation between facial actions and heart rate, and player's emotional states. Expectedly, participants retained a neutral face for longer periods of time during the boring game parts. The study concluded that fusing the two cues is more likely to detect the emotional states. Authors in (Asteriadis et al. 2012a) used images of human faces and expressions in an attempt to assess the emotional state of a player. Player frustration and engagement as well as the challenge imposed by gameplay were used to alter the game in response. Other examples were previously discussed in (Hamdy and King 2017) to develop AGs through emotional NPCs that can believably respond to a player's facial affect.

Some affective expressions are reflected better through the body than the face. Cameras and motion detection devices enable the development of posture tracking techniques to construct models of body movement. The most common technique to capture motion is a suit with visibly trackable markers where posture is reconstructed by observing the subject with a camera and analysing the imagery. This is a well-established technique widely used in film animation and could easily be functional in games. Alternatively, markerless optical systems are available with no special equipment needed, like Microsoft's Kinect.

A simple yet very effective five-dimensional representation of body expressions was introduced in (Caridakis et al. 2010) and proved to have a strong association with how humans perceive emotions in real environments, making them strong candidates for affective HCI systems including games. In (Savva and Berthouze 2011), a motion capture system was attached to subjects playing a Wii tennis game to identify their affective states from non-acted body movements. The most dominant motions were used with a neural network (NN) classifier to identify eight emotions. Similarly, Kleinsmith et al. (2011) represented postures as rotations of the joints and assessed players in Wii sports games after winning or losing a point. Distance between body joints was used in (De Silva and Berthouze 2004) to recognise four basic emotions. Interestingly, the acted dataset of postures was labelled by observers from different cultures. The research in (Kapoor et al. 2004) examined non-acted postures through a multimodal system of facial expressions, body postures, and game state information. They reported the highest recognition accuracy from posture, although a limited description of the body was used. A system was proposed in (Gunes and Piccardi 2009) to identify emotions using a Hidden Markov Model (HMM) and a SVM to fuse facial and body cues to identify user affect. The database, however, did not include any real body pose information and was of a single subject.

Other vision-based modalities of player input that have been explored use pupilometers and gaze tracking, which are argued to be implausible within commercial development due to unreliability (sensitivity to distance, light and screen lamination) (Yannakakis et al. 2016). However, eye tracking is able to reveal information on attention from the duration of fixation, and hence is a good candidate for sensing player's engagement (Bradley et al. 2008; Xu et al. 2011)

Speech is one of the important behavioural modalities for detecting emotions. However, compared to facial expressions, emotions may not be captured as clearly in voice. In terms of vocal emotional dimensions, arousal is reflected by voice intonation and acoustics and has the strongest impact on speech, hence, can distinguish emotions better. Valence on the other hand is reflected by spoken words and is much harder to estimate from voice (Guthier et al. 2016).

Automatic speech recognition (ASR) is currently available on most low resource devices, smart phones, and game consoles, but mostly focus on the recognition of some context-dependant keywords. Although this is limited, it is a robust feature against possible interferences from game sounds, music and NPC voices in natural gaming environments. Hence, there is the trade-off of including a "heavy" continuous ASR engine in the game, or limiting the emotion analysis to a few affective words (Schuller 2016). It is important to note that even lower accuracies of ARS modules are proved to be sufficient to identify emotions from a word in a consistent context (Metze et al. 2010). In addition to words, nonverbal expression of emotions like laughter or groans convey a lot of information about the speaker's affective state, and can also be handled by the ASR engine.

Games have been used as means of eliciting emotions for data collection in speech research implying the rich spectrum of affect present in or by games (Schuller 2016). However, it is argued that a player is less likely to want to speak to the game (Jones and Deeming 2008) and only a few games truly made use of the ability to recognise emotion from speech.

A voice activated game for identifying four attitudes from children's speech was presented in (Yildirim et al. 2011). Spontaneous dialog interactions were carried out with computer characters and acoustic, lexical, and contextual features were captured. Interestingly, results showed that the selected features have varying performance with different assessed affective states and that fusion of all three cues significantly improved classification results.

Authors in (Jones and Sutherland 2005) developed a game with an acoustic recognition system to identify player's emotions from affective cues in speech and alter the behaviour of the game NPC accordingly. This was extended to a system capable of capturing 40 acoustic features from voice to assess five emotions where the character is better able to overcome obstacles based on the emotional state of the player.

In (Kim et al. 2004), affective speech and physiological signals were collected from players to elicit certain reactions in a pet NPC. A pre-selected set of features were used with a simple threshold-based classifier. Results showed improved accuracy when the two affective channels are combined. In a slightly different perspective, Rudra and Tien (2007) proved the feasibility of recognising voice emotions of a game character. Arbitrary utterances from the artificial Pidgin language was classified using a SVM to identify neutral and anger states of the NPC.

The work in (Alhargan et al. 2017) combined eye tracking with speech signals in a game that elicits controlled affective states. A SVM was used to classify emotions based on arousal and valence. Recognition results revealed eye tracking outperforming speech in affect detection and, when fused at decision level, the two modalities were complementary in interactive gaming applications.

2.2 Physiological

The Cognitive-motivational-emotive model assumes the human emotional state consisting of both affect and physiological response (Szwoch and Szwoch 2015). Visible affect can be controlled to an extent, but it is almost impossible for the average person to control physiological reactions. Furthermore, studies show that experienced players tend to stay still and speechless while playing (Asteriadis et al. 2012a), hence, the need for other forms of affect expressions. However, it is argued that in games, with enough practice, players' skills could allow them to control their physiological responses, converting an AG loop into a straight-forward biofeedback (Nacke et al. 2011).

The most popular biometric signals used in adaptive player-centric games are summarised in Table 1 with their correlation to emotion and feasibility in practice (Christy and Kuncheva 2014; Bontchev 2016; Garner 2016).

Typically, measurement of such signals requires standard hardware sensors borrowed from psychological research,

Table 1: The Common Physiological Signals for Affect Detection

Signal	Measurement/tool	Features
Electrodermal Activity (EDA) or Galvanic Skin Response (GSR) or Electrical Skin Response (ESR)	Electrical conductivity of the skin surface. A band between two fingers on either hand.	Reliable indicator of affective arousal like stress and anxiety. Simple and low cost. Common alone or combined with other techniques. Widely used for affect detection including in games. Easy to adapt well into games controllers. Suffers latency. Unsuitable for games with hand controllers unless sensors are attached to the controller.
Electromyography (EMG)	Electrical activity from muscles. Non-invasive electrodes.	Vary across subjects and cultures. Need to be placed at various body locations.
Electroencephalography (EEG)	Electrical signals from the brain. Non-invasive electrodes.	Used in various contexts and superior for games due to portability, ease of use, temporal resolution and affordability. Able to detect presence of emotions and identify the discrete classes. Excellent for examining attempts to conceal or pretend emotions. Spatial resolution is relatively low and may be insufficient for complex emotion detection.
Respiration	Breaths speed. Respiration belt or sensors.	Not as accurate as other signals. Mainstream applications could be hindered as sensors are embedded into clothing.
Blood volume	Heart rate and blood oxygen. Optical technology clip.	Good indicator of affective arousal like stress and anxiety.
Temperature	Body temperature. Contact and contactless sensors.	Related to specific emotional states. Has been used in games. Sensitive to movement causing inaccuracies in collected data.

which could be expensive and not suitable for active game play. Wearable sensors were introduced by (Picard and Healy 1997) for AC applications, which can be embedded in clothes or glasses making them fitting for AGs. Seamless sensors, where the user should not be aware of any interaction, come into contact with the body for a limited time through classical interfaces like mouse and keyboard. Some research attempts to incorporate traditional game controllers with such sensing ability. Scheirer et al. (2002) proposed a system that combines physiological data with behavioural data, namely mouse-clicking patterns, to build a HMM classifier of affect classification.

In (Christy and Kuncheva 2013), a fully functional mouse was designed with GSR and HR frequency measuring capability for capturing clean physiological signals from the player in real time. Rarely, an adaptive AG may use keyboard pressure as indication of changes in player effort or emotion during gameplay (Tijs et al. 2008). The game Rush for Gold (Bontchev and Vassileva 2016) used the GSR of the player to assess their arousal level and alter game components accordingly.

Attempts have been made to commercially produce multimodal affect-aware games. Companies had to limit their trials within the capabilities of existing sensing technology and emotion recognition algorithms. However, some were involved in manufacturing the necessary hardware and software to include affective elements in their games. Christy and Kuncheva (2014) provide a tabulated historical survey of AC, specifically psychophysiological system developments, and industry trends with respect to producing commercial AGs. Though unsuccessful, some "retro" systems employed the affective concept in player input since the early 80's with custom tailored sensing equipment. More recent AGs that made it to the commercial environment are found in (Kotsia et al. 2016).

3. MULTIMODAL AFFECTIVE GAMES DESIGN

Integrating AC into games involve interdisciplinary fields; signal processing, machine learning, and input from psychology. The sensing part of the AG loop is about fitting a supervised classifiers component into the design and development of the game while maintaining real-time performance and adaptation. Below, we discuss the feasibility of the modalities in section 2 in games context.

3.1 Modalities

Face and Body

Classification is one of the difficult tasks to automate, and when visual data is involved, this becomes more problematic. Facial and body movement prove very rich and useful for examining emotion expressions but are computationally expensive and time consuming (Kaplan et al., 2013). Camera-based modalities are highly within reach and do not require expensive equipment. However, the majority of vision-based affect detection systems cannot operate well in real-time (Zeng et al. 2009) and often require a well-lit environment that is not always available or preferred by gamers, in addition to posing privacy issues. Fortunately, this can be resolved to an extent by the advances in computer vision and hardware, and the increasing number of available vision-based emotion detection software. A rich collection of databases exist of facial expressions for primary and secondary emotions. However, due to the difficulty of obtaining natural emotions in experimental settings, only few databases exist that show spontaneous emotions. Real expressions could differ greatly from posed ones in terms of facial geometry and timing. This deems the majority of exiting datasets unsuitable for real-time generalisation especially in game environments where natural emotion is key. Furthermore, the validity of vision-based affect is highly subjective since observations vary between cultures, races and social environments (Jack et al., 2012). On another level, open space or collaborative games may require several cameras posing even further challenges of stereo-vision, real-time detection, and handling several occlusions due to space limitations and presence of several people.

Speech

As with visual cues, speech is a highly accessible real-time and unobtrusive modality, yet it is only applicable for games controlled by speech which are not that common. That is why few games up to this point make use of the ability to recognise emotion from speech, in addition to environmental audio posing additional challenges (Schuller 2016). Speech signals may not require as much processing power as visual cues and it is an advantage that sound recognition has been employed in HCI for quite some time, and with reliable performance. A rich number of affective speech resources are available although only a few cover different age groups with realistic spontaneous emotions. However, this is still missing for many languages and cultures. Similar to facial expressions, speech emotions are obtained by recording performing actors to acquire intense clean samples avoiding background noise that accompanies ordinary voice samples. The content is often scripted and meaningless for emotion detection as opposed to natural speech where some emotions appear more than others depending on mood. This increases the generalisation error of the trained detector in real environments. Furthermore, the validity of voice-based labelling in realistic recordings is highly subjective and prone to disagreement (Guthier et al. 2016). It is also worth noting that most datasets and ASR systems focus on verbal content rather than "animated" sounds like laughter and sighs, which seem to be the more common in a game environment.

Physiological signals

Even though the core technology for physiological signals is well founded and developed, hardware for affective gaming is still not widely available. Furthermore, most of these signals respond to other external/internal factors such as subject's health, physical condition, temperature, etc., deeming them unfit for the usual computer usage, not to mention gaming. Since video games mostly require active players, affective input to AG must be comfortable and intuitive as the sensors should not hinder player enjoyment of the game. AG would benefit best from seamless contact sensors, hence the need for them to be populated outside testing labs and into affordable consumer devices (Picard 2010). Nevertheless, great development is witnessed for non-intrusive low-powered sensors for remote collection of physiological and behavioural data from people. In addition, for existing hardware, real-time collection can be done through comfortable affordable wristbands and stored on local devices for further processing (Yannakakis et al. 2016). Furthermore, several computer manufacturers are considering embedding physiological sensors into game controllers (Szwoch and Szwoch 2015); Valve and Sony have implied that EDA and HR could soon be incorporated into standard controllers (Christy and Kuncheva 2014). A major leverage physiological signals have over other modalities as Table 1 indicates, is that they have been widely used in AC and games research and proved reliable indicators/classifiers of real-time emotions. Also, they do not lack generalised data, and are robust across the populous.

3.2 Model

Affect recognition systems, being a multimodal type of classifier, are more likely to incorporate methods applied in machine learning (ML) applications. Acquiring rich amounts of data from different affective signals seems appealing as it helps improve recognition and complement situations where some signals are not available. However, collecting physiological signals is subject to standard pre-processing and noise removal methods. Moreover, incompatibility, dimensions, and fusion of the collected signals present further challenges. Research in (Al Osman and Falk 2017; Calvo and D'Mello 2010) analyses automatic multimodal affect recognition and the challenges imposed by the need to acquire, process and fuse different types of data. Games pose additional challenges with respect to the ML model components, as many factors affect the collected data that not even carefully designed environments can eliminate without affecting player experience (Yannakakis et al. 2016).

Input signals

Most relevant sensors used in the gaming context are highly intrusive affecting the quality of gathered data. In addition, the fast-paced rich data from games may reflect rapid movement and quick alteration in emotions which may not be accurately captured or may even be missed. In general, physiological responses are affected by factors like mood, age, health in addition to external elements. When recording, it is often needed to offset the signal before modelling to calibrate the interaction model and eliminate subjective biases (Picard et al. 2001). This means a user will be recorded for a short resting time before any interaction, which may not be feasible for players. Nevertheless, it could be a suitable start for AG to exploit player dependent classifiers for better prediction. The tutorial level usually used to familiarise the player with the game, controls, and characters, can be exploited to calibrate the system to expressions of the specific player. This can also dynamically train AI companions to be accustomed with this player's forms of affect, hence more aware and believable in their responses. Surely, this raises feasibility issues and poses more constraints regarding system resources, game design and adaptation.

Features

Due to the rich affective interaction and the varying types of emotions experienced in games, the produced signals are complex and non-trivial to sample. Some extra features may need to be engineered for better distinction of displayed emotion. Standard extraction methods may suffice for AC applications, but for games, research shows that other complex methods such as sequence mining and deep learning offer richer representations of affect in games (Yannakakis et al. 2016). To reduce computational effort of training and real-time performance, it is best if the model is based on a minimal number of features that yield the highest prediction accuracy. Dimensionality reduction methods like principal component analysis (PCA) and Fisher's linear discriminant analysis (LDA) are all applicable, but current work in AG focussed so far on sequential forward selection, sequential backward selection and genetic search-based feature selection (Martínez and Yannakakis 2010). Another important issue to consider is the sampling rate. Most studies use an event-based approach where important game events determine the response time window that features are extracted from (Ravaja et al. 2006; Kivikangas et al. 2011).

Modelling (Classification)

Mapping features to emotions primarily depends on the representation model of emotions. If classes or annotated states are used to model players affect, any of the traditional ML algorithms can be used to build an affective classifier. On the other hand, if a pairwise preference (rank) format is used, the problem becomes a preference learning (Yannakakis 2009). Dynamic models of player behaviour can be used to infer affect in real time and induce appropriate emotions during gameplay (Bontchev 2016). However, Novak et al. (2012) conclude that the majority of adaptive physiological systems use static data fusion methods. The practical challenges result in emotions being identified with a widely varying accuracy (51%–92% according to (Nicolaou et al. 2011)) over the literature. Nevertheless, it is fair to say that a margin of error is allowed in games as an entertainment media (Christy and Kuncheva 2014). If the AG convinces players it is recognising and interacting with their emotions, then occasional misclassifications should not have a significant impact on the player's experience. This can relax the design constraint put on the system, especially for commercial products.

4. CONCLUSION

Although game developers have traditionally focused their efforts on improving the graphic quality of games, speculations is that the advancements in graphics will plateau, forcing them to discover new ways of adding attraction to their games. This is expected to open a commercial perspective for AG (Christy and Kuncheva 2014; Lara-Cabrera and Camacho 2018), which are basically classification systems with a variety of biometrics preferred as input.

Behavioural affective inputs are highly accessible but add the traditional challenges associated with audiovisual data processing and hence, require robust algorithms with higher generalisation level. Besides, with games being a global entertainment industry, cross-cultural and social experiences influence on emotions must be addressed (Kleinsmith et al. 2006; Sauter et al. 2010). Ethical implications arise when the game requires to audiovisually record players consistently, which are barely addressed in the literature.

Physiological signals offer a commonly acceptable alternative. Contact-based sensors produce a wider range of reliable, objective and quantitative data (Guthier et al. 2016). However, most existing biometric sensors are rather impractical and highly intrusive for interactive applications and some are still very costly for a broad use in gaming. Also, wearable devices can obscure a significant part of the face/body and influence players to exhibit unusual behaviour, even subconsciously, which may affect interaction and subsequent actions. In such a context, information from different channels is required.

Studies show that behavioural and physiological signals can be used to model players state continuously during interactive gameplay without interruptions, making the gathered data more temporally reliable as opposed to post-game interviews and questionnaires (Mandryk et al. 2006). Although it is evident from the literature that combining modalities of different types increases classification precision, novel methods for modelling/predicting interactions are required and efficient fusion of multimodal data remains an open problem. Nevertheless, while reliable recognition seems required, independent of external factors or personality profiles, Christy and Kuncheva (2014) suggest that AG should not exclusively rely on accuracy of emotion recognition. Clever game design can reimburse misclassifications for an uninterrupted game experience.

The most obvious way to represent emotion computationally is as labels for a limited number of discrete emotion categories. This scheme is easy to implement, but may be too general to be useful. Samples of affective data are often obtained from laboratory experiments with limited context, mostly of acted postures or stereotypical expressions (Kotsia et al. 2016). Picard (2000) highlighted the common emotions experienced or expressed around computer games, and the significance of systems that can recognise such affect from players. This can narrow the gap in HCI with development of more user-centred systems (Hudlicka 2003) when trained on emotions more likely related to gamers. However, emotion recognition is mostly done to standard predefined classes as spontaneity is an extra challenge (Kotsia et al. 2016). The experimental research is often done in heavily controlled environments limiting its chances of being deployed in practice, and results of AG research conducted in commercial settings are rarely published.

According to (Borod et al. 1998), the valence hypothesis suggests that there is a difference between processing and displaying positive and negative emotions. Hence, it may be obvious not to treat all basic emotions equally as it is less likely that all emotions will occur with the same probability in daily life. This however, could be slightly different for games as the genre, content or level are most likely intended to elicit particular affective states. In relevance, one thing to consider with affect-aware games is signal habituation (Sokolov 1963). Getting too familiar with the stimulus, such that bodily reactions tend not to be triggered as much, is a phenomenon commonly observed with experienced gamers or people who spend a lot of time on the same game or level. Successful interaction design should be dynamic enough to offer ranges of stimuli and keep the game exciting (Garner 2016).

It also worth noting that the majority of research in AG addresses single player scenarios. Physical space limitations are understandable, in addition to the added complexity of having to track and process biometrics of multiple players in a virtual environment, while keeping up system performance. Moreover, modelling multiplayer free interaction and how it influences their subsequent emotions is still a novel field of research (Kotsia et al. 2016). Although emotion recognition requires to be a real-time application with reasonable resources and ability to run on local platforms, it is a huge advantage to be able to distribute the recognition between the local console and a server (Schuller 2016).

Although home consoles do not by default incorporate biometrics, research shows that interest in biofeedback applications is growing and it is anticipated that in ten years, biometrics within games will become mainstream (Garner 2016). This move should inspire the game industry to consider design and development of AG loops in their products. It is argued that the future of affective gaming lies in more sophisticated, smaller, noise-free devices (Kotsia et al. 2016; Christy and Kuncheva 2014). Fitting affective input devices and fast reliable pattern recognition algorithms in a game, while maintaining the desired game adaptation, is the biggest challenge for AG, especially in products affordable to the average player.

REFERENCES

Adams E. 2014. *Fundamentals of Game Design*. New Riders Publishing, USA.

Al Osman H. and T.H Falk. 2017. "Multimodal Affect Recognition: Current Approaches and Challenges". In *Emotion and Attention Recognition Based on Biological Signals and Images*, S.A. Hosseini (Eds.). IntechOpen, 59-86.

Alhargan A.; N. Cooke; and T. Binjammaz. 2017. "Multimodal Affect Recognition in an Interactive Gaming Environment using Eye Tracking and Speech Signals". In Proceedings of the 19th ACM International Conference on Multimodal Interaction (Glasgow, UK, Nov.13-17). ACM, N.Y., 479-486

Asteriadis S.; K. Karpouzis; N. Shaker; and G.N. Yannakakis. 2012. "Does your Profile Say it All? Using Demographics to Predict Expressive Head Movement During Gameplay". In *20th Conference on User Modeling, Adaptation, and Personalization (UMAP 2012) Workshops* (Montreal, Canada, Jul.16-20) User Modeling Inc. 11.

Asteriadis S.; N. Shaker; K. Karpouzis; and G.N. Yannakakis. 2012. "Towards Player's Affective and Behavioral Visual Cues as Drives to Game Adaptation". In *LREC Workshop on Multimodal Corpora for Machine Learning* (Istanbul, Turkey, May 22). LREC, 6-9.

Bevilacqua F.; H. Engström; and P. Backlund. 2018. "Changes in Heart Rate and Facial Actions During A Gaming Session with Provoked Boredom and Stress". *Entertainment Computing*, No.24 (Jan), 10-20.

Bontchev B. and D. Vassileva. 2016. "Assessing Engagement in An Emotionally-Adaptive Applied Game". In *Proceedings of the Fourth International Conference on Technological Ecosystems for Enhancing Multiculturality* (Salamanca, Spain, Nov.2-4). ACM, N.Y., 747-754.

Bontchev B. 2016. "Adaptation in Affective Video Games: A Literature Review". *Cybernetics and Information Technologies* 16, No.3 (Sep), 3-34.

Borod J.C.; B.A. Cicero; L.K. Obler; J. Welkowitz; H.M. Erhan; C. Santschi; I.S. Grunwald; R.M. Agosti; and J.R. Whalen. 1998. "Right Hemisphere Emotional Perception: Evidence Across Multiple Channels". *Neuropsychology* 12, No3 (July), 446-458.

Bradley M.M.; L. Miccoli; M.A. Escrig; and P.J. Lang. 2008. "The Pupil as A Measure of Emotional Arousal and Autonomic Activation". *Psychophysiology*, 45 No4 (July), 602-607.

Calvo R.A. and S. D'Mello. 2010. "Affect Detection: An Interdisciplinary Review of Models, Methods, and their Applications". *IEEE Transactions on Affective Computing* 1, No1 (Jan), 18-37.

Caridakis G.; J. Wagner; A. Raouzaiou; Z. Curto; E. Andre; and K. Karpouzis. 2010. "A Multimodal Corpus for Gesture Expressivity Analysis". In *Multimodal Corpora: Advances in Capturing, Coding and Analyzing Multimodality Workshop* (Malta, May 18). LREC, 80.

Christy T. and L.I. Kuncheva. 2013. "Amber Shark-fin: An Unobtrusive Affective Mouse". In *The Sixth International Conference on Advances in Computer-Human Interaction* (Nice, France, Feb.24-Mar.1). IARIA XPS Press, 488-495.

Christy T. and L.I. Kuncheva. 2014. "Technological Advancements in Affective Gaming: A Historical Survey". *GSTF Journal on Computing (JoC)* 3, No4 (Apr), 32-41.

De Silva P.R. and N. -Berthouze. 2004. "Modeling Human Affective Postures: An Information Theoretic Characterization of Posture Features". *Computer Animation and Virtual Worlds* 15, No3-4 (July), 269-276.

Drachen A.; L.E. Nacke; G.N. Yannakakis, G.N.; and A.L. Pedersen. 2010. "Correlation Between Heart Rate, Electrodermal Activity and Player Experience in First-Person Shooter Games". In *Proceedings of the 5th ACM SIGGRAPH Symposium on Video Games* (L.A., California, July 28-29) ACM, N.Y., 49-54.

Fairclough S.H. 2008. "Fundamentals of Physiological Computing". *Interacting with Computers* 21, No.1-2 (Nov), 133-145.

Garner T.A. 2016. "From Sinewaves to Physiologically-Adaptive Soundscapes: The Evolving Relationship Between Sound and Emotion in Video Games". In *Emotions in Games*. K. Karpouzis and G.N. Yannakakis (Eds.). Springer International Publishing, Switzerland, 197-214.

Giakoumis D.; A. Vogiannou; I. Kosunen; D. Devlaminck; M. Ahn; A.M. Burns; F. Khademi; K. Moustakas; and D. Tzovaras. 2009. "Multimodal Monitoring of the Behavioral And Physiological State of The User in Interactive VR Games". In *Proceedings of Multimodal Interfaces eNTERFACE'09* (Genoa, Italy, July13-Aug.7), 17.

Gunes H. and M. Piccardi. 2005. "Affect Recognition from Face and Body: Early Fusion Vs. Late Fusion". In *IEEE International Conference on Systems, Man and Cybernetics* (Waikoloa, HI, USA, Oct.12). IEEE, 3437-3443.

Gunes H. and M. Piccardi. 2009. "Automatic Temporal Segment Detection and Affect Recognition from Face and Body Display". *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)* 39, No1 (Feb), 64-84.

Guthier B.; R. Dörner; and H.P. Martinez. 2016. "Affective Computing in Games". In *Entertainment Computing and Serious Games,* R. Dörner et al. (Eds.). Springer International Publishing, 402-441.

Hamdy S. 2018. "Human Emotion Interpreter." In *Proceedings of SAI Intelligent Systems Conference (IntelliSys) 2016. IntelliSys 2016. Lecture Notes in Networks and Systems* 15, S. Kapoor et al. (Eds.). Springer, 922-931.

Hamdy S. and D. King. 2017. "Affect and Believability in Game Characters – A Review of The Use of Affective Computing in Games". In *GameOn'17 – The Simulation and AI in Games Conference* (Carlow, Ireland, Sep.6-8). Eurosis, 90-97.

Hudlicka E. 2003. "To Feel or Not to Feel: The Role of Affect in Human–Computer Interaction". *International Journal of Human-computer Studies* 59, No.1-2 (July), 1-32.

Jack R.E.; O.G. Garrod; H. Yu; R. Caldara; and P.G. Schyns. 2012. "Facial Expressions of Emotion Are Not Culturally Universal". *Proceedings of the National Academy of Sciences* 109, No.19 (May), 7241-7244.

Jones C. and A. Deeming. 2008. "Affective Human-robotic Interaction". In *Affect and Emotion in Human-Computer Interaction. Lecture Notes in Computer Science* 4868, Peter C. and Beale R. (Eds.). Springer, 175-185.

Jones C.M. and J. Sutherland. 2005. "Creating an Emotionally Reactive Computer Game Responding to Affective Cues in Speech". In *Proceedings of the 19th British HCI Group Annual Conference* 2. Springer-Verlag, 1-2.

Kaplan S.; R.S. Dalal; and J.N. Luchman. 2013. "Measurement of Emotions". In *Research Methods in Occupational Health Psychology: Measurement, Design, and Data Analysis*. R.R. Sinclair et al. (Eds.). Taylor & Francis Group, FL, 61-75.

Kapoor A.; R.W. Picard; and Y. Ivanov. 2004. "Probabilistic Combination of Multiple Modalities to Detect Interest". In Proceedings of the 17th International Conference on *Pattern Recognition* (ICPR 2004) (Cambridge, UK, Aug.26). IEEE. 969-972.

Kim J.; N. Bee; J. Wagner; and E. André. 2004. "Emote to Win: Affective Interactions with a Computer Game Agent". *Lecture Notes in Informatics (LNI)* 50, No.1 (Sep), 159-164.

Kivikangas J.M.; G. Chanel; B. Cowley; I. Ekman; M. Salminen; S. Järvelä; and N. Ravaja. 2011. "A Review of the Use of Psychophysiological Methods in Game Research". *Journal of Gaming and Virtual Worlds* 3, No.3 (Sep), 181-199.

Kleinsmith A.; P.R. De Silva; and N.B -Berthouze. 2006. "Cross-cultural Differences in Recognizing Affect from Body Posture". *Interacting with Computers* 18, No.6 (June), 1371-1389.

Kleinsmith A.; N.B. -Berthouze; and A. Steed. 2011. "Automatic Recognition of Non-acted Affective Postures". *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)* 41, No.4 (Aug), 1027-1038.

Kotsia I.; S. Zafeiriou; G. Goudelis; I. Patras; and K. Karpouzis. 2016. "Multimodal Sensing in Affective Gaming". In *Emotions in Games*. K. Karpouzis and G.N. Yannakakis (Eds.). Springer International Publishing, Switzerland, 59-84.

Lara-Cabrera R. and D. Camacho. 2018. "A Taxonomy and State of the Art Revision on Affective Games". In press to appear in *Future Generation Computer Systems*. (Jan), North-Holland. [Available at DOI: 10.1016/j.future.2017.12.056].

Mandryk R.L.; M.S. Atkins; and K.M. Inkpen. 2006. "A Continuous and Objective Evaluation of Emotional Experience with Interactive Play Environments". In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (Montréal, Québec, Canada, Apr.22-27). ACM, N.Y., 1027-1036.

Martínez H.P and G.N. Yannakakis. 2010. "Genetic Search Feature Selection for Affective Modeling: A Case Study on Reported Preferences". In *Proceedings of the 3rd International Workshop on Affective Interaction in Natural Environments* (Firenze, Italy, Oct.29) ACM, N.Y., 15-20.

Metze F.; A. Batliner; F. Eyben; T. Polzehl; B. Schuller; and S. Steidl. 2010. "Emotion Recognition Using Imperfect Speech Recognition". In *Proceedings of INTERSPEECH,* ISCA, 478–481.

Mourão A. and J. Magalhães. 2013. "Competitive Affective Gaming: Winning with a Smile". In *Proceedings of the 21st ACM International Conference on Multimedia* (Barcelona, Spain, Oct.21-25). ACM, N.Y., 83-92.

Nacke L.E.; M. Kalyn; C. Lough; and R.L. Mandryk. 2011. "Biofeedback Game Design: Using Direct and Indirect Physiological Control to Enhance Game Interaction". In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (Vancouver, Canada, May7-12). ACM, 103-112.

Nicolaou M.A.; H. Gunes; and M. Pantic. 2011. "Continuous Prediction of Spontaneous Affect from Multiple Cues and Modalities in Valence-arousal Space". *IEEE Transactions on Affective Computing* 2, No.2 (Apr), 92-105.

Novak D.; M. Mihelj; and M. Munih. 2012. "A Survey of Methods for Data Fusion and System Adaptation Using Autonomic Nervous System Responses in Physiological Computing". *Interacting with Computers* 24, No.3 (May), 154-172.

Picard R.W. 1997. *Affective Computing*. The MIT Press, Cambridge, MA.

Picard R.W. and J. Healey. 1997. "Affective Wearables". *Personal Technologies* 1, No.4 (Dec), 231-240.

Picard R.W. 2000. "Toward Computers that Recognize and Respond to User Emotion". *IBM systems journal* 39, No3.4, 705-719.

Picard R.W; E. Vyzas; and J. Healey. 2001. "Toward Machine Emotional Intelligence: Analysis of Affective Physiological State". *IEEE Transactions on Pattern Analysis and Machine Intelligence* 23, No.10 (Oct), 1175-1191.

Picard R.W. 2010. "Emotion Research by the People, for the People". *Emotion Review* 2, No.3 (June), 250-254.

Ravaja N.; T. Saari; M. Salminen; J. Laarni; and K. Kallinen. 2006. "Phasic Emotional Reactions to Video Game Events: A Psychophysiological Investigation". *Media Psychology* 8, No.4 (Nov), 343-367.

Rudra T; M. Kavakli; and D. Tien. 2007. "Emotion Detection from Male Speech in Computer Games". In *TENCON 2007-2007 IEEE Region 10 Conference* (Taipei, Taiwan, Oct.30-Nov.2). IEEE, 1-4.

Sauter D.A.; F. Eisner; P. Ekman; and S.K. Scott. 2010. "Cross-cultural Recognition of Basic Emotions through Nonverbal Emotional Vocalizations". *Proceedings of the National Academy of Sciences* 107, No.6 (Feb), 2408-2412.

Savva N. and N.B. -Berthouze. 2011. "Automatic Recognition of Affective Body Movement in a Video Game Scenario". In *Intelligent Technologies for Interactive Entertainment. INTETAIN 2011. Lecture Notes of the Institute for Computer Sciences, Social Informatics and Telecommunications Engineering 78*. Springer, Berlin, 149-159.

Scheirer J.; R. Fernandez; J. Klein; and R.W. Picard. 2002. "Frustrating the User on Purpose: A Step Toward Building an Affective Computer". *Interacting with Computers* 14, No.2 (Feb), 93-118.

Schuller B. 2016. "Emotion Modelling via Speech Content and Prosody: In Computer Games and Elsewhere". In *Emotions in Games*. K. Karpouzis and G.N. Yannakakis (Eds.). Springer International Publishing, Switzerland, 85-102.

Sokolov E.N. 1963. "Higher Nervous Functions: The Orienting Reflex". *Annual Review of Physiology* 25, No.1 (Mar), 545-580.

Szwoch M. and W. Szwoch. 2015. "Emotion Recognition for Affect Aware Video Games". In *Image Processing & Communications Challenges 6. Advances in Intelligent Systems and Computing* 313, R. Choraś (Eds.). Springer, 227-236.

Tijs T.J.; D. Brokken; and W.A. IJsselsteijn. 2008. "Dynamic Game Balancing by Recognizing Affect". In *Fun and Games. Lecture Notes in Computer Science* 5294, P. Markopoulos et al. (Eds.). Springer, Berlin, 88-93.

Xu J.; Y. Wang; F. Chen; H. Choi; G. Li; S. Chen; and S. Hussain. 2011. "Pupillary Response Based Cognitive Workload Index Under Luminance and Emotional Changes". In *CHI'11 Extended Abstracts on Human Factors in Computing Systems* (Vancouver, Canada, May7-12). ACM, N.Y. 1627-1632.

Yannakakis G.N. 2009. "Preference Learning for Affective Modeling". In *The 3rd International Conference on Affective Computing and Intelligent Interaction and Workshops (ACII 2009)* (Amsterdam, Netherlands, Sep.10-12) IEEE. 1-6.

Yannakakis G.N. and A. Paiva. 2014. "Emotions in Games". In *The Oxford Handbook of Affective Computing*, R. Calvo et al. (Eds.). Oxford University Press, 459-471.

Yannakakis G.N.; H.P. Martinez; and M. Garbarino. 2016. "Psychophysiology in Games". In *Emotions in Games*. K. Karpouzis and G.N. Yannakakis (Eds.). Springer International Publishing, Switzerland, 119-137

Yildirim S.; S. Narayanan; and A. Potamianos. 2011. "Detecting Emotional State of A Child in A Conversational Computer Game". *Computer Speech & Language* 25, No.1 (Jan), 29-44.

Zeng Z.; M. Pantic; G.I. Roisman; and T.S. Huang. 2009. "A Survey of Affect Recognition Methods: Audio, Visual, and Spontaneous Expressions". *IEEE Transactions on Pattern Analysis and Machine Intelligence* 31, No.1 (Jan), 39-58.

WEB REFERENCES

Exclusive Interview: Valve's Gabe Newell on Steam Box, biometrics, and the future of gaming. 2013. Available from: https://www.theverge.com/2013/1/8/3852144/gabe-newell-interview-steam-box-future-of-gaming. [Accessed May 2018].

AUTHOR BIOGRAPHIES

SALMA HAMDY obtained her degrees in computer science from Ain Shams University in Egypt. She is a lecturer at Abertay University in Scotland. Her research interests include computer vision, AI, and affective computing.
Email: s.elsayed@abertay.ac.uk.

DAVID KING is a lecturer in Maths and Artificial Intelligence at Abertay University teaching on the Computer Games Technology and Computer Games Application Development Programmes. Email: d.king@abertay.ac.uk.

Player Age and Affordance Theory in Game Design

Hamna Aslam, Joseph Alexander Brown, Elizabeth Reading
Artificial Intelligence in Games Development Lab, Innopolis University, Russia, 420500
email: h.aslam@innopolis.ru, j.brown@innopolis.ru, beth_reading@hotmail.com

KEYWORDS

Game Design, Intuitive Design Analysis, Human Factors in Board Games, Affordance Theory in Game Design

ABSTRACT

Games are for everybody and should be designed to be intuitive for all people irrespective of gender, age, or occupation. However, do human factors such as age and years of education contribute to the difference in the level of intuitiveness of the game design among different people? This paper presents a study conducted to examine board games from the perspective of their level of intuitiveness for different people. The participants of this study are categorized into three groups, students (School and University) and professors. The groups differ considerably in age, educational years, and gameplay experience. The games presented (Hive Pocket and Hanabi) have had their rule books removed and participants were required to figure out game rules merely from the design of game pieces. Results show that increased years of education and higher age encourages logical thinking to explore gameplay, which ultimately leads to a correct understanding of the game. Participants could still figure out the basic theme of the game without paying much attention to the associativity among game components, but this does not lead them to finding the actual game mechanics.

INTRODUCTION

This paper presents a study to investigate the impact of the human factors of age and education on the player's ability to understand the game mechanics and intuitively deduct from game objects use.
Addressed are three questions:

1. Does age affects a player's ability to observe intuitiveness in game objects?

2. Are games designed for all age groups?

3. Do more years of education mean more wisdom in playing?

The first and second questions are addressed by selecting participants with three age groups. The groups consisted of participants with an average age of fourteen, twenty two and forty years. The second question is investigated using the same groups as fourteen years old were school students, the young adults were university students and the third group had participants who were all professors with at least ten years of experience in research in the IT, mathematics, and computing domains. The motivation for our study is to promote universal design for games. As stated in Story et al. (1998):

> Universal design can be defined as the design of products and environments to be usable to the greatest extent possible by people of all ages and abilities. Universal design respects human diversity and promotes inclusion of all people in all activities of life. It is unlikely that any product or environment could ever be used by everyone under all conditions. Because of this, it may be more appropriate to consider universal design a process, rather than an achievement.

The role of age and experience in play is becoming an important area of study due to the changing user demographics of digital games. These demographics show a larger trend towards players becoming older. The 2016 and 2017 reports (Entertainment Standards Association 2016; 2017) by the ESA on player demographics shows an average player age of 35, up four years from 31 in the 2014 report (Entertainment Standards Association 2014). Not only are players aging, but new players are entering the market. As players age, their abilities and goal sets also change. In a large user study of first person shooter players, (Tekofsky et al. 2015) found that increased age leads to players being less skilled, resulting in a lower kills to deaths ratio, however, older players were more likely to resort to objective based play, focusing on winning the game via factors such as control points. They were less risky in their actions, favoring character classes and vehicles which promoted objective play and safety; medics and tanks over assault troops and jet fighters. These focuses on objective play may aid in board games where there is a requirement for deliberative thought, and turns allow for player thinking time not to be the constraint in many cases.
In the context of better designs, (Browne 2015) talks about embedding the rules in the objects of the game to make player avoid mistakes. The idea is similar to *poka yoke* which is a Japanese term used for any mechanism

that aids people in industries to avoid mistakes especially while handling equipments. The method demonstrates some games that already have poka yoke factor which leads players in the right direction without compromising the fun element in the game. (Yermolaieva and Brown 2017) examine dice as a basic object in a game experience and demonstrate that even small factors such as shape, symbols, contrast, and size in the design can affect gameplay during a user study.
(Norman 2013) examines the role of design developing a conceptual model, the object provides *affordances* for actions, i.e. what actions are physically/conceptually possible for the user, and *signifiers* to actions such as which set of existing actions are highlighted to the user. If the affordances in the design are well developed to prevent errors from occurring, then it is a poka yoke. If the design restricts errors via messages, text, or symbols, then this is a signifier. Gibson's approach on affordances conveys that: *"The affordances of the environment are what it offers the animal, what it provides or furnishes, either for good or ill"* (Gibson 2014). Gibson further expands that object's affordances are a set of ways an actor can relate to an object (Cardona-Rivera and Young 2013). The affordances are present in the environment to be perceived and as individuals explore the environment, their perception gets wider and richer(Gibson 2014). (Norman 2013) has described seven stages of action, one for goals, three for execution and three for evaluation. These include-in order (1) *Goal*-forming the goal (2) *Plan*-the action (3) *Specify*-an action sequence (4) *Perform*-an action sequence (5) *Perceive*-the state of the world (6) *interpret*-the perception and (7) *compare*-the outcome with the goal. These seven stages of action are influenced by an individual's age and experience and age affects affordances at how an individual perceive's the world and how they interpret their perceptions. The research methodology explained in the following section enables to analyze an individual's affordances both from Norman and Gibson's idea of affordances.
Good design give the opportunities to explore or leads the user to the correct development of a conceptual model, ideally using poka yoke, and otherwise using clear signifiers. Rule books, much like the manuals which Norman does not endorse, are antithetical to good designs. Hence, when looking at the games for this study, they have been removed, leading our players to only examine the parts of the game. This expands on Daviau's anecdotal study(Daviau 2011) in which a group of 25 MIT researchers in pairs were given a set of board games and asked to determine the rules of the game without rulebooks in only five minutes. Daviau reported that "they did an amazing job," but there was no rigorous attempt to examine the reasoning for this outcome, though he goes on to agree with a good design of a game being one with as little need for a rule book as possible.
The remainder of the paper discusses research methodology, describes our rationale behind using Hanabi and Hive Pocket for playtesting, present responses from students and professors about game play and analyse participant's responses qualitatively and quantitatively. The following sections detail implications of our research and draw conclusions as well as give directions for future research.

Research Methodology

The methodology for our research has been adopted from Daviau's anecdotal study (Daviau 2011). We presented two games to each participant and the rule books as well as the packaging that has rules written over it were removed as shown in Figure 1 and Figure 2. The participants had thirty minutes to figure out the rules for the game play. During the experiment, the participants were not allowed to use the Internet or discuss with each other.
The questionnaire presented to all participants had three questions: 1) how many players can play this game and who plays first, 2) what are the rules of the game and the rationale behind each rule, and 3) what is the winning condition.

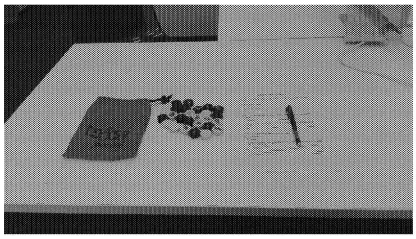


Figure 1: Hive Pocket for User Testing

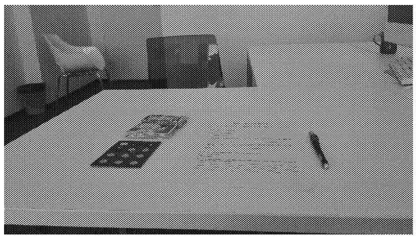


Figure 2: Hanabi for User Testing

The participants are grouped by the age factor mainly, which also led to the difference in educational years for each group. The forty two student participants are university students studying Information Technology. Their average age is twenty two years. The students have spent an average of three hours playing board games and an average of twenty one hours playing digital games over the period of past thirty days. The twenty three teenagers were the students of school with an average age of fourteen years. They have spent six hours on average playing board games and ten hours playing digital games.

The sixteen professors who participated in the study are active researchers in different spheres of information technology. Thirteen of the professors had not played any games in the last thirty days, of the three others, two had played for only a single hour, and the last had played for fifteen hours. This is a statistically significantly lower rate of gameplay ($p < 0.05$).

The participants were given maximum thirty minutes for each game. They were also permitted to exchange thoughts and opinions with the observer. At the end of the user test, the observer played the game with each participant to demonstrate the actual game rules. During this process, participant's emotional responses such as excitement after knowing the rules, comments about the game and game design were also recorded. Both student groups consisted of Russian citizens and the professors group included various nationalities.

For this study we have not considered gender and demographic variables, acknowledging that these are significant for analysis. The research paper presently focuses on broader theme such as age and education factor to find common ingredients for universal design. Later the research will be continued to break down data on demographic variables.

Rationale for selecting Hive Pocket and Hanabi for play testing

Board games vary and there is no restriction on the type and quantity of objects, colors, and patterns used for game design. Game designers try to make a game design as interesting as possible. They compel players by introducing variety of schemes and combinations of object while attempting to prevent objects from introducing unnecessary confusion and complexity in the game. When this process is done well - the design can be said to be intuitive. The extent of intuitiveness of the design can be checked by the player's responses.

While there is research going on finding and investigating factors that contribute to the intuitiveness of the game design, the aim of this study is to investigate the influence of human factors such as age and years of education on the level of intuitiveness of a game to a user. For this purpose we selected games whose objects do not give a direct indication about game play through written text.

Hive Pocket (Yianni 2001) is appealing since it has pieces of two colors with pictures of insects over them. The pieces do not have any text written over them but gives an indication about number of players through two colors. Other than the color, the hexagonal shape of objects gives an indication that they must be connected in a certain way. Number of pieces with a certain insect such as one Honey Bee, two Spiders, three Beetles etc. and the name of the game, Hive Pocket, give an indication about the importance of Honey Bee as a winning goal. Moreover, the different type of insects also give an indication about the moves or actions associated with them. Overall, Hive Pocket's objects are acting as signifiers without any text written over them. This allows us to determine to what extent participants with different ages and educational years are able to discover hints to their use given by the objects.

Hanabi (Bauza 2010) has been selected since it is a card game. Hanabi is also appealing for our research because it also does not give a direct indication about play rules. There is no text written on cards; just the numbers. Hanabi's cards have different colors with a number written over them. Each card has a picture of fireworks over it. During play, the players hold cards towards other players and it is a cooperative game. The different colors and the picture of fireworks give an indication that the game is about lighting up fireworks. The tokens which are round objects with clocks and bomb printed over them gives an idea of turns or time. The same question has been addressed in the study as to how much information users are able to extract from Hanabi objects.

Student's and Professor's Responses in Play Testing

Hive Pocket

The similar points in participant's responses are shown in Table 1. The different ideas given by university students about game mechanics are shown below:

1. Players should connect stones of their color in a group of three and the opponent must oppose this action by introducing his own stones that are stronger in points.
2. White stones should be placed turn by turn to make a chain of stones and black stones should be used to block the chain and vice versa.
3. Every turn, players ask each other one question. Whoever gives the right answer gets a stone of his color to place them on the table. Whoever has lain more stones wins the game.
4. Some hive stones can bite other hive stones. Players randomly pull out one stone from bag in their turn. The stone is compared with the previously placed stone and decided, will the player be bitten or not. Player who has been bitten lesser number of times, wins the game.

Game mechanics proposed by school students are:

1. Placing stone in pairs and gaining points.
2. Pieces must be connected by color or symbol.

Table 1: Similar Responses from Students and Professors (Hive Pocket)

C-Correct to Game Rules; I - Incorrect to Game Rules)	**G1**-Number of responses from University Students (42)	**G2**-Number of responses from Professors (16)	**G3**-Number of responses from School Students (23)	Significant difference found using χ^2 with Marascuilo
The game is played by two players(C)	39	14	17	NO
Player with white stones plays first (I)	19	11	3	YES, G1-G3, G2-G3
Player with Black stones plays first (I)	1	1	0	NO
One Hive stone should be placed each turn (C)	14	11	5	YES, G1-G2, G2-G3
Each Hive insect gives certain points(I)	8	3	4	NO
Each Hive insect has weakness or strengths (C)	12	8	2	YES, G2-G3
Players should complete the Hive to win (C)	2	2	0	NO
Some insects are important than other insects(C)	2	2	4	NO
Honey Bee is the goal (C)	6	3	0	NO
Winning goal is to leave opponent with no possibility of move (C)	1	0	4	YES, but the expected frequency condition was violated and no significant factors are found with a Marascuilo
Players should place stones matching by color or insect (I)	1	0	13	YES, G1-G2, G2-G3

3. Game is played like Domino.

Some of the game mechanics proposed by professors are described below.

1. The player with white and black stones should try to make a shape of rhombus from their stones and player with most rhombuses wins the game.

2. There are points associated with each insect on the stone. And in each turn one player puts a stone on the table and if the other player can put the stone with the same figure in the next turn; he can capture the stone already placed. The player with maximum stones wins the game.

3. The goal of the game is to capture the Honey Bee of the opponent and there exists a hierarchy from Ant to the Honey Bee; Honey Bee is the strongest in hierarchy.

4. There are certain actions associated with each insect and the goal of the game is to capture the Bee.

5. All stones are placed faced down on the table and players pick one stone randomly in their turn and put it face up on the table. There are rules about connecting stones and if these rule violate than the player has to skip his turn. The winner is decided based upon which player has lain most stones on the table.

Hanabi

The similar points between three groups of participants are described in Table 2.
However, we also received contradictory ideas from the groups such as some university students said that the rules are similar to the games *Uno* or *Durak* and professors have associated rules with *Poker* and *Lino*. Durak is a Russian card game, of two to six players, where the goal is to empty your hand of card by playing the cards to the table in an increasing order. About the game dynamics, We got various rules from the three groups, The various game mechanics suggested by university students are:

1. Players take some cards turn by turn and on each turn, they match the sum of their total cards with the previous player; if the total sum is more than the previous player's sum than one time token is rewarded to the player and the player with most tokens wins the game.

2. One card is put on the table and each player has taken 8 cards randomly. In each turn, players should try to put the same card which is already placed on the table. If they do not have that card, they have to skip their turn. The game continues and each time a card is randomly placed on the center of the table and players try to throw the same card from their hand.

3. The time tokens are given to the player as a punishment, each time they do not have the same card as the one placed on the table.

4. Different color cards have different moves such as red cards moves the clock to the next stage, white color freezes the clock, blue and green color cards help prevent the bomb from exploding. Yellow color card is responsible for giving the fuse token to the player. The multi-color cards give some advantage to the player in the form of an action.

5. Players are randomly distributed time token in each turn and when a bomb explodes, the game stops and sum of cards in each player's hand is calculated to find the winner.

Table 2: Similar Responses from Students and Professors (Hanabi)

C-Correct to Game Rules; I - Incorrect to Game Rules)	**G1**-Number of responses from University Students (30)	**G2**-Number of responses from Professors (13)	**G3**-Number of responses from School Students (25)	Significant difference found using χ^2 with Marascuilo
Random shuffling of cards at start (C)	6	5	0	YES, G1-G3, G2-G3
Deal each player five cards in the beginning/ Game is for 2-5 players (C)	8	3	8	NO
Comparison of cards based on colors or numbers(C/I)	11	7	3	YES, G2-G3
Fireworks light at some point in the game (C)	7	2	6	NO
Player with maximum number of cards in the end wins the game (I)	5	2	0	NO
Players with no cards left win the game (I)	6	4	13	YES, G1-G3
Making a sequence of cards from 1 to 5 leads to winning (C)	2	1	0	NO

6. Each player should try to make their own firework by collecting as many cards of each color as possible.
7. The rules are the same as *Durak* or *Uno*(Robbins 1971).

The school students described game as *Chinese Poker* or *Uno*. And the main idea proposed by them was regarding the winning condition, which is getting rid of cards in hand.
From professors group, game mechanics proposed are described below:

1. It is a fire works competition and player with more variety of colors wins.
2. Players are distributed equal number of cards and play starts by placing cards on the table and getting time tokens in each turn based upon some rule for each card. The goal is to reach a score of 12'o clock by adding number on the card placed and time on the time token. The number on the card is considered as minutes that can be added to the time on the time token to see if the sum reaches 12'o clock or not.
3. The idea of the game is to light up a firework. Players should try to make a fire work by putting three cards of the same number. Each time a firework is lit by a player the time token is placed in the middle and time goes till it reaches 11:45. The winner is the one who lights a firework at 11:45.
4. Players should try to get rid of cards dealt to them as soon as possible by placing a card on the table that is higher in value then the card already placed.
5. The rules are the same as *Poker* or *Lino*(van den Bulk 2010).

Qualitative Analysis of Participant's Responses

While analyzing the answers from the participants, we got some similar as well as different ideas about game mechanics. The similarities among responses suggest that some parts of the game design are intuitive for all groups, however these similar ideas are mostly related to the general theme of the game. The responses mostly differed when participants answered question about game mechanics and rules.
For Hive pocket, majority of the participants from three groups correctly predicted the number of players. Among young adults, six out of forty two participants mentioned that honey bee is the goal. The same answer is given by three professors and among school students, nobody could predict the honey bee as a most important insect instead one of the students mentioned ladybug as the insect to be captured as a goal. An answer given by one of the professors is:

> *I think a player wins when he can place the yellow queen bee piece surrounded by its soldiers.* There should be a rule about which piece can connect to which e.g. spider can connect to any other insects since it can use its web. Mosquitoes can attack the bee pieces when they are not protected by a spider since spider can catch a mosquito. *Intuitively, I think the less number of pieces, the more important. Therefore, Ants and green insects are just like pawns in chess game.* Spiders are more powerful soldiers which can protect the Bee. *I chose the Bee to be the Queen, since there are 3 insects that are only 1 piece and among them, mosquito looks more like an evil soldier who can fly and kill any enemy. And the lady bug is not elegant enough to be a queen.* Ladybugs can attack ants, I guess.

The emphasized points show that the professor has carefully paid attention to each component of the game and there's a reasoning involved for each rule decided.
On the other hand, an answer given by one of the university students is:

1. *Insects which can fly can move over other insects more than the insects which cannot fly.*

2. *Spider and mosquito cannot be connected as spider can eat a mosquito.*

3. *Different insects have different actions, purple and red looks more aggressive so purple can block opponents insect for one turn and red can block for two turns. Gray and Brown are also aggressive but can be used as guards. Green and blue look more peaceful.*

4. *Winning condition is to build a hive.*

The answer reflects that student has tried to associate meaning with hive objects based on their appearance but does not elaborate as to why beetle and lady bug look aggressive and why grasshoppper and soldier ant appear peaceful to them.

The responses from both group of participants reveal that logical thinking while extracting game rules ease the process of discovering associations between game objects. By logic we mean: One being able to define a *conceptual model* based upon the given *signifier*, and *perceived affordance* of the game object.

Such logical arguments came from the set of people who are already used to of thinking hard as they are professors and active researchers in their fields. While students have associated meanings to objects, could not support all of their arguments with a rationale. Although the professor's group has spent significantly less time for playing over the past thirty days as compared to student's groups, the effect of age and experience is apparent in the playtest displaying better performance. Hanabi proved to be much tougher for participants as it is a collaborative game and does not follow the traditional approach of card games in which cards are hidden from opponents. The basic theme of lighting up fireworks is visual as all cards have picture of fireworks and all cards comprises of different colors used in fireworks. The both students groups and professors came up with variety of rules for Hanabi. They have tried to use all components of the game in the rules. We have made the comparison based on which group could extract the idea of fireworks. The results does not show a significant difference as seven out of thirty university students, six out of twenty five school students and two out of thirteen professors mentioned fireworks theme in their rules.

The study showed that professors group, which was older than both students groups and have more years spent in education sector displayed logics for their rules. However on the other hand, university students who are younger and possess much less experience in educational sector did not display an effort to link game components logically. School students explained game mechanics by comparing the games with the games they have already played before. They compared *Hive pocket* with *Domino* and *Hanabi* with the game *UNO*.

Considering affordance theory, we see a mixture of Gibson and Norman's concepts behind student's responses. As an example, when participants made the point regarding, *white player goes first*, the number is very low in case of school students and only three among twenty three students mentioned this because they were not familiar with *chess*. When professors and university students made this point, they referred to *Chess*. Participants preferred following preconceived notions over Gibson's theory when they could propose an action that is possible but is not usual based upon their prior experience with similar objects. This depicts a trend that, it is less likely for the participants to propose an object interaction which they have not experienced before though the object design clearly support this interaction. In *Hanabi*, only one participant who was a university student mentioned that it might be a cooperative game, all other participants considered *Hanabi* similar to either *Uno* or *Poker*. In this sense, it is important to pay more attention towards signifiers in game design if the game mechanics does not follow usual schemes.

Observational Analysis

During user testing, professors applied a systematic approach: (1) placing game objects in front of them and counting all objects, (2) separation of objects with colors or numbers, (3) writing down all information about numbers and types of objects, (4) taking time to analyze all information about game objects and finding associations among them, and (5) explaining different possibilities about game mechanics and finally deriving conclusions. From both student's groups observation, we found that (1) students did not take notes about number and types of game objects, (2) majority of students were easily bored during the user testing and wanted to finish as fast as possible, and (3) students quickly gave up on the problem.

The observational process reveals that professors group were more keen to view the games as a problem to be solved. After user testing, we explained the rules of the game to the participants. All participants showed an incredible amount of excitement after hearing the actual game rules for Hanabi. The participants stated that Hanabi is very different from other card games since it does not allow a player to see their own cards. When told of the correct rules of the game, users were relieved and many expressed joy at there being an explanation to their frustration. One user went so far as to buy five copies of Hanabi not long after the test as he was so intrigued by the idea of players holding the cards away from themselves. Emotions, in this case *relief*, that users felt, allowed them to excuse some of the design decisions, see other examples in (Norman 2004).

Our qualitative analysis of results and observational study of the user testing process highlights three points:

1. Games, if difficult to figure out quickly bore young players earlier than the older players.

2. Violation of preconceived notions of game play leads to excitement due to novelty.

3. Violations of preconceived notions require signifiers in the game design to draw participant towards the intended action naturally, in other case, the game understanding highly depends on the rule book.

Investigating the importance of human factors gives an insight into as to what affects player's ability to extract meaning from game objects. However, a concrete conclusion on the significance of age and education factor for the ability to observe intuitiveness in game objects requires more experiments with different groups of people and with more games.

Quantitative Analysis of Participant's Responses

In order to determine the significance of the responses quantitatively, a number of common responses were summed into binomial categorical data; the user wrote this statement or did not write the statement. A hypothesis test was used to find the the difference on the categorical characteristic proportion between the three groups known as a χ^2 test which examines the difference in frequency was used at the $\alpha = 0.05$ significance level. The null Hypothesis is that there are no difference between the three groups. In the event that we reject the null hypothesis in a test, a Marascuilo procedure was applied at the $\alpha = 0.05$ significance level.
The tables 1 and 2, respectively show the response of Hive and Hanabi, if the response was correct or incorrect to the actual rules of the game, the counts of the response, and the results of the statistical test as described above including which groups are found to be different.

Hive

Hive shows that school students were less likely to incorrectly state that white always moves first. This perhaps has to do with a lack of the perceptual model of chess. Both older students and Professors were more likely to state that each hive stone is placed one at a time. The belief in stones having different powers also was found more easily by the professors than school students. Further, school students were far more likely to see this as a game of matching. The results seem to imply that the conceptual hook for older participants was to see this as a game like chess, with various characters having movements, white moving first. The school students came with a conceptual model of *Hive* being a matching game, perhaps using the bag as though it was Dominoes.

Hanabi

Hanabi demonstrates a number of differences in the thinking in regards to the use of the cards. First the school students did not ask for the deck to be shuffled and highlighted the use of the comparison of cards based on color or number. They were also more likely to state that the objective of the game was to remove cards from a hand.

Implications of Research

The research methodology applied enables games developers to identify user's perceptions of game objects to make a comparison between real affordances and perceived affordances of these objects. Norman defines perceived affordances as how an object may be interacted with based on perception (Norman 2013). The playtesting methodology, involving the removal of rules, provides the absolute conditions for a player to make actions according to their perceptions. A comparison between the player's perceived affordances and real affordances of game objects allows the analysis of game objects designs to see which objects have led the player towards the correct mapping of affordances and which objects persuaded the player towards an incorrect or might be an opposite mapping of game mechanics.
The playtesting environment provided in the research for the game is not that of actual in competition game play. An individual is examining the game as if they first opened up the box and initially attempts to understand what is in the box. This allows for greater freedom in the player's mind as other players are not enforcing a set of known rules. As Gibson defines affordances to be, all action possibilities latent in the environment irrespective of an individual's ability to recognize them(Gibson 2014), In a non competitive environment, playtesters are naturally propelled to try out various actions with objects that could make them recognize affordances which were not observable by their perception initially.
A game design from Gibson's point of view of affordances is significant to understand all possible actions an object offers. At the same time, the removal of rule book has brought a change to the Game's environment. This helps to measure the change of object based on the existence and non-existence of rules. Moreover, the study of game design from Norman's point of view of affordances is also crucial to understand how well the game design supports its play mechanics.
Our adopted methodology for playtesting investigates game designs from both Gibson's and Norman's approach towards affordances. Knowledge about player's perceived affordances helps to measure the impact of human factors on players understanding about game actions that can affect their level of enjoyment with the game. Hence, games designers can account for the extent, as well as, the impact of human factors to achieve

the intended purposes of the game. Moreover, as players are propelled to try out atypical actions in a non- competitive testing environment with the absence of a guidance manual or a rule book, this methodology also gives knowledge about all possible affordances with objects and allows for the improvisation of new games designs.

Conclusion

The motivation for this study is to investigate the impact of human factors on how players perceive games. It is obvious that not all games are intended for a specific category of users. Games have a wide audience which varies in many respects such as age, education, occupation, culture, personality, etc. Game design should satisfy the whole audience in terms of intuitiveness and entertainment. This idea requires a clear understanding of the involvement of human factors on the level of intuitiveness a player finds in a game design. User testing was applied to three categories of users; school students, university students and professors, to see the impact of age and educational years on their ability to explore game play. This reveals how intuitive these games are for the different class of the participants.

Our study results show that professors could derive via logic the associations among game objects that helped them in figuring out game play. While students did not support their arguments with logic, still they could observe the intuitiveness in game objects. University students inferred the basic theme of the game merely by looking into game objects but they were unable to unfold the actual game mechanics. The school students compared the tested games with the games they have already played and proposed game mechanics close to previously seen games. The majority of professor's responses correctly explained the game rules along with the ending goal while inferring the purpose and associations among game objects. However, the study size must be increased in order to validate the hypothesis that there is a clear distinction in professor's performing better because of their logical thinking, in the task statistically. Future studies will expand upon the size of the testing groups in order to further extend upon this hypothesis.

The study results highlight that the participants have followed preconceived notions rather than trying out a new action possibility. A design that is aligned with a player's preconceived notions does not require mental effort to discover usability. However, it is worth noting that after a game, which violated a players preconceived notion was explained, the participants were excited towards using the design.

For future work, it is significant to consider each human factor separately to measure its impact accurately. As an example, We have seen that professors are more aligned toward logical thinking, but it is difficult to estimate which human factor among age or education is deriving them toward logical thinking. Our study has not investigated the impact of cultural background upon what could be considered intuitive as this has a very clear cultural lens. The student's groups consisted of all Russian participants and professors group have various nationalities. To prove or rule out cultural impact, the study must be done with a user group considering culture variable only. Moreover, gender is another variable that we will take into account to investigate its influence upon a player's perception and intuitiveness about games.

REFERENCES

Bauza A., 2010. *Hanabi.* R&R Games Inc.

Browne C., 2015. *Embed the Rules. Game & Puzzle Design*, 1, no. 1, 60–70.

Cardona-Rivera R.E. and Young R.M., 2013. *A Cognitivist Theory of Affordances for Games.* In *DiGRA Conference.*

Daviau R., 2011. *Design Intuitively.* In *The Kobold Guide to Board game design*, Open Design, Kirkland, Washington. 42–49.

Entertainment Standards Association, 2014. *2014 Essential Facts about the Computer and Video Games Industry.* Tech. rep.

Entertainment Standards Association, 2016. *2016 Essential Facts about the Computer and Video Games Industry.* Tech. rep.

Entertainment Standards Association, 2017. *2017 Essential Facts about the Computer and Video Games Industry.* Tech. rep.

Gibson J.J., 2014. *The ecological approach to visual perception: classic edition.* Psychology Press.

Norman D., 2013. *The Design of Everyday Things.* Basic Books, expanded ed. ed.

Norman D.A., 2004. *Emotional Design: Why We Love (or Hate) Everyday Things.* Basic Books.

Robbins M., 1971. *UNO.* AMIGO Spiel + Freizeit GmLH.

Story M.F.; Mueller J.L.; and Mace R.L., 1998. *The universal design file: Designing for people of all ages and abilities.*

Tekofsky S.; Spronck P.; Goudbeek M.; Plaat A.; and van den Herik J., 2015. *Past Our Prime: A Study of Age and Play Style Development in Battlefield 3. IEEE Transactions on Computational Intelligence and AI in Games*, 7, no. 3, 292–303. ISSN 1943-068X. doi:10.1109/TCIAIG.2015.2393433.

van den Bulk C., 2010. *Lino.* Giuoco.

Yermolaieva S. and Brown J.A., 2017. *Dice Design Deserves Discourse. Journal of Game & Puzzle Design*, 3, 64–70.

Yianni J., 2001. *Hive Pocket.* Gen42.

PLAYING STYLES

Dynamically Extracting Play Style in Educational Games

Reza Khoshkangini
Annapaola Marconi
Fondazione Bruno Kessler (FBK), Trento, Italy
E-mail: khoshkangini,marconi@fbk.eu

Santiago Ontañón
Jichen Zhu
Drexel University, Philadelphia, USA
E-mail: so367,jichen@drexel.edu

KEYWORDS
Player modeling, machine learning, game AI

ABSTRACT

Player modeling is a key aspect for game personalization. However, most existing works consider play style as a static characteristic of the player. This is far from true, since players often change their play style, especially in long gameplay sessions. To overcome this limitation, we aim at dynamically extracting the play style and use it in customizing the game *on-the-fly* to each individual player during the game. To this end, we have designed a framework that automatically recognizes players' play styles that can equip the system with the ability to adapt the game session by session. Hereby we evaluate our proposed framework in the context of an educational game called *Solving the Incognitum.*

Introduction

Player modeling has been widely used in the domain of electronic games (Georgios N., Pieter, Daniele, & Elisabeth, 2013), educational games (Göbel, Wendel, Ritter, Steinmetz, et al., 2010; Sawyer, Smith, Rowe, Azevedo, & Lester, 2017), digital and smart entertainments (Zong, Kveton, Berkovsky, Ashkan, & Wen, 2017; Mahlmann, Drachen, Togelius, Canossa, & Yannakakis, 2010; Hashemi & Kamps, 2017) to minimize players' frustration and increase players satisfaction and engagement (Riedl, Stern, Dini, & Alderman, 2008) in the game. Most of the studies involving adaptive games based on player modeling assume that the play style of players is a fixed property which does not change over time as they play. However, as we showed in our previous work (Valls-Vargas, Ontanón, & Zhu, 2015), this is far from true, and players, in fact, often switch play styles within a single session of play.

In this paper, we build upon our previous work on identifying the dynamic play style exhibited by players in educational games. The key idea of the proposed approach is to automate the process of extracting the play style of the player in each stage of the game to enable dynamic adaptation of the game to each personal player over time. In addition, the framework exploits a type of *Scoring* function capable of uncovering the whole play styles that the particular player behaved during the game. The proposed approach can be divided into five main modules: *Data Collection*, *Data Preparation*, *Play Style Recognition*, *Play Style Prediction* and *Game Adaptation.* The main focus of this paper, however, are the play style *Recognition* and Prediction modules.

We claim that accurately recognizing and predicting the play style of players can significantly impact the way adaptive games are built, by choosing or even constructing new game content to fit the current player's play style. Game adaptation could be implemented for various aspects of game elements such as visual appearance, levels, challenges, difficulty and even story-lines (Bjork & Holopainen, 2004) by using systematic techniques like procedural content generation (PCG) (Shaker, Togelius, & Nelson, 2016). The proposed approach has been evaluated on data captured from an educational interactive game-based learning called *Solving the Incognitum* (Valls-Vargas et al., 2015), which aimed to teach the relationship between fossils and geological time record taken in the historic Charles W. Peale's Museum of Art in Philadelphia.

The rest of the paper is organized as follows: We briefly report some basic notions of play style and player modeling in Background. In the next section we introduce our proposed framework that followed by experimental results. Finally, the paper closes with discussion and conclusions.

Background

Our work relates to the concept of player modeling and the methodology to capture and model the players' play styles, in the domain of video games –in general– and *Educational Game-Based Learning* in particular, with the goal of achieving personalized experiences.

The concept of *player modeling* refers to the study of computational models that have been widely used in the domain of video games for customizing game content to player's preferences, traits, abilities and personalities (Riedl et al., 2008; Hendrikx, Meijer, Van Der Velden, & Iosup, 2013; Smith, Lewis, Hullet, & Sullivan, 2011). Advancements in player modeling have been driven by the need to increase players' engagement, e.g, in educational games (Valls-Vargas et al., 2015; Paras, 2005).

Play style represents the actual player's substrate and provides an understandable pattern to the system to be used for adapting the game based upon the player's behavior (Van Der Werf, Uiterwijk, Postma, & Van Den Herik, 2002), may therefore hold value to augment the players' engagement for a long term, or recommending a new personalized product to the player (Skocir, Marusic, Marusic, & Petric, 2012).
In this work, we build upon existing work in play style taxonomies (Ferdig, 2008; R. A. Bartle, 2004; R. Bartle, 1996; Heeter & Winn, 2008) to define 4 main play styles in *Educational Game-Based Learning*:

- *Achievers (or goal seeker)* are problem solvers and willing to play as fast as possible with a minimum number of errors. They usually focus on game content that is necessary to complete and win the game as soon as possible.
- *Explorers* analyze and investigate all the game elements. E.g., they might visit all the game items with a significant amount of attention, and also make few mistakes in the game.
- *Careless (or uninterested)* play quickly and make many mistakes.
- *Lost players* play slowly and make many errors.

These last two play styles usually do not pay any attention to the game and make little effort to play well.
Many approaches have been focused on analyzing player behavior throughout the game to obtain and characterize the play style of the player. Examples are the work of Canossa (Canossa, 2013), Hsieh and Sun (Hsieh & Sun, 2008), or Weber et al. (Weber, Mateas, & Jhala, 2011). Within these studies, some constructed the play style in the context of a game, based on feature selection and segmentation. For instance, similar work to our proposed framework has been introduced by Drachen et al. (Drachen, Canossa, & Yannakakis, 2009) such that a large-scale empirical experiment on modeling the play style conducted in which an unsupervised learning algorithm via Emergent Self-Organizing Maps (ESOMs) (Thurau, Bauckhage, & Sagerer, 2003) is applied to identify play styles for the popular game title *Tomb Rider: Underworld (TRU)*.
Segmenting the whole game period into different time-intervals (a.k.a. *time windows*) is another strategy which was employed by Bifet et al. in (Bifet & Gavalda, 2007) and also Martin et al. in (Rajchl et al., 2016), such that the player's behavior is captured in each time-interval to characterize the play style and used to predict the style for the next time-intervals.
Having in mind Bifel and Martin's strategies to capture the play style, in this study we proposed a module that automates the construction of player modeling in educational game-based learning and applied it in both on time scale and segmentation. The module is

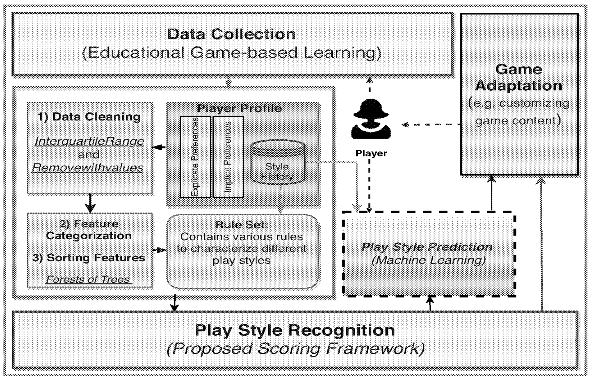


Figure 1: The Conceptual View of the Module Integrated to the Main Framework.

equipped by five main components, in which a scoring system is designed using a utility function to construct the play style of the player. The proposed scoring system is generic in the sense that it enables the framework to implement and score infinity play styles (as long as their traits are defined) in various context.

Technical Approach

In this section, we describe the proposed play style identification approach that is constructed based on a *Utility Function*. The proposed approach exploits a type of Scoring Function so that at the core of the framework there is an agent that iterates through the features to identify the play style of players by constructing a vector of scores (for each player and each game section). This vector is obtained by applying the utility function on the sorted collection of features gathered from gameplay sessions. Sections can be manually or automatically determined by the game designer or based on the structure of the game respectively (e.g., completing a level in the game, or achieving a certain amount of points in the game).

Hereby, we describe the module and its five main components responsible for play style recognition: *Data Collection*, responsible for collecting data while players are playing the game; *Data Preparation*, where data is polished from noise and outliers; *Play Style Recognition*, which groups players based on their play style; *Play Style Prediction*, which predicts the play style of the player at hand given the groups identified in the previous step; and finally *Game Adaptation*, which is responsible for customizing the game content to an individual player's characteristics. These components are shown in Figure 1, however, the last component is out of the scope of this study and we focus on just the first four.

Figure 2: The Screen-shot of the Game and its Environment.

Data Collection

This component is in charge of collecting data from players who participate in the game. In general, it logs players' raw data in their profiles that indicate the actual player's activity and participation during the game. From domain to domain, various kinds of techniques (e.g., monitoring, questioners), equipments (such as smart devices, mobile, tablet), interfaces (including web and mobile applications, etc.) could be used to capture the data. The data that we have used in this study to validate the proposed play style recognition module was collected from *Solving the Incognitum* (Valls-Vargas et al., 2015), which is a game-based interactive learning environment. A screenshot of the game and its environment is shown in Figure 2. We have postponed the detailed description of the game and the experimental set-up to Section .

Data Preparation

The purpose of this component is two-fold: cleaning the data from noise and outliers, and discretizing continuous features into categorical values. Accordingly, the component contains two main sub-components of *Data Cleaning* and *Data Categorization.*

Data Cleaning: Cleaning data from noise and outliers is critically important for two reasons. First, noise can have negative influence on data categorization, when we proceed to discretize the continuous values into discrete labels. Secondly, noise can also be the source of errors in extracting and predicting play styles, and consequently game adaptation. We use domain knowledge to select the suitable set of features among the available features that were recorded to characterize the meaningful play style. Thereafter, WEKA[1] libraries are used to clean the data from outliers and extreme values. To this end, we execute *InterquartileRange* function (coded in Python) to recognize the outliers. Then, we use *Removewithvalues* to remove those outliers from the data set. These result in a data set free from noises that skew feature categorization. In Evaluation section we detailed how many samples are excluded from the data set due to containing extremely noisy data.

Feature Categorization: Converting continuous data to categorical data has been widely studied in the literature of data analysis and machine learning (Kaufman & Rousseeuw, 2009), and can bring advantages and disadvantages in various cases (Tan et al., 2006). For the specific case of play style prediction, our experiments show the admissible improvement with respect to directly using the raw continuous data (Valls-Vargas et al., 2015)). In our experiments, we have discretized each continuous feature into three categories (*Low*, *Medium*, and *High*). To perform this discretization, three intervals are defined by finding four points, where the first and last correspond to the minimum and maximum values of the feature, and the two middle points are calculated as follows:

$$Second\ Point = (mean(data) + minimum)/2$$
$$Third\ Point = (mean(data) + maximum)/2$$

where *data* refers to the list of data points that players have done in a particular feature (e.g., "*item visited*" that refers to the number of items that a player visited in a specific section or time-window of the game.). Then, the three intervals are defined as: *Low* that indicates the region between the first point and the second point; *Medium* including the interval between the second point and the third interval; *High* that includes the range of players whose activities are between the third and the forth points.

Finally, the list of the categorized features, which are sorted in importance using Scikit-learn[2] package, will be sent to the next component to be considered in the recognition process.

Since every feature has a different weight in characterizing a play style, we used *Forests of trees* function to recognize this importance. Hence, we exploit the ground-truth data set which are labelized by the expert researchers (detailed in Section) to specify the weight of each feature w.r.t the target values (play styles). Although, the framework is applied on each individual player's data to extract the play style, this categorization and extraction is related to the other players' activity.

[1]We took the advantage of *InterquartileRange* and *Removewithvalues* functions made available in WEKA (Hall et al., 2009).

[2]To sort the list, we exploit the *forests of trees* function which is available in Scikit-learn (Pedregosa et al., 2011) in Python.

Play Style Recognition

This component concerns the extraction of the play style according to the players' behaviors, which are yielded by monitoring their activities throughout the game and recorded in the form of a vector of features (described in Section).

As we have shown in (Valls-Vargas et al., 2015), play style of the players do not fall into a particular play style and they act differently through the different game sections. E.g., a player who is flagged as an Achiever in one section could be changed to an Explorer in the next game section.

Thus, in this study we automate the procedure of recognizing the dynamicity of play style introduced in (Valls-Vargas et al., 2015) by designing a Scoring framework that is able to automatically recognize this dynamicity with its confidence value, and uncover all play styles in accordance to which the particular player exhibited during the same game section.

To this end, a *Utility Function* is designed and used in the core of the framework to score the play style that she has behaved in each session of the game in a supervised way. Since different features have different impact on characterization of a player's play style, the function utilizes weight and order of computation for each individual feature to calculate the score of each style for a given player. The utility function is defined in the below equation:

$$\varsigma_j^s = \sum_{i=1}^{n} x * wf_i^{s^j} \ni x = \begin{cases} 0 & \text{if } (f_i^{s^j} = \text{“M”}) \\ 1 & \text{if } (f_i^{s^j} = rs_i^{s^j}) \\ -1 & \text{otherwise} \end{cases} \quad (1)$$

where M stands for medium, n is the number of features and wf indicates the weight of the feature for the specific style s^j in the given vector. x refers to the statement that whether the features' values are identical in comparison between player's behavior which is defined by the vector of feature f and the characteristics of a play style defined in the Rule Set rs.

We use “Forests of Trees” (Breiman, 2001) function to order the features (in each vector) in importance on an artificial classification task. This function enables the estimation of feature importance on a specific model (in our case study the play style e.g., Achiever, Explorer). Consequently, this importance value is used as a weight in the utility function to calculate the score of each play style.

In addition, the framework is equipped with a Confidence Calculator Component (CCC) to measure the confidence value of the retrieved styles for each player. The component uses a type of *Harmonic Series* equation (illustrated in Equation 2) to obtain the confidence value of the play styles, where the calculated score “ς^s” is divided by the sum of all weights of the sorted features (in the list) that can be possible to fulfill the requirements of a pure play style.

$$Conf^{s_j} = \varsigma^{s_j} * \frac{1}{\sum_{i=1}^{n} w_i} \quad (2)$$

Thus, given the vector of features that reflects the player's demeanor in a game session, the algorithm uses an agent to iterate trough the features and compare the values (these values are described later on in the Evaluation section) against the rule set that defines characteristics of all play styles in a supervised manner. If any feature from the vector is identical to any play style's characteristic defined in the rule, the algorithm adds a score for the particular style. The score will be penalized with weight and order of the feature, where the player has behaved exactly the opposite compared to the specified style in the Rule Set. Since there may be a feature with a *medium* value in a vector, the algorithm considers the feature as the neutral feature toward the specific style in the scoring procedure.

Due to the similarity among players' play style characteristics e.g., Careless and Achiever players usually play quickly, or Explorer and Lost players play slowly, it is not possible that a player acts only in a specific style in a particular session. Thus, this scoring system can bring out the styles with their confidence, which enables the framework to recognize which player plays in margin, and which one trends toward a specific style with high confidence. Consequently, to assign a style to a player, the algorithm can pick up the style that has a high score and confidence values.

Play Style Prediction

This component concerns the prediction of the players' play styles for the next session of the game to generate and personalize game content. The action flow is detailed in Figure 1, where players' past behaviors are fed to the prediction component to train the model exploiting machine learning classification algorithms. Thus, given the model, the module by feeding the new data of the player is able to predict the play style for the next section or the time-windows of the game. This prediction allows the system to dynamically adapt the game content to the individual player, by taking into account players' past behavior in different sections of the game. A full description of the exploited classification methods can be found in the following sections.

Evaluation and Results

Solving the Incognitum is a first-person, point-and-click 3D interactive e-learning environment that was developed to teach the relationship between fossil and geological time zone. The game environment is built based on the largest museum of natural history in the early 19th century known as *Charles W. Peale's Museum of Art*[3], which is located in Philadelphia city (USA). During the game, players can interact with museum's exhibits such as minerals, strata deposits, fossils and portraits of renowned historical figures related to the exhibits. In addition, players are able to collect the jewelry of the museum by correctly answering the questions they face in various sections of the game. Questions are mainly

[3] `https://www.philamuseum.org`

associated to fossils, minerals or other information related to geological time, location, etc.
The game consists of 6 sections such that players after passing a short section as "*Tutorial*" are able to know what the game is and how to play it. They then, can explore the virtual space in the main four sections of the game called "*Quests*" to assess various exhibits, as well as answering multiple questions related to them to reach and complete the objective of the game. Finally, *Exploration* section, where players are allowed to continue the exploration of the game.
In this study there were 75 freshmen students (in Digital Media department) who volunteered to participate to the game to study the correlation between player type and their influence on learning in Solving the Incognitive. In this experiment, volunteers were not told that they must complete the game, hence they could have quit the game whenever they prefer or get bored of playing the game. Before starting the game, participants were asked to complete a pre-knowledge test consisting 12 questions about the essential earth science context (to validate and grantee the reliability of the survey questions Cronbach's alpha (Santos, 1999) was used). They were also given up to 1 hour with an instruction to play the game.
We implemented the proposed module on the data-set including 75 participants gameplay's data that was used in (Valls-Vargas et al., 2015), in which a player modeling framework was constructed based on episodic segmentation of gameplay traces and sequential machine learning. Basically, researchers represented how the players change their play styles during the game. In that study two researchers by recording (screen recording) the participants' activities tried to label their play styles by looking their behavior and activity regarding to visited items, navigation in the game, use of the concept of map, questionnaires, etc. As the inter-rate reliability, if the two researchers obtain similar play style for a player, the label is selected with high confidence, otherwise, they discussed to convince each other. Hence, Explorer, Achiever, Lost and Careless types of play styles are assigned to those players.
Borrowing the strategies "Time-based windows and Segmentation" that was used in (Valls-Vargas et al., 2015), we evaluate our proposed automatic play style recognition on players' gameplay data that was recorded from their behavior in the game. Three players are excluded via *InterquartileRange* and *Removewithvalues* functions explained in the above section, because of having extreme and outlier values. In addition to be consistent with the numbers of players in the ground truth data-set in order for validating the proposed approach, we also excluded 17 players' data due to miss-information, monitoring, questionnaire.
Behavior of each individual player is represented by around 70 features in each section of the game which vary widely, from how the player moves in the game to visiting location and items, the number of questions that she answered (including correct and wrong), to the time that she spent to read and answer the questions. This information is captured individually while she plays the game.

Evaluation Objectives

To evaluate the proposed approach we take into account only 4 sections of the game and 2 minutes time-windows. To assess the performance of our proposed framework, we have looked at two complementary aspects, which are reflected in the following research questions:

- RQ1) To what extent the play styles that are automatically extracted for a player are similar to the labels selected and administered by means of research members for the same player in the same section of the game? and with how much confidence players played toward the recognized play styles?
- RQ2) To what extent can we predict the play style considering the previous styles?

We applied the proposed framework on the four main Quests (sections) and considered two minutes time-windows. The list of features which are important to characterize the play styles of interest (see Section) were selected among the 70 features that are logged in the data-set using the domain knowledge; to characterize those styles (e.g., survey, questionnaires). Thus, in this experiment we utilize only the extracted features from the data-set without demanding any question from players to model their behavior.
Three main categories of features were extracted such as *Time spent* in the game, *Number of visited game content*, and the game content that reflects the *player's feedback* from the game, for a total of 11 features for each player. Features were selected in basis of usability for characterizing players' pattern in the game. In other words, the listed features deliver a map that the proposed system exploits to extract the play styles following the defined Rule Set. Given the above rationale, the selected features are depicted in Table 1.

Table 1: Selected Features

Category	Features	Description
Time Spent	Total Time for Reading	Shows the total time that the player spent to read the content of the game. The average time taken to read the game content during the game was recorded around 2.42 sec.
	Navigation Time	Points to the total time that the player spent navigating around the game.
	Time for Map	Talks about the total time that she spent reading the Map in the game.
	Reading Min	Shows the minimum time that the player spent to read/visit a single component of the game around 0.670 sec.
	Reading Max	Shows maximum time that the player spent to read/visit a single component of the game. The maximum time record during the game is 3.77 sec.
Visiting game content	Total Item Visited	Depicts the total number of items that the player visited.
	Item Visited New	Points to the number of new item visited in a section.
	Item Revisited	Illustrates the total number of component that she re-visited in the game.
	Questions Visited	Talks about the total number of visited questions.
Player's feedback	Questions Right Ratio	Refects the percentage of correct answers that the player provided.
	Questions Wrong Ratio	Shows the percentage of incorrect answer that the player responded.

Rule Set: We take into account the features relating to Time and Visiting the game component to interpret how *fast* and *slow* players play in the game, and consider the feedback of the player to how *precise* they play in the game to characterize play styles. The above features are considered as the *Rule set* (ground truth) to characterize the style described in background Section.
The propose module is used to construct Table 2 (*columns B*) that shows the ratio of players' play styles and how their play styles changed over time in various sections of the game, according to the collected data. As we have shown in this

study (Valls-Vargas et al., 2015), play style is not constant and players change their play styles over the course of the game (see Table 2 *columns A*). Dynamicity of play styles were also obtained by implementing the proposed automatic approach on the collected data, however different results in ratio of play styles are evident. E.g., in section 2, there was a 36% of *Explorers*, whereas in section 3 there was around 20% of *Explorers*, and the number of players classified as *Careless* changed from 16% in section 1 to around 13% in section 2.

Results

Research Question 1

To address RQ1, we have implemented two comparison tests. First, in a "*Peer to Peer*" comparison, we compare the results that was reported in (Valls-Vargas et al., 2015) (for simplicity, we call Group A from now on) against the achieved results in this study (for simplicity, we call Group B from now on). Secondly, we conduct the same A/B test between Group A against Group B by eliminating the players who played in Margin (played ambiguosly). In this experiment, the players whose confidence values are less than X% are considered and highlighted as Margin players (see Figure 3).

Table 2: Proportion Comparison

	section 1		section 2		section 3		section 4	
	A	**B**	**A**	**B**	**A**	**B**	**A**	**B**
Explorer	50%	35%	25%	36%	29%	20%	13%	13%
Careless	25%	16%	18%	13%	20%	29%	16%	20%
Achiever	17%	27%	40%	27%	38%	38%	63%	51%
Lost	~8%	22%	17%	24%	13%	13%	~8%	16%
SqDif	Δ =0.377		Δ = 0.036		Δ = 0.0162		Δ = 0.425	

SqDif: Square Difference.

In the first experiment, we have conducted an A/B test between the labels that the proposed module assigned to players vs. the ones obtained and reported in (Valls-Vargas et al., 2015) that are shown in Table 3. Then, we compared *peer-to-peer* play style of the players that are assigned in each section of the game. In other words, the play style of every single player in Group A is compared to the same player's play style in Group B. We named this test as *Absolute Comparison*. As it is detailed in Table 3, overall 60% of the labels are identical between Group A and Group B, that demonstrates a moderate labelization accuracy in Group B using the proposed framework. The resulting low accuracy is not surprising: surveys results, administered to players after the game session, were exploited in the manual labeling of play styles as additional information and turned out to be essential for the labeling process in all cases of ambiguous play styles.

In the second comparison experiment, we took into account the confidence values calculated by Equation 2 for each player in all sections of the game. Hence, we excluded the players who have less than 10% confidence value from Group B and mutually from Group A. Then, we carried out an A/B test on the two groups. Apart from the comparison test that provides a value (a percentage of agreement) that could be used to validate the proposed module compared to manually extraction of play styles, these confidence values reveal with how much certainty a player behaves toward a specific style. Having the above statement, thus the system is able to highlight players who played in margin by adjusting a threshold value to meet (e.g., Conf=10%). Although, the numbers in absolute comparison shows a moderate similarity between the results, by excluding those who played in margin from Group A and B, the accuracy of correctly labelization increased up to 70%. Table 3 indicates the percentages of play styles, which are identified with confidence and without confidence value in Group B.

Table 3: The Ratio of Players Play Style With and Without Confidence Value in All Sections

		Absolute Comparison			With Confidence value: >10%		
	Play Styles	# of Correctly Labeled (per type and total)		# of Players	# of Correctly Labeled		# of Players
Section 1	Explorer	18	33	55	18	33	48
	Achiever	7			7		
	Careless	7			7		
	Other	1			1		
Section 2	Explorer	9	29	55	5	21	39
	Achiever	8			13		
	Careless	6			1		
	Other	6			2		
Section 3	Explorer	10	36	55	10	35	48
	Achiever	16			15		
	Careless	8			8		
	Other	2			2		
Section 4	Explorer	4	32	55	1	22	26
	Achiever	23			20		
	Careless	4			1		
	Other	1			0		
Total # of Players		130		220	111		163
Overall Accuracy		60%			~70%		
Test		X^2=9.46, P = 0.05			X^2 = 2.61, P = 0.05		

We can also look at the significance of the correctly labeled play styles between the groups (with confidence value, for simplicity we call Group C as for now, and Group B that defined above) vs. the ground truth (Group A). To that end, we applied the *Chi Square*[4] test on these groups. Once between Group B vs. Group A and once between Group C and Group A. The score $X^2 = 9.46$ shows the value of x^2 statistic is higher than the critical value that is 7.81 for a 95% confident value level. This indicates that labels in Group B need a considerable improvement which did not well-fit w.r.t the ground truth. While the value of $X^2 = 2.61$ for the second comparison (between Group C and A) is less than the critical value "7.815" with a 95% confidence. This shows that when outlier samples are drawn from the populations, the improvement in correctly play style labelization is statistically significant.

The plot in Figure 3. presents the distribution of the extracted play styles with their confidence values in the four sections of the game. y axis indicates the confidence values in ranges $[-\infty, +50]$ in which the negative values refer to the fact that recognized styles are far from the characteristics of the styles compared to the *Rule Set*. While, the more positive value shows players are close to the style that are labeled. Thus, we reached to this statement that the more close the confidence value to zero, the more that player plays ambiguously.

Looking at the confidence values of all possible play styles that every individual player recorded in each section of the game, it is undoubtedly evident that players relatively trended toward behaving in a particular style in each game

[4]We took the advantage of *chisq.test()* function from *FunChisq* package in r for the statistical test.

section. E.g., player "Bob" behaved ~30% as Careless, and ~20% as Explorer in section one, and played 20% as Achiever, and 35% as Explorer in the next section of the game. Thus, the module picked up and assigned the play style that has higher confidence value (Careless with ~30% against Explorer by ~20% in the first section).
This relatively behaving in the game brings out a critical challenge in adapting game content that highlights the need for exploiting machine learning to predict the next chain of changes in players play style.

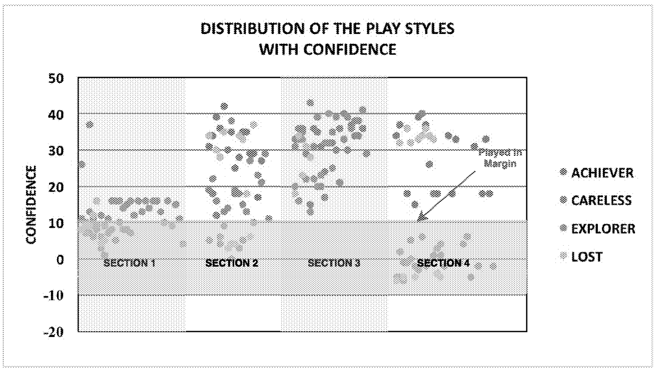


Figure 3: Confidence Value of the Extracted Play Styles.

The above proportion therefore provide no support for a statistical difference for assigned play style of the players groups of this magnitude. We can thus answer RQ1 by: *There is no significant difference in proposed automatic extraction play styles compared to the ground truth that administered by expert researchers, when margin players are excluded.*

Research Question 2

To evaluate RQ2, in this experiment we take into account only the styles which we used in absolute A/B test comparison (*one single style*) without considering their confidence values. Thus, we have constructed three different experiments as follows:

- *Section by Section (Individually)*; in this experiment, we aim to predict the play style of individual player in each section of the game "*Separately*".
- *Cumulative*; in this evaluation, to classify play style for a given player, we consider all the player's data from all sections prior to the one at hand.
- *Time Window*; to conduct this experiment we ignore sections, and just consider the data collected from the past 2 minutes whenever we want to classify play style.

For each experiment, we constructed different Training and Test sets which are detailed in its section, and in all experiments we try to predict the play style that already labeled in previous sections or time-windows.

Section by Section (Individually):

To evaluate this experiment, we took into account the data of each player separately (player's performances only in that section) and evaluated modeling and prediction pipelines. Due to the fact that the number of samples for each player is quite limited, we also took into account the data of the first section of the game (Tutorial). Hence, to construct the Training and Test set with maximum 5 instances for every player, the iterative leave-one-out cross validation is used to evaluate the classification problem. Each round, one sample is left out to be considered as the *Test set* and the remaining samples are used to learn the model. Finally, the average of all players' classification performances are calculated and reported as the result of the classification approach. In other words, the comprehensive performance evaluation of each classification methods in all players are integrated as follows; the sum of True Positive predictions of all players as TP, the sum of False Positive predictions of all players as FP, the sum of True Negative predictions of all players as TN, and lastly the sum of False Negative predictions of all players as FN, which are depicted in Figure 4.

To train the model, we employed 5 different ML classification techniques such as "Naive Bayes, Decision Tree, Bayesian Network (BN), Logistic Regression (LOG REG) and One-R" and compared vs. each other to find out which method can provide the best result w.r.t our data. Once the model is trained, the Test-set is used to validate the classification method. Thereafter, well-known metrics such as Precision, Recall and F-measure are used to accredit the experiment.

Table 4: Classification Performances-Individually

Metrics	Recall		Precision		F-measure		AUC	
Methods	Ind	CuM	Ind	CuM	Ind	CuM	Ind	CuM
Naive Bayes	57%	81%	52%	74%	53%	76%	58%	90%
Decision Tree	41%	61%	27%	46%	33%	53%	-	-
BN	58%	83%	53%	77%	54%	79%	61%	90%
LOG REG	55%	78%	46%	68%	50%	72%	51%	39%
One-R	50%	70%	43%	61%	44%	65%	50%	54%

Ind: Individual result.
CuM: Cumulative result.

The classification result of 5 approaches are depicted in Table 4, where *Bayesian Network* with 58% recall, 53% precision and F-measure 54% and Naive Bayes by 57% recall, 52% precision and 53% F-measure (almost similar efficiency) accomplished better among the other approaches, however these numbers are still considered as a poor result.In addition, we quantify the difference between the methods by the *Area Under the Curve (AUC)* (Jurasinski, Koebsch, & Hagemann, 2012) metric that represents a probability that the classifier will randomly chosen positive observation higher than randomly chosen negative observation. Hence, looking at AUC value of Bayesian Network and Naive Bayes, Bayesian approach performed marginally better with AUC of 61% vs. Naive Bayes with 58%. Following the above two models, Decision Tree, Regression Logistic and One-R classification approaches also obtained a very poor results as shown in Table 4.

These low classification results on the players' individual recorded data is not surprising, due to the lack of well-enough instances in Training set to properly model the individual behavior. Another reason for this low performance could be the change of play styles sourced from players behavior during the game.

Cumulative Experiment:
To evaluate this experiment, we took into account the same sections used in the Individual experiment. Thus, the cumulative data-set of each personal player contains five samples that logged in her profile/history as follows; *section 0-0; section 0-1; section 0-2; section 0-3* and *section 0-4*, which each section indicates player's behavior up to that section. E.g., *section 0-0* represents the behavior that a player (Alice) has done in section 0, *section 0-1* refers to the activity that Alice behaved in section 0 and section 1, and so on. The conceptual view of this cumulative experiment is depicted in Figure 4.

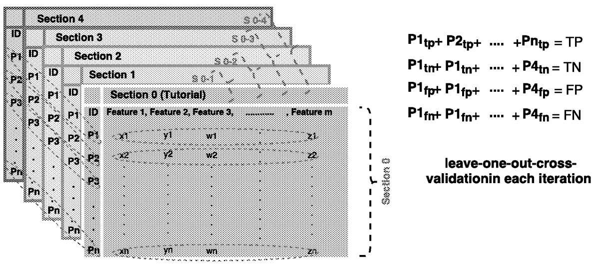


Figure 4: The Structure of Training and Test Set in Individually and Cumulative Experiment.

We performed leave-one-out cross-validation on players' cumulative data using Training and Test set and implemented the above 5 classification approaches. Similar to the *Individual Experiment*, the sum of True Positive, False Positive, False Negative and True Negative off all cumulative sections are integrated to calculate the comprehensive performance of the classification method. Table 4 represents recall, precision, F-measure and the area under the curve (AUC) values obtained through cross-validation on the Training set using the validation set (denoted as Test set).
In this experiment a significant improvement in classification results is observed. E.g., Bayesian Network provided a remarkable improvement with recall 83%, precision 77% and ~80% in F-measure, which compared to the individual experiment around 20% is improved in all metrics. A similar betterment is also achieved by Naive Bayes approach with 81% in recall, 74% precision and F-measure with 76%. Although, the numbers in Logistic Regression and One-R showed a significant improvement compared to the achievement in Individual experiment, both approaches are still poor in AUC value with 39% and 54% respectively. This significant improvement may be due to that the cumulative experiment could exploit data richer, which properly characterize players' play styles. The results presented in the above three experiments highlight that *Bayesian Network* performs better w.r.t our small data-set of samples. Hence, in the next experiment we determine to implement Bayesian Network to model players' behavior with the time frame.

2-minutes Time Windows:
In this analysis we ignored the 4 main sections of the game, and segment the whole game into equal 2-minutes time windows (see Figure 5). Thus, to classify play style we just consider the data collected from the past 2 minutes. As it is

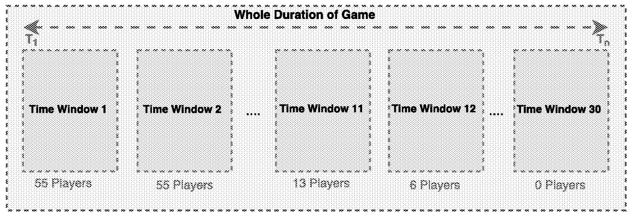


Figure 5: Conceptual view of 2 Minutes Windows Segmentation.

mentioned in Section , each player had been allocated for 60 minutes to play the game. But many players quit the game before completing the whole game (for any reason), so we are limited to only 12 2-minutes windows. We then exclude players who have less then 6 time windows (basically players who played minimum 12 minutes in the game). Hence, leave-one-out cross-validation is used to construct Training and Test sets on each player's time windows.
Similar to Section by Section (individually) experiment the outcome of test samples for each player (we refer to TP, FP, TN and FN) are integrated together to compute the overall result. Naive Bayes, Decision Tree, Bayesian Network, Logistic Regression and One-R classification approaches are implemented on each time window. Indeed, in this experiment we intend to represent the performance of the prediction while players change their play styles in different time windows.

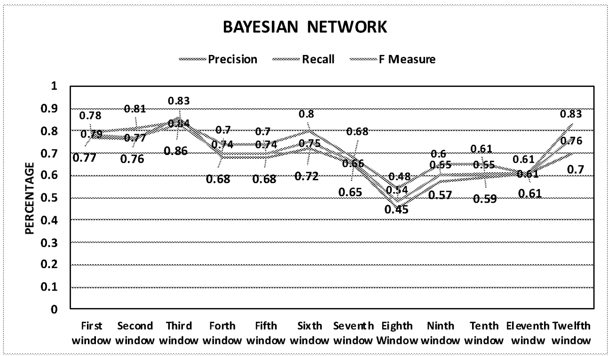


Figure 6: Classification Result for 2-minutes Windows.

Results, illustrated in Figure 6, show the flow of changes in classification performance through the 12 time windows.
We can observe reliable performances till window 6, with ranges of 78% to 80% for recall, 79% to 75% for precision, 77% to 72% for F measure. While, the classification performance smoothly started to decline from overall 72% (F-measure) in window 6 to overall 61% (f-measure) in window 11, in which window 8 recorded the worst numbers by 54% recall, 45% precision and 48% F-measure. This decrement of performance might be due to the less number of samples (maximum 6 time windows) that those 13 players recorded in their profiles.
Classification validity measures achieved in window 12 (which is the last time window) show an admissible performance with 76% in all metrics. Since, sample size is a significantly important feature in any empirical study (spe-

cially modeling in machine learning), this result in that last time window indicates, considering more time windows to learn and model the styles, led to obtain a good prediction. Taking into account the above three evaluations, we can thus answer RQ2 by: *The proposed prediction approach proved to be able to predict play styles with good performances, particularly when cumulative strategy is exploited. However, lower performances have been obtained whenever dealing with poor or limited data. Moreover, Section by Section and Time Window strategies, present issues when dealing with players behaving relatively towards specific play styles.*

Discussion and Conclusion

In this paper, we have provided a recognition framework that automates the procedure of extracting play style of players in educational games. The proposed recognition approach is constructed exploiting a *Utility Function* that is able to extract play styles by scoring players' behavior related to a specific style in the game.

The automatic play style recognition framework presented in this study is capable of modeling players' relatively behavior in each stage of the game that allows the system to tailor the game w.r.t the recognized style. Evaluation results from the data which was captured from an educational game called *Solving the Incognitum* (Valls-Vargas et al., 2015), has been presented. The aim of the evaluation was to assess whether the proposed automatic framework is able to reliably extract and predict the play styles of player during different sections of the game. Prediction results could be used to advance game customization and thus improving players' engagement in the game.

We have evaluated the play style extraction system through an A/B test on the data collected from an interactive educational game-based environment. The main objective of this preliminary work and experiment was to compare the play styles that the automatic framework assigned to players with the play styles that were manually determined (for the same player) by expert researchers reported in (Valls-Vargas et al., 2015). Thus, we evaluated the framework based on the following key aspects:

- Comparison between the play styles that are extracted and assigned for the same player using the two methods.
- To what extend can we predict the play style considering the previous styles?

Taking into account the first aspect (RQ1), the evaluation result indicated that the play styles assigned by the proposed automatic framework and the play styles that assigned to the same players by the expert researchers are very similar (70% similarity), when players who played ambiguosly are removed from the whole players population (57 out of 220 samples).

The results reported for RQ2 specified that cumulative individual player's data is preferred in considering the model player's behavior by achieving around 80% accuracy in the test set vs. Individual Experiment and Time Windows with overall 60% accuracy in the test set.

The small number of participants might be sourced such a poor classification performance (Individual and Time Windows experiment). Moreover, the issue could be intensified by noisy data that players recorded during the game. This highlights the need of larger population to learn and model players' play styles in the game. The clear constraint of the proposed play style recognition is that the framework is a rule based system that limits the system to extract play styles, which are out of the Rule Set.

An interesting extension of the proposed framework can improve the Rule Set. Instead of relying on the Rule Set that we manually constructed, an unsupervised machine learning algorithm (Association Rule Learning) could be implemented on the data to recognize the relations between the features to discover a map (chain of changes) for each individual player's behavior. This would mean that we can extract the rules directly from the recorded data by a Data Mining process that would require a larger number of samples. This also enables the prediction module to effectively contribute to game adaptation by generating personalized content for the next section of the game.

As part of our future-work we plan to implement the proposed module in Urban Mobility gamification (Khoshkangini, Valetto, & Marconi, 2017) with a larger number of participants to extract and model players' behavior.

References

Bartle, R. (1996). Hearts, clubs, diamonds, spades: Players who suit muds. *Journal of MUD research*, *1*(1), 19.

Bartle, R. A. (2004). *Designing virtual worlds.* New Riders.

Bifet, A., & Gavalda, R. (2007). Learning from time-changing data with adaptive windowing. In *Proceedings of the 2007 siam international conference on data mining* (pp. 443–448).

Bjork, S., & Holopainen, J. (2004). Patterns in game design (game development series).

Breiman, L. (2001, Oct 01). Random forests. *Machine Learning*, *45*(1), 5–32.

Canossa, A. (2013). Meaning in gameplay: Filtering variables, defining metrics, extracting features and creating models for gameplay analysis. In M. Seif El-Nasr, A. Drachen, & A. Canossa (Eds.), *Game analytics: Maximizing the value of player data* (pp. 255–283). London: Springer London.

Drachen, A., Canossa, A., & Yannakakis, G. N. (2009). Player modeling using self-organization in tomb raider: Underworld. In *Computational intelligence and games, 2009. cig 2009. ieee symposium on* (pp. 1–8).

Ferdig, R. E. (2008). *Handbook of research on effective electronic gaming in education.* IGI Global.

Georgios N., Y., Pieter, S., Daniele, L., & Elisabeth, A. (2013). Player Modeling. In *Artificial and computational intelligence in games* (Vol. 6, pp. 45–59). Dagstuhl, Germany: Schloss Dagstuhl–Leibniz-Zentrum fuer Informatik.

Göbel, S., Wendel, V., Ritter, C., Steinmetz, R., et al. (2010). Personalized, adaptive digital educational games using narrative game-based learning objects. In *Edutainment* (pp. 438–445).

Hall, M., Frank, E., Holmes, G., Pfahringer, B., Reutemann, P., & Witten, I. H. (2009, November). The weka data mining software: An update. *SIGKDD Explor. Newsl.*, *11*(1), 10–18.

Hashemi, S. H., & Kamps, J. (2017). Where to go next?: Exploiting behavioral user models in smart environments. In *Proceedings of the 25th conference on user modeling, adaptation and personalization* (pp. 50–58). New York, NY, USA: ACM.

Heeter, C., & Winn, B. (2008). Implications of gender, player type and learning strategies for the design of games for learning. *Beyond Barbie to Mortal Combat: New perspectives on games, gender, and computing*.

Hendrikx, M., Meijer, S., Van Der Velden, J., & Iosup, A. (2013). Procedural content generation for games: A survey. *ACM Transactions on Multimedia Computing, Communications, and Applications (TOMM)*, *9*(1), 1.

Hsieh, J.-L., & Sun, C.-T. (2008). Building a player strategy model by analyzing replays of real-time strategy games. In *Neural networks, 2008. ijcnn 2008.(ieee world congress on computational intelligence). ieee international joint conference on* (pp. 3106–3111).

Jurasinski, G., Koebsch, F., & Hagemann, U. (2012). *Flux rate calculation from dynamic closed chamber measurements, r package version 0.2-1.*

Kaufman, L., & Rousseeuw, P. J. (2009). *Finding groups in data: an introduction to cluster analysis* (Vol. 344). John Wiley & Sons.

Khoshkangini, R., Valetto, G., & Marconi, A. (2017). Generating personalized challenges to enhance the persuasive power of gamification..

Mahlmann, T., Drachen, A., Togelius, J., Canossa, A., & Yannakakis, G. N. (2010). Predicting player behavior in tomb raider: Underworld. In *Computational intelligence and games (cig), 2010 ieee symposium on* (pp. 178–185).

Paras, B. (2005). Game, motivation, and effective learning: An integrated model for educational game design.

Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., ... Duchesnay, E. (2011). Scikit-learn: Machine learning in python. *Journal of Machine Learning Research*, *12*, 2825–2830.

Rajchl, M., Baxter, J. S., McLeod, A. J., Yuan, J., Qiu, W., Peters, T. M., & Khan, A. R. (2016). Hierarchical max-flow segmentation framework for multi-atlas segmentation with kohonen self-organizing map based gaussian mixture modeling. *Medical image analysis*, *27*, 45–56.

Riedl, M. O., Stern, A., Dini, D., & Alderman, J. (2008). Dynamic experience management in virtual worlds for entertainment, education, and training. *International Transactions on Systems Science and Applications, Special Issue on Agent Based Systems for Human Learning*, *4*(2), 23–42.

Santos, J. R. A. (1999). Cronbachs alpha: A tool for assessing the reliability of scales. *Journal of extension*, *37*(2), 1–5.

Sawyer, R., Smith, A., Rowe, J., Azevedo, R., & Lester, J. (2017). Enhancing student models in game-based learning with facial expression recognition. In *Proceedings of the 25th conference on user modeling, adaptation and personalization* (pp. 192–201). New York, NY, USA: ACM.

Shaker, N., Togelius, J., & Nelson, M. J. (2016). *Procedural content generation in games: A textbook and an overview of current research.* Springer.

Skocir, P., Marusic, L., Marusic, M., & Petric, A. (2012). The mars–a multi-agent recommendation system for games on mobile phones. *Agent and Multi-Agent Systems. Technologies and Applications*, 104–113.

Smith, A. M., Lewis, C., Hullet, K., & Sullivan, A. (2011). An inclusive view of player modeling. In *Proceedings of the 6th international conference on foundations of digital games* (pp. 301–303). New York, NY, USA: ACM.

Tan, P.-N., et al. (2006). *Introduction to data mining.* Pearson Education India.

Thurau, C., Bauckhage, C., & Sagerer, G. (2003). Combining self organizing maps and multilayer perceptrons to learn bot-behaviour for a commercial game. In *Game-on* (p. 119).

Valls-Vargas, J., Ontanón, S., & Zhu, J. (2015). Exploring player trace segmentation for dynamic play style prediction. In *Proceedings of the eleventh aaai conference on artificial intelligence and interactive digital entertainment* (pp. 93–99).

Van Der Werf, E., Uiterwijk, J. W., Postma, E., & Van Den Herik, J. (2002). Local move prediction in go. In *International conference on computers and games* (pp. 393–412).

Weber, B. G., Mateas, M., & Jhala, A. (2011). Using data mining to model player experience. In *Fdg workshop on evaluating player experience in games.*

Zong, S., Kveton, B., Berkovsky, S., Ashkan, A., & Wen, Z. (2017). Get to the bottom: Causal analysis for user modeling. In *Proceedings of the 25th conference on user modeling, adaptation and personalization* (pp. 256–264). New York, NY, USA: ACM.

PLAYING STYLES IN STARCRAFT

Yaser Norouzzadeh Ravari
Department of CS and AI,
Tilburg University,
P.o. box 90153, 5000 le,
Tilburg, the Netherlands
Y.NorouzzadehRavari@uvt.nl

Sander Bakkes
Department of CS and AI,
Tilburg University,
P.o. box 90153, 5000 le,
Tilburg, the Netherlands
S.c.j.bakkes@uvt.nl

Pieter Spronck
Department of CS and AI,
Tilburg University,
P.o. box 90153, 5000 le,
Tilburg, the Netherlands
p.spronck@uvt.nl

KEYWORDS
Play Style, Clustering, Player Behavior

ABSTRACT

Understanding playing styles in video games may assist game designers to create entertaining game content for different players. Numerous factors determine how distinct players may approach a game, e.g., player preference, game literacy, and player motivation. In Real-Time Strategy (RTS) games, for understanding player behaviour, it is particularly important to model player preferences and adopted game strategies. As such, as a continuation of previous work, the present paper investigates how distinct human players approach the popular *StarCraft* game in terms of preferences and strategies that may be inferred from game observations. In particular, we investigate how distinct match-types relate to the different playable races in the game. To this end, we propose features that reflect playing style, and uncover unique variations in playing style by means of Principal Component Analysis (PCA). Findings of experiments with clustering player styles of *StarCraft* players reveal that playing styles can indeed be distinguished in different match types. While one may expect playing style to affect the chance of winning, results reveal win probability is not significantly affected by player style, but the length of matches is.

INTRODUCTION

Gaining an understanding the variations of the playing styles of players of video games, may assist game designers to customize the game content based on the behaviors of their player base. For instance, some studies explored players' ingame behavior (Bateman et al. 2011) to determine how a game can entertain different player types. Previous research investigated the development of a general model that fits a variety of games (Bateman 2009; Yee 2006), or a model applicable to specific games (Drachen et al. 2009; Gow et al. 2012). In our study, we explore playing styles in the popular Real-Time Strategy (RTS) game *StarCraft*.

In RTS games, players generally have plenty of commands at their disposal, that can be enacted at every iteration of the game. In *StarCraft*, players can build various buildings and create different units, while units can perform different actions. A high variety of playing styles is possible by choosing the type and order of buildings, by preferring certain units over others, and by using units in different ways. Therefore, players may play the same game in different ways, which we here informally refer to as different playing styles.

StarCraft includes different race types: Terran (T), Protoss (P), and Zerg (Z). At the start of the game, players may choose one of the races to play with. Races have different unit types and command types. We expect that playing styles are related to the race used by a player, and probably also to the race used by the opponent. For human vs. human matches, six match types are possible: (i) Protoss vs. Terran (PvT), (ii) Protoss vs. Zerg (PvZ), (iii) Terran vs. Zerg (TvZ), (iv) Protoss vs. Protoss (PvP), (v) Zerg vs. Zerg (ZvZ), and (vi) Terran vs. Terran (TvT).

In previous work (Norouzzadeh Ravari et al. 2016), we investigated winner prediction in *StarCraft*. We showed that the top-10 features used for winner prediction are more or less the same across all match types, but there are some notable exceptions, which, as we concluded, depend on the match type of the game.

In this paper, we will further analyze the playing styles in *StarCraft* in relation to match types. We will discuss a new feature set that we propose to distinguish playing styles. The feature set reflects what kind of commands the players use during a match. In brief, in this paper we will answer the following three questions:

1. Are there distinctly different playing styles in *StarCraft*?
1. Do playing styles differ across the match types?
2. Do playing styles differ across the races?

Next, we will discuss literature that is relevant to the topic of the present paper.

RELATED WORK

Different players employ different playing styles. To distinguish different players, several attempts have been made to define playing styles. Researchers looked at players' in-game behavior and players'personalities. Bartle (Bartle 1996) proposed one of the first divisions of playing styles. According to his study, playing style has two dimensions: action vs. interaction and player-orientation vs. world-orientation. Later another study (Yee 2006) reported that Bartle types are not a general prototype, and that they suffer from biases.

The connection between playing style and in-game behavior was made by multiple researchers. Others (Drachen et al. 2009) modeled players' behavior in *Tomb Raider Underworld* and they observed four playing styles.

Reseachers (Gow, et al. 2012) also studied playing styles in *Snakeotron* (an arcade game with a highly limited state space) and *Rogue Trooper* (a third-person shooter game). Playing styles is also studied in *Battlefield 3* (Normoyle and Jensen 2015), and authors found that a player can have multiple playing styles simultaneously.

Few studies have been done on playing styles in RTS games. Bakkes (Bakkes et al. 2009) adopted a case-based approach to model opponent players based on behavioural-similarity metrics. Si (Si et al. 2016) conducted a study on map exploration style in *StarCraft*.

In our previous research (Norouzzadeh Ravari et al. 2016), we found that a general model can predict the winner across all of the match types in *StarCraft*. As such, in the present paper, we propose different feature sets based on match types that exceed 300 features for each player. This feature set is subsequently reduced and analyzed by employing Principal Component Analysis (PCA). We will build on the top Principal Components (PCs) to show how playing styles vary across the match types. Next, we will employ *k*-means clustering for grouping together playing styles accross match types.

Table 1: Specification of the *StarCraft* Dataset

	PvT	PvZ	TvZ	PvP	ZvZ	TvT
Players	4032	1680	1624	748	398	790
Features	511	479	533	371	380	416

DATA

We use a *StarCraft* dataset of expert players (Robertson and Watson 2014); it included approximately 4,000 full replays that the most of players have played only in a single match. An overview of the number of players available in each of the match types is provided in Table 1.

For the purpose of the present analysis, features are defined per race type. Since possible actions vary based on race type, the number of proposed features vary based on the race type. Table 1 shows, besides the number of players, also the number of features per match type. The features encompass the type, frequency, distance to the base, and the number of units that are involved in an action that the player used. We chose these features to discover the role of action attributes in addition to the action type. Given a large number of possible features to investigate, we employ a subset of the features for our analyses. Specifically, for each action, the following features are extracted:

- *Action frequency*: for each player, how many times an action is repeated during the match up.
- *Group size*: how many units the player associated with an action.
- *Group size variance*: to represent the variety of size of groups that are used in the style of player.
- *The number of unique groups*: players can use a group many times, or they create different groups for different tasks.
- *Mean and variance of distance between base and target*: the location of units that perform an action gives useful information. For each action, we computed mean and variance of normalized distance between the primary base and group target.

ANALYSIS

In this section, first we describe feature dimension reduction and feature analysis by PCA. Subsequently, the results of clustering playing styles are given, and the relationships between playing styles, wins/losses and game-length are presented.

Analysis of playing styles using PCA

Our analysis of playing styles builds upon PCA; it is a statistical procedure that is widely used for dimension reduction and for discovering discriminative features (Van Der Maaten et al. 2009). We analyzed our features by PCA to discover playing styles across the match types. We limited ourselves to the top-2 components, and we keep the PCA coefficients above 0.1 to focus on the strongest features.

The result of PCA analysis reveals that top-2 coefficients cover between 37% to 45% of the variance of the features in all of the match types. As such, top PCs can be considered the most discriminative features for distinguishing playing styles. In the following sections, we discuss in detail the most discriminative features for non-symmetric match types (different races playing against each other) as well as for symmetric match types (races playing against themselves).

Analysis of non-symmetric match types

We examined the top PCs for non-symmetric match types (PvT, PvZ, and TvZ) to find the most discriminative features that distinguish playing styles. We found that for the match types PvT and TvZ, the top PCs cover 23% of the variance of features for each of the races involved. For the TvZ match type, it covers 29% of the variance of features, for both of the races involved. For all match types, the second PCs cover between 14% and 17%, while all other PCs score considerably lower. Therefore, it makes sense to focus first and foremost on the top two components, as they cover the major playing styles.

We note that the top PCs in PvT include `research' and `upgrade' commands, while these commands are missing in the top PCs in PvZ and TvZ match types, which seems to indicate that when Zerg players are involved in a match, research and upgrades have little influence on play style. Moreover, we noted that the `train' command is found in the top PCs in all non-symmetric match types. Interestingly, the first PC includes some commands that are limited to a specific race, such as `siege', `unsiege', and `lift' (for Terran) in PvT and TvZ; and `burrow', `unburrow', and `morph' (for Zerg) in TvZ and PvZ. This demonstrates clearly that playing styles depend partially on the race type.

Analysis by race type

We observed that playing styles notably vary in non-symmetric match types. In this subsection, we figure out whether playing styles in a match type differs based on the opponent's race type. As such, we cluster playing styles of each race type in a match type. For instance, in PvT match type, we repeat the earlier described clustering procedure to

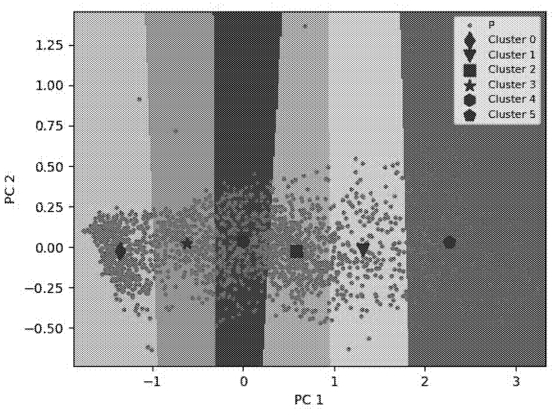


Figure 1: Clustering of Playing Styles of Protoss Players in PvT Matches

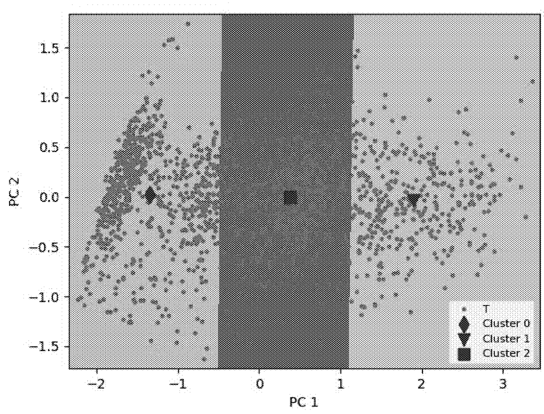


Figure 2: Clustering of Playing Styles of Terran Players in PvT Matches

discover to what extent playing styles within the Protoss and the Terran race are different.

Figures 1 and 2 show clusters of Protoss and Terran players in PvT matches. Protoss playing styles are clustered into 6 clusters, and Terran playing styles are clustered into 3 clusters. This observation shows that the variety of playing styles of the Protoss race is larger than that of the Terran race in PvT matches.

Figures 3 and 4 show the comparison of discriminative features in these races. Each feature label is replaced with the equivalent command code; Table 2 shows the complete list of command codes and feature labels. In Figure 3, we observe that the mean value of some features is close to zero because these commands are limited to Terran race, such as command codes 22, 23, 24, 132, and 139. Interestingly, cluster 2 has the lowest mean of values among all of the features, but at the same time, the difference between mean values of features in the other clusters is more or less the same across all of the features. The highest value of the features belongs to the commands 4, 7, 12, and 13, respectively. In Figure 4, cluster 1 has the lowest mean of feature values. For Terran players, the commands 4, 7, and 12 also have high values, just as the Protoss. Generally, we observe that some commands are used in both races with the similar frequency.

Clustering playing styles

When designers wish to tailor the game experience to a group of players -- or have the game design be informed by observing distinct behaviors -- it helps if they can determine discrete groups of players, each group fitting a particular playing style. To automatically determine such groups for *StarCraft* players, we emploeyd *k*-means clustering procedure. The features listed in the Data section used for clustering after normalization.

To get an impression of the required number of clusters, we utilized the Calinski-Harabasz (CH) criterion (Caliński and Harabasz 1974); it is a common clustering optimization criterion that has been used successfully for cluster analysis (Maulik and Bandyopadhyay 2002). Building upon this method, we varied the number of clusters from 2 to 14 and examined the CH index values. We observed that the CH value has a peak at 4 clusters.

The extracted feature set is high-dimensional. We therefore employed PCA for dimension reduction. We found that PCA with two components covers more than 37% of the variance of features. Therefore, we used the top two PCs in k-means from the scikit-learn package in Python for clustering.

In non-symmetric match types, we examine playing styles by two approaches: principal component analysis of playing styles without considering the race type of the opponent (opponent-independent) and principal component analysis of playing styles by considering the race type of the opponent (opponent-dependent).

Playing styles in non-symmetric match types are presented for PvT matches in Figure 5. The top two PCs for each of the match types determine the axes. We observe that the players from different race types are generally placed in different clusters. For instance, in Figure 5 Protoss players are mostly located in clusters 0 and 1, while Terran players belong to clusters 2 and 3. Therefore, we conclude that the playing styles between each of two races is different. Moreover, the results suggest that also within race types there are different play styles possible. A comparison between dispersions in different match types shows that dispersion in PvT is lowest, which means that in a PvT match similarity in playing styles within a race is highest.

We observed that different races have different playing styles. To discover playing styles in a race, firstly we separate players in a match type based on their race type. Then, we utilize PCA to find more informative features for each race. We keep the PCs above 0.1. Next, we select the top-two PC, which together cover more than 39% the variance of the features.

In PvT matche types, the first PC in the both races have some commands in common such as `research', `upgrade', and `train', but there are some differences too. For instance, Terran components include `siege' and `unsiege' (which are Terran-only commands). The Protoss top PCs include `use tech' and `train fighter,' which are not included in Terran components. The top PCs of Protoss and Zerg players in PvZ matches share `research', `upgrade', and `train' commands, but the `use tech' command is only included in the Protoss component. The Zerg top PCs include the Zerg-only `burrow', `unburrow', and `morph' commands.

In TvZ matches, the top PCs of Terran and Zerg players show more dissimilarities than similarities. They only share the `train' command. The top components of Terran include besides the Terran-only `siege' and `unsiege' commands, the

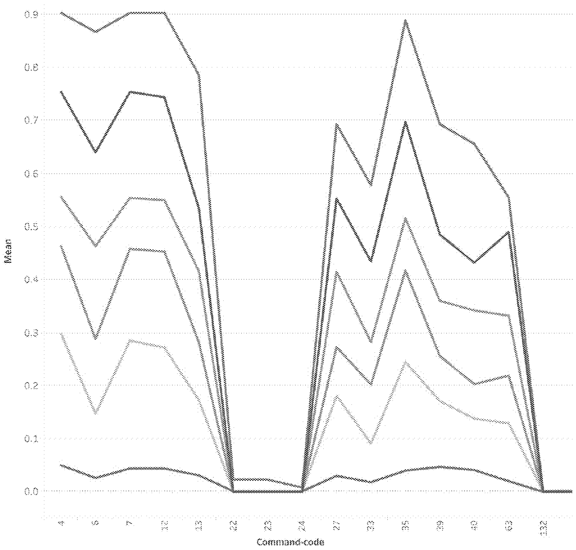


Figure 3: The Comparison of Discriminative Features' Mean of Protoss Clusters in PvT Matches

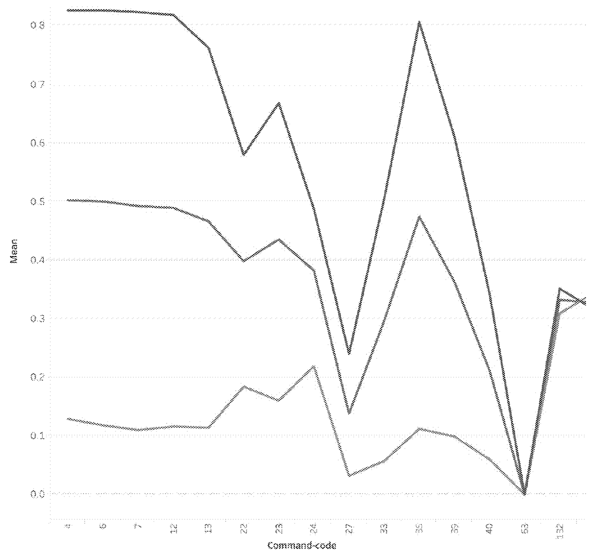


Figure 4: The Comparison of Discriminative Features' Mean of Terran Clusters in PvT Matches

general commands `research', `upgrade', and `use tech.' The Zerg player's top PC, besides `train,' are limited to Zerg-only commands such as `morph', `burrow', and `unburrow'.

Playing styles in symmetric match types

In symmetric match types, the players can build the same buildings and select the same commands. While the players could potentially use exactly the same playing styles, we found that there are still variations of playing styles employed.

For each of the match types, we separated the players according to their race type and performed once more a PCA analysis and *k*-means clustering per race. For the resulting clusters, each of which arguably represents a different style, we calculated the average win-loss ratio and average game length. The results are summarized in Table 3 for non-symmetric match types. We removed the results for symmetric match types due to the space limitation.

In Table 3, playing styles, the win-loss rate, and mean game-length are presented for each cluster in non-symmetric match types. We used the Wilcoxon ranksum test, finding that for each race type the game-length between most of the clusters are significantly different ($p < 0.05$). The clusters of Terran players in PvT have the same properties; that is, while the win-loss rates are similar, the game-lengths vary. Interestingly, the clusters of players of PvZ and TvZ matches also have comparable win-loss ratios, and varying game-lengths. In summary, we may conclude that playing styles affect game-length.

However, the win-loss ratios for none of the match types differed significantly from each other. While one might be tempted to think that Protoss players have a higher chance of winning than Terran players in PvT matches, and that Terran players have a higher chance of winning than Zerg players in TvZ matches, a Wilcoxon rank sum statistical analysis shows that the differences between all ratios are not significant at $p< 0.05$. We found that similar conclusions held for symmetric match types, i.e., game lengths differed significantly in most cases, while win-loss ratios did not.

Table 2: Command Code of Each Feature Label

Code	Label	Code	Label
4	Train	28	Unload All
6	Research	33	Cancel Construction
7	Upgrade	35	Cancel Train
12	Hold Position	39	Rally Point Unit
13	Stop	40	Rally Point Tile
16	Return Cargo	63	Train Fighter
18	Burrow	132	Place Mine
19	Unburrow	139	Cast Scanner Sweep
22	Siege	142	Cast Psionic Storm
23	Unsiege	143	Cast Irradiate
24	Lift	145	Cast Consume
27	Unload	147	Heal Move

CONCLUSION

This paper investigated playing styles of *StarCraft* players, insofar they relate to match types. We found that there are definitely different playing styles available to players, which are partially -- but not completely -- based on the commands that are unique to a race, and the opponent race. Even when a

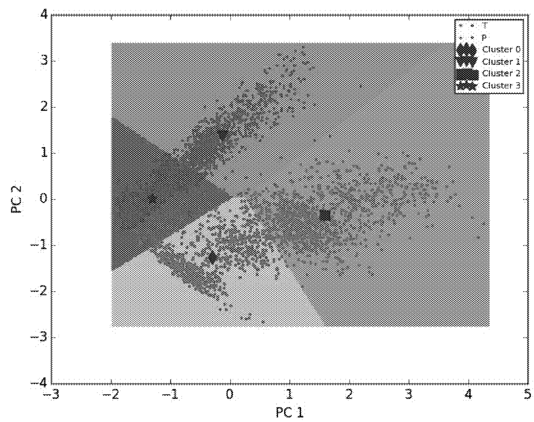


Figure 5: Clustering of Playing Styles in PvT Matches.

race plays against itself, different playing styles are used.
We found that for the expert players, choice of playing style does not influence their win-loss ratio, though it does influence game length.

REFERENCES

Bakkes, S. C., Spronck, P. H., and Van Den Herik, H. J. 2009. “Opponent modelling for case-based adaptive game AI”. 1, pp. 27-37. Elsevier.

Bartle, R. 1996. “Hearts, clubs, diamonds, spades: Players who suit MUDs”., Journal of MUD research 1.1, p. 19.

Bateman, C. 2009. “Beyond game design: Nine steps toward creating better videogames”. Cengage Learning.

Bateman, C., Lowenhaupt, R., and Nacke, L. E. 2011. “Player typology in theory and practice”. Digital Games Research Association.

Caliński, T., and Harabasz, J. 1974. “A dendrite method for cluster analysis”. Communications in Statistics-theory and Methods, 3, 1-27.

Drachen, A., Canossa, A., and Yannakakis, G. N. 2009. “Player modeling using self-organization in Tomb Raider: Underworld”. IEEE symposium on computational intelligence and games, (pp. 1-8).

Gow, J., Baumgarten, R., Cairns, P., Colton, S., and Miller, P. 2012. “Unsupervised modeling of player style with LDA”. 4, pp. 152-166. IEEE.

Maulik, U., and Bandyopadhyay, S. 2002. “Performance evaluation of some clustering algorithms and validity indices”. IEEE Transactions on Pattern Analysis and Machine Intelligence, 24, 1650-1654.

Normoyle, A., and Jensen, S. T. 2015. “Bayesian Clustering of Player Styles for Multiplayer Games”. 11th AAAI Conference on Artificial Intelligence and Interactive Digital Entertainment (AIIDE-15) pg, (pp. 163-169).

Norouzzadeh Ravari, Y., Bakkes, S., and Spronck, P. 2016. “Starcraft Winner Prediction”. 12th Artificial Intelligence and Interactive Digital Entertainment Conference.

Robertson, G., and Watson, I. D. 2014. “An Improved Dataset and Extraction Process for StarCraft AI”. FLAIRS Conference.

Si, C., Pisan, Y., and Tan, C. T. 2016. “Understanding players' map exploration styles”. Australasian Computer Science Week Multiconference, (p. 69).

Van Der Maaten, L., Postma, E., and Van den Herik, J. 2009. “Dimensionality reduction: a comparative”. J Mach Learn Res, 10, 66-71.

Yee, N. 2006. “The demographics, motivations, and derived experiences of users of massively multi-user online graphical environments”. 15, pp. 309-329. MIT Press.

Table 3: Comparison of Win-Loss and Game-Length (mean-gl and std.dev in minutes) in Non-Symmetric Match Types. The Last Column Denotes with which Clusters there are Significant Differences for Game Length ($p < 0.05$))

Matc Type (#players)	Cluster	#players	Win-loss	mean-gl	std.dev	Significantly different from
PvT (4032)	P0	504	1.16	24	9.6	P1, P2, P4
	P1	480	1.35	28	9.5	P0, P3, P4, P5
	P2	228	0.96	29	10.5	P0, P3, P5
	P3	370	1.05	25	10.2	P1, P2, P4
	P4	130	1.28	32	16.6	P1, P3, P5
	P5	304	0.97	23	10.6	P1, P2, P4
	T0	360	0.87	28	10.6	T1, T2
	T1	895	0.88	27	11.8	T0, T2
	T2	761	0.88	24	10.4	T0, T1
PvZ (1680)	P0	158	1.13	28	13.8	P2, P4, P5
	P1	125	0.86	29	10.1	P2, P4, P5
	P2	163	0.66	18	7.6	P3, P4, P5, P6
	P3	85	1.18	29	10.8	P2, P4, P5
	P4	30	1.14	39	18	P0, P1, P2, P3, P5, P6
	P5	121	0.75	21	10.2	P0, P2, P3, P4, P6
	P6	158	1.05	27	12.8	P2, P4, P5
	Z0	470	1.04	21	10.7	Z1
	Z1	370	1.15	31	12.6	Z0
TvZ (1624)	T0	212	1.19	27	9.7	T1, T2
	T1	271	1.13	22	9.2	T0
	T2	329	1.23	22	7.9	T0
	Z0	302	0.97	23	8.2	Z1, Z2
	Z1	370	0.79	21	8.9	Z0, Z2
	Z2	140	0.72	29	8.4	Z0, Z1

PROCEDURAL GAME GENERATION AND MECHANICS

Computational Creativity and Game Design: Towards Procedural Game Generation

Jorge Ruiz Quiñones and Antonio J. Fernández Leiva.
Department Lenguajes y Ciencias de la Computación.
Universidad de Málaga, Andalucía Tech, Málaga, Spain.
E-mail: jorge.ruiz@alu.uma.es, afdez@lcc.uma.es.

KEYWORDS

Computational Creativity, Game Design, Video Game Description Language, Procedural Games Generation.

ABSTRACT

Based on the idea of procedurally generating games, this paper presents an initial study on important topics related to creativity in general and with what is commonly called Computational Creativity in particular. We review some of most important work in different fields such as music, writing or painting. All main aspects related to Game Design tasks also depend on creativity. The paper describes how and when creativity is applied during the different phases and tasks of the Game Design process. Finally, a new approach to build games procedurally is presented using the concept of an extended video game description language and including Artificial Intelligence search techniques for obtaining better variations of a base game according to previously specified game designers' requirements.

I. INTRODUCTION

Exploring the idea of procedural games generation using any kind of procedure encouraged us to think about and review the base processes involved in game development and generation. Setting aside all obvious coding and other computational work, there is an important concept that is present in game development. The concept, considered to be a key factor, as it applies to several phases of game development [1], is that of creativity and this paper looks into the relationship between creativity and game design.

This paper starts by defining creatitivy and how it is related to several different fields where computational processes are applied. Section II discusses Computational Creativity and Section III details the state of the art of Computational Creativity with respect to several different art-related fields. This section also introduces the Computational Creativity applied to videogames and game design. Section IV explains how to address the problem of evaluating games generated procedurally. Finally, Section V explains in detail the theoretical approach for obtaining complete games generated using a computer process.

Creativity in designing processes

Creativity is a term that can be applied to many areas in a general way. Definitions have been given for creativity depending on the specific moment and author and can change depending on the area where the term is applied. In general terms definitions of creativity are directly related to the **process of building** any kind of product and **solving problems**.

Definitions, applied to a general creativity, are focused on linking creativity with some kind of work, produced by an entity. As an example, definitions like "*the tendency to generate or recognize ideas, alternatives, or possibilities that may be useful in solving problems, communicating with others, and entertaining ourselves and others*" [2]. Note the relevant importance of this definition together with the terms *solving problems* and *entertaining*, two concepts strongly related to the game industry. From the point of view of novel products of value and the entity that produces a given work, creativity can also be described as the capacity to produce such work [3].

Other approaches note the importance of having a concrete outcome attached to any work process and relate creativity with the tasks of experimenting and discovering. This is a very interesting definition from our point of view, in the sense that our aim is to try to create and design games procedurally, and so, these two tasks are directly present in our work (creating and generating solutions and finding the optimal for our purposes).

II. COMPUTATIONAL CREATIVITY

Creativity is an extremely abstract concept that has so many definitions and distinct points of view and even different definitions for specific areas, as mentioned. We centre on creativity from the point of view of computational processes, which is commonly called Computational Creativity (CC). The CC (also known as artificial creativity or even creative computing/computation) can be defined as a multidisciplinary activity nestled between several fields such as Artificial Intelligence (AI), Psychology and the Arts. The goal is to model, simulate or replicate human works found in the aforementioned fields using some kind of computer assistance that helps the process of creating any of the concrete targets of those fields. We could say that a new AI area called Creative Reasoning has come into play [4].

The CC is a very promising field in several aspects. It takes advantage of Artificial Intelligence, in itself a very important field of research and also considering that CC can be applied to many different areas with several different targets product/solutions to be generated. So, it is not just about CC as a general term, but CC may have its own subareas of research. Some examples are journals such as the *Journal of Creative Music Systems* (JCMS) or conferences such as the annual *International Conference on Computational Creativity* (ICCC) or the *Conference on Computer Simulation of Musical Creativity* (CCSM).

Nowadays it is easy to imagine computers helping with processes that require creativity in some sense. However it is important to note that this concept has been a key point of study for a long time. For instance, "*The digital computer as a creative medium*" a publication from 1967 [5] stresses the importance of a computer to help authors create music or artistic visual images.

A more recent trend is to consider not only the creative process or product in itself, but also to see CC from other points of view, including of the person and the media within the creative entity it is related to [6]. From here, we will explain how CC is applied to different areas and discuss some of the tools that have been developed to aid in the process of generating creative content.

III. COMPUTATIONAL CREATIVITY IN ARTS

Creativity is present in all kinds of art works made by humans. It could be said that these art works would not exist without creativity. Can a computer emulate humans in this sense, able to create art works belonging to disciplines such as writing or painting? This is not new as we can find several attempts to do so from the 70s and 80s as discussed in Sections III and III.

Since computers have been present in our lives, they have proved to be an important tool for creating works exclusively applicable to humans. In the following paragraphs, we see examples from the art world namely writing, painting and music, but creativity is present in many more subjects such as business, organizations or education, to name a few.

Writing

Tools for generating narrative in many ways is not a modern concept at all. Tale-Spin, a program designed to write stories interactively, is an example from 1977 first published in the 5th International Joint Conference of Artifical Intelligence. Dehn's (1981) AUTHOR and Lebowitz's (1985) UNIVERSE are other examples that clearly show that in those years there was active research in this field.

Tools in this field have improved over time, and there are several examples that automatically generate narrative. Poetry is a very specific kind of writing much explored in this sense. Tools like A.D.A.M. a Computer Poetry Generator or V.E.R.N. Visual Examples of Rhetoric Notations are a couple of implementations. In some cases, the tool is not only concerned with the text itself, but also with the representation of the texts as an art form in themselves.

More recent applications and a current line of research is oriented towards generating papers, even papers intended for publication.

More geared towards the world of research, is an interesting study found in [7], including both perspectives, narrative generation and a visual tool to arrange and visualize narratives into a two-dimensional space based on their structural features. Along the same lines, a visual tool that hierarchically organizes and visualizes generated narratives into a two-dimensional space based on their structural feature values can be found in [8].

This field is generally held to be a vibrant area of research, with many publications and is increasingly giving more importance to the final user and the value that the texts can bestow. Also important are the increasing number of commercial tools being designed to automatically generate content for professional businesses, trying to generate texts with added value for the reader and trying to ensure these texts are accurate, relevant and give valuable information.

Painting

With respect to the visual arts, software for producing both abstract and representational art, has lots of successful examples. The pioneer in this area is a tool called *AARON*. Originally developed in 1973 by Harold Cohen, it is able to generate colour paintings including objects or people in 3D scenes, and has been successful enough to see some of its creations in important art museums like the University Art Gallery at Mandelville Centre [1] in 2017 or Electro Museum in Moscow [2] in 2016.

NEvAr (Neuro Evolutionary Art) [4] is an especially interesting example for us as this tool is based directly on a genetic algorithm that is able to evolve the image genotype to produce the final results. This is important from our point of view because a fitness function is defined to evaluate the results as the generation evolves. This is very promising because most of the tools mentioned here, do not have those kinds of evaluations. Moreover, the fitness function is also fed with human interaction. Each step of the algorithm can be evaluated by a human user and that feedback is used to guide the algorithm in further generations.

It is important to note that, although there are many references in the research world to computational painting, it is not common to find research and tools relating to how to categorize it. In this sense, Painting-91 [9] is one of the few succesful research attempts to categorize paintings in three different ranges (artist, style and saliency detection) and DeepDream [10] also classifies images automatically via convolution and neuronal networks.

Music

Music helped or automatically produced by a computer has existed for years. Due to the advances in computing power, it is possible to apply different techniques and approaches, like, for instance, evolutionary computation, to generate sophisticated computer-based compositions.

CSIRAC (1950s) is recognized as the first computer able to play music [11]. Its purpose was not to generate compositions

[1] http://visarts.ucsd.edu/events/harold-cohen-creating-computational-creativity
[2] http://electromuseum.ru/

but rather to play popular melodies. Another contribution of that era was the *Ferranti Mark 1* computer which is responsible for the one of the first recordings of computational music.

More recently IAMUS [12] has been able to create completely new contemporary classical music developed within the Melomics project and Music Plus One [13] a real-time system that creates accompaniments for orchestra composition, making use of the Machine Learning technique.

Once again, Evolutionary Computation takes the stage. In [14] the authors explain how different approaches of EC and Cellular auotmata can be used successfully to create musical compositions.

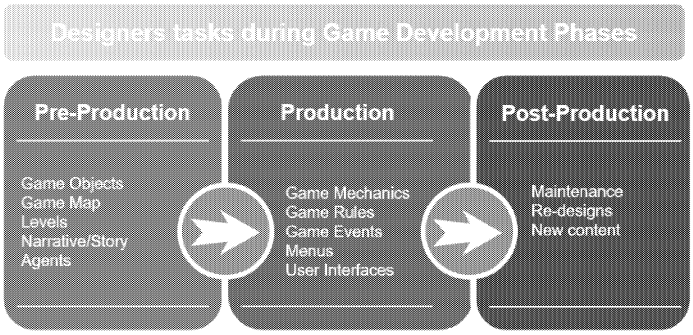


Fig. 1: Game Design related tasks within game development phases.

Games

As seen in Section I, creativity can be applied to processes, whether performed by humans or by artificial intelligence. A quick review of game developmemnt tasks gives us an idea of where we can consider that it is applied directly to game development processes and tasks. We can consider the process of developing a game to be split into three different phases named Pre-production, Production and Post-production. In these phases, well-known and accepted game modelling practices include Game Design tasks of any kind such as game mechanics design, aesthetic design or interaction design [15] [16].

As shown in Figure 1, creativity is present in many different ways during the Pre-production and Production phases. This is especially true for those processes and tasks purely concerned with visual design, like creating graphics for objects, maps, levels and menus. It is easy to think of creativity as being present in tasks involving any kind of content generation or narrative but even the creation of agents may also need creativity in some sense [17] as they need to simulate human behaviors. It is significant that game design takes on an important role in other more *development-oriented* tasks such as defining rules, mechanics, events and conditions. Game designers are involved in defining those aspects of the game, and creativity is a key factor in creating innovative and attractive products as they will affect the gameplay itself and so, in turn, the player experience.

Note that even in the Post-production phase, creativity could be involved in tasks such as promotion and ads, packaging or any redesign task performed during the product maintenance. Here again can be considered visual-related tasks, but in addition are the tasks involved in refining the product, such as defining new or better game mechanics applied during maintenance.

Creativity in game design is therefore a key aspect within the overall process. Is there an opportunity therefore to apply it to all of these processes? This is a key question because it becomes a major requirement when automating the whole process of building a game. It should be feasible to programatically implement all referenced tasks. In a second phase, it must also be possible to apply any kind of algorithm to those processes, combining the idea of CC and evolving or finding solutions within a search space. This will enable the process to produce even better games.

Although the idea expounded in Section V is how all these process could be programatically automated in order to obtain a completely procedurally generated game, game designers are still a key factor and they certainly cannot be replaced. They play a key role in defining a base game, to be used as a seed and may also participate in the process of obtaining and evaluating solutions in a given point of the search. It is also important to consider their feedback in the process to find better solutions according their needs and thoughts.

IV. EVALUATION OF COMPUTATIONAL CREATIONS

There are several papers which attempt to evaluate computational creations. *SPECS* [18], proposes a standard procedure to evaluate computational creations. However automatically evaluating the results of any tool, according to creativity and how good they are, is a very complicated mission. Even if some human help is available, it is quite subjective, because each individual will feel differently about each creation because of the highly subjective nature of the generated products. So, there is a big gap to explore and something that we need to take into account in our study when automatically evaluating our results as the algorithms work.

Some of the tools evaluate the results principally considering the *well form* of the solution, validating the output and checking the completeness. An example of this kind of evaluation is used in IAMUS [12], simply based on the form and correctness (musically speaking) of the compositions. This is a nice approach but possibly insufficient for our purposes.

Other approaches have been taken to evaluate the final entity *objectively* and *automatically* like in NEvAR [4], where the complex functions produced for each generation are considered to be better images than the preceeding set of images as well as being acceptable to a human.

V. TOWARDS PROCEDURAL GAME GENERATION

Taking all the processes involved in game development and considering that creativity is a key aspect in most of them, could we imagine successfully generating complete games procedurally?

The reader may quickly answer that question in negative as there are many significant problems to address. First, it is obvious that experienced designers cannot be replaced by any algorithm. Also, we can consider creativity and its results as a subjective issue which makes the process even more difficult to evaluate, which adds an extra challenge. Standards for describing games and managing game definitions are needed in a first step towards generating and running games automatically.

Despite this being one of the more difficult steps to address procedurally, Section IV refers to several theories and publications that can help in this process. Automation for evaluating games is also a highly subjective task and there is still much work to be done in this area. This makes it maybe one of the more problematic steps in the process and another hurdle in completely automatic execution.

Existing works

Some interesting approaches include Angelina, an AI system that can intelligently design videogames [19] or studies creating collaborative AIs and game content [20]. Some important advances in terms of tools have been made and are mentioned in Section V and well-documented in other papers [1].

Although these approaches are good contributions, some of them are too tightly focused on the theory itself or lack continuity over time. There are still some important gaps pending that have to be studied and solved. Things like generating games not based on a specific architecture or describing any kind of game (3D, 2D, first person, platforms, arcades, etc.) generically, and not heavily related to a specific base game are the main motivations behind advancing in this area.

Theoretical approach

Despite the nature and complexity of all the tasks involved, we should be able to imagine that, assisted by expert designers, processes involved in game development are programmatically possible. Counting on tools that aid the whole process, should mean that this approach can be automatically performed. This paper proposes a generic way of creating games procedurally, assisted in the first instance by designers. Also, in a more exhaustive way, designers will also be allowed to receive feedback from the generation process to allow them to refine the solution and obtain the best expected results. Figure 2 summarizes the main steps for a basic algorithm. These include:

- Designers, helped by a *tool*, will describe the game aspects. These include game rules, mechanics, associating bitmaps with game objects and maps and levels. That tool has to be powerful enough to allow designers to define every aspect necessary to complete the game. The better the tool used, the better the results obtained. For instance, if the tool does not allow generic end conditions for a game to be described and just uses a fixed number of them (e.g. lives zero or timeout conditions), designers will be tied in some way. The tool used to describe games must be flexible enough to generate a vast variety of games.
- Once the main game has been described, the tool should be able to export the game definition to any kind of format that can be read and even updated later on. In practice, the description tool should export to a file, or a set of files, that can subsequently be managed manually by designers or, more importantly for the survey, by a process in a procedural way.
- Given that the game description is complete and available in any format, we should be able to count on a generic game engine able to interpret it and run the game.

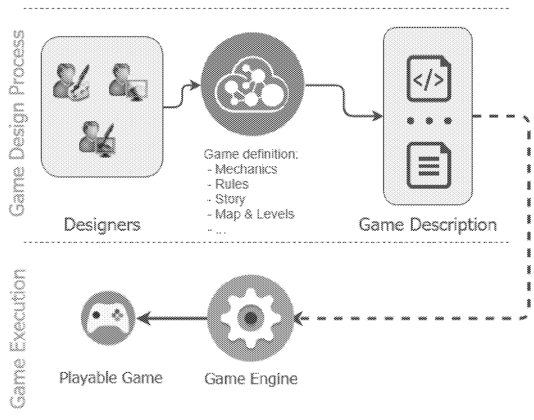


Fig. 2: Basic game generation process

The *tool* is crucial to the whole process as it has to be open enough to give designers the freedom to describe any kind of game and cover all necessary game design topics. The game description should be written in a standard format to be validated and managed by a game engine, which, in turn, must be able to read and run the game.

The process described is complete and the output is a game, but with some restrictions. This game will respond exclusively to the designers' input. Taking advantage of the computational power, the process may be refined and improved, including for generating games procedurally, evaluating and searching for best games given the designers' input. Adding the following aspects to the process will complete it in order to explore a search space and find better games. Figure 3 summarizes the process.

- Once a base game has initially been described, we can apply a search technique to help in the process of finding better games from a given initial game. Search and AI techniques have been used widely in game design and have been proved that they can help during the process [1].
- Using AI search techniques means the process has to have an evaluation procedure which is, as seen in previous sections, not an easy task to accomplish. It is essential that designers could define their preferences for evaluating each of the possible solutions. Apart from the initial design, they should provide objectives for the algorithm to be able to compare solutions and score them. In this way, they aid in the search procedure, leading to a solution that maximizes the evaluation criteria.
- Both the tool used to describe games and the game engine in charge of running the generated games must provide the possibility of applying AIs to enemies and other objects present in games.

- Game Engine will run games and generate an output that will be evaluated and processed by the selected search technique in order to generate a score for that complete simulation.
- Allow designers to take action during the game generation process, stopping the simulation, checking current results and giving feedback to the process to build even better games.
- Once the process has finished and considering final search conditions, the game engine will be able to export the output with the best evaluation score.
- The solution can be exported to any of the most well-known and commercial game engines such as Unity or Unreal allowing those games to be played and easing the step of generating games to final platforms like Windows, Unix, iOS or Android.

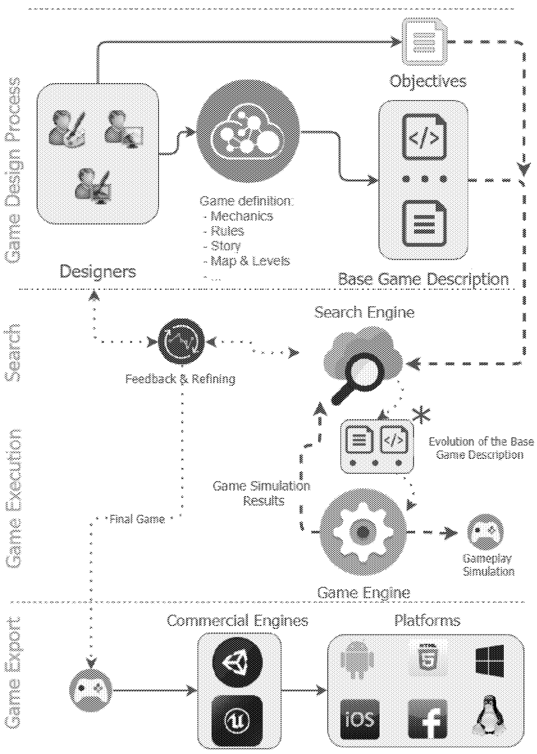


Fig. 3: Complete process of procedural generation of games using an AI search technique

A practical implementation

Some existing tools focusing on Game Design [1] will help in the process but one of the main points in the procedural generation of games is the game description. Videogame description languages (VGDL) [21] are definitely good candidates to be used in this part of the process. Existing papers cited in [1] may help us to find a good implementation but could still be found lacking for our purposes. The community does not yet have a standard VGDL, so designers have to choose one of the existing ones. Current VGDL proposals are more theoretical than practical approaches and they are not usually extensible. Additionally, they are not based on a standard language that can be managed by generic tools and that means the process has to be specific to each VGDL implementation.

A submitted paper under revision looks to be a very promising advance in this area, the Xml-Based Video Game Description Language (XVGDL) [22]. The XVGDL will solve many of these problems as it allows designers:

- It takes advantage of all previous VGDL research and implements not only game description elements but also allows specific properties to be defined and then considered by the game engine.
- It is based on XML, so it inherits its main features to make it extensible, maintainable, portable and open to any tool to be used easily. It is also flexible and open to any required future upgrades to improve the current definition.
- Provides mechanisms to describe main game aspects such as game objects, rules, mechanics, end conditions, physics and events.
- It is designed to be as abstract as possible in order to allow any kind of game to be defined including generators (for maps or levels) or assigning AIs to game objects.

XVGDL can be easily extended to allow designers to describe objectives. These objectives are a key part of the search process as they will be used to score each of the generated games. So, the search process will be directed mainly by these objectives and will try to find a good solution according to designers' expectations.

Once the input for the search process has been obtained and the designers' objectives are known, any of the well-known AI search techniques can be used. Options like a bioinspired algorithm, deep learning or reinforcement deep learning are widely-accepted techniques by the community [1] and can be very helpful in this process. The main point here is to map the game description into entities managed by those kinds of algorithms, for instance, coding the game description into a gene entity will allow the search process to be implemented as a genetic search algorithm. In this example, the gene could be generated using a codification of the game objects, rules, mechanics, events, etc. This gene can evolve and mutate as the algorithm iterates. The fitness function will be obtained based on the objectives configured by the designers.

A *game engine* is needed to execute game simulations. The way of determining if generated games are good or not is to play them. In this case, the idea is that the game engine could play those games generated during the search process, completely simulating game executions. At this point, we note that it is really important that the XVGDL not only allows AIs to be configured on game objects like items, enemies, projectiles, etc, but also it can assign an AI to game player. The game engine should also provide an interface that allows the search algorithm to capture game results such as final score, winning conditions, player lives left, execution time, among others. This information may then be used by the search algorithm to apply the fitness function and determine how good a solution is according to designers' defined objectives. Note that the game engine used need not be a commercial engine, just an engine that allows simulations of the configured games and give the necessary feedback to game designers.

When the search process finishes, designers will have a game configuration (could be described in the same XVGDL) maximizing their objectives. The next steps are then focused on obtaining a commercial game. This is an ambitious point but may lead the process to generate games for real platforms. As stated, using engines like Unity or Unreal can help in that task so the final process of converting the XVGDL to one of those commercial engines seems to be an especially important point for getting commercial games published across well-known different platforms.

CONCLUSIONS

After an initial study of the Computational Creativity panorama, reviewing the most important projects developed in several different fields like music, painting or writing, this paper has presented the idea of computational creativity applied to video games generation.

Creativity is strongly related to all game design processes and, thanks to the computational creativity perspective, all those processes are good candidates to be configured and processed automatically by any kind of computer process.

Game designers are still a key factor in the procedural generation process, contributing the initial base game definition and actively participating in the search process by giving extra feedback about current better solutions.

The idea of procedural games generation does seem possible now and a theoretical approach for performing that task has been explained in detail. Some existing studies and tools will be very helpful in the whole process but it may be that they are still not powerful enough to accomplish the most ambitious solution. The XVGDL, an XML-based VGDL proposal, provides a wide range of options and could make the process of describing games finally good enough to create different generic types of complete games. XVGDL is extensible and manageable and also allows applying any of the most well-known AI search techniques. That will lead to a process with procedurally generated games responding to initial designers' requirements.

An implementation of a game engine is needed for completely simulating the execution of a game. Providing enough information to the selected search algorithm, it may help in the process of finding better solutions.

Although XVGDL is being presented to the community, the game engine supporting XVGDL is still a work in progress, which once finished, may help to complete the process. This will finally allow exporting the end results to commercial game engines specific formats which in turn will lead to commercial games in several different platforms, completing the process of videogame generation.

ACKNOWLEDGEMENTS

This work has been supported by Spanish Ministry of Economy and Competitiveness (MINECO) projects: EphemeCH (TIN2014-56494-C4-1-P), and DeepBio (TIN2017-85727-C4-1-P) – Check: http://blog.epheme.ch and http://blog.deepb.io –, and Universidad de Málaga, Campus de Excelencia Internacional Andalucía Tech.

References

[1] J. Ruiz-Quiñones and A. Fernández-Leiva, "Game design: a preliminary review on techniques and tools," in *Proceedings of the 4th Congreso de la Sociedad Espaola para las Ciencias del Videojuego*, 2017, pp. 189–201.

[2] R. E. Franken, *Human motivation*. Pacific Grove, Calif, Brooks/Cole Pub. Co, 1994.

[3] R. W. Weisberg, *Creativity: beyond the myth of genius*. New York: W.H. Freeman, 1993.

[4] P. Machado and A. Cardoso, "Nevar – the assessment of an evolutionary art tool." The MIT Press, 2000.

[5] A. M. Noll, "The digital computer as a creative medium," *IEEE Spectrum*, vol. 4, no. 10, pp. 89–95, Oct 1967.

[6] A. Jordanous, "Four perspectives on computational creativity in theory and in practice," *Connection Science*, vol. 28, no. 2, pp. 194–216, 2016.

[7] "A visual interface for a collection of computer-generated narratives," *Information Engineering Express*, vol. 1, no. 4, December 2015.

[8] A. Taisuke and O. Takashi, "A hierarchical visualization program for computer-generated narratives," in *IIAI 4th International Congress on Advanced Applied Informatics*, July 2015, pp. 85–90.

[9] F. S. Khan, S. Beigpour, J. van de Weijer, and M. Felsberg, "Painting-91 : a large scale database for computational painting categorization," *Machine Vision and Applications*, vol. 25, no. 6, pp. 1385–1397, 2014.

[10] Deep dream. Accessed: 3rd August 2018. [Online]. Available: https://deepdreamgenerator.com/

[11] Csirac. Accessed: 3rd August 2018. [Online]. Available: https://cis.unimelb.edu.au/about/csirac/

[12] F. J. Vico, "Iamus - http://www.melomicsrecords.com," 2012, accessed: 3rd August 2018. [Online]. Available: http://www.melomicsrecords.com

[13] C. Raphael, "Demonstration of music plus one: A real-time system for automatic orchestral accompaniment," in *Proceedings of the 21st National Conference on Artificial Intelligence - Volume 2*, ser. AAAI'06, 2006, pp. 1951–1952.

[14] E. R. Miranda, "At the crossroads of evolutionary computation and music: Self-programming synthesizers, swarm orchestras and the origins of melody," *Evolutionary Computation*, vol. 12, no. 2, pp. 137–158, June 2004.

[15] C. Crawford, *The Art of Computer Game Design*. New York, NY, USA: McGraw-Hill, Inc., 1984.

[16] T. Fullerton, *Game Design Workshop : a Playcentric Approach to Creating Innovative Games*. A K Peters/CRC Press, March 2014.

[17] D. Ventura, "Beyond computational intelligence to computational creativity in games," in *2016 IEEE Conference on Computational Intelligence and Games (CIG)*, Sept 2016, pp. 1–8.

[18] A. Jordanous, "A standardised procedure for evaluating creative systems: Computational creativity evaluation based on what it is to be creative," *Cognitive Computation*, vol. 4, no. 3, pp. 246–279, Sep 2012.

[19] m. Michael Cook. Games by angelina. Accessed: 3rd August 2018. [Online]. Available: http://www.gamesbyangelina.org/

[20] M. Nogueira, J. M. Gálvez, C. Cotta, and A. J. Fernández-Leiva, "Hall-of-fame competitive coevolutionary algorithms for optimizing opponent strategies in a new game," in *Antonio J. Fernández-Leiva; Carlos Cotta Porras y Raúl Lara Cabrera, (eds.). 13th International Conference on Intelligent Games and Simulation. GAME-ON 2012*. Málaga, España: EUROSIS-ETI, Nov. 2012, pp. 61–63.

[21] N. Love, T. Hinrichs, D. Haley, E. Schkufza, and M. Genesereth, "General game playing: Game description language specification," *Stanford University*, 2008.

[22] J. Ruiz-Quiñones and A. Fernández-Leiva, "Submitted: Xvgdl: Xml-based video game description language," 2018.

FAST CONFIGURABLE TILE-BASED DUNGEON LEVEL GENERATOR

Ondřej Nepožitek, Jakub Gemrot
Faculty of Mathematics and Physics, Charles University in Prague
Malostranské náměstí 25, 118 00, Prague 1, Czech Republic
E-mail: ondra@nepozitek.cz, gemrot@gamedev.cuni.cz

KEYWORDS

Procedural content generation, Dungeon levels, Stochastic method, Simulated Annealing

ABSTRACT

Procedural generation of levels is being used in many video games to increase their replayability. But generated levels may often feel too random, unbalanced and lacking an overall structure. Ma et al. (2014) proposed an algorithm to solve this problem; their method takes a set of user-defined building blocks as an input and produces layouts that all follow the structure of a specified level connectivity graph. In this paper, we present an implementation of this method in a context of 2D tile-based maps. We enhance the algorithm with several new features and propose speed improvements. We also show that the algorithm is able to produce diverse layouts. Benchmarks show that it can achieve up to two orders of magnitude speedup compared to the original method. As the result, it is suitable to be used during game runtime.

INTRODUCTION

Procedural content generation (PCG) is a method of creating content algorithmically rather than by hand (Togelius et al. 2010). In video games, it is often used to increase game's replayability. The classic example is the game Rogue that contains procedurally generated dungeon levels, treasures and monster encounters, which lead to unique experience on every playthrough. Procedural techniques are also used in newer games including Borderlands, Diablo or Minecraft.

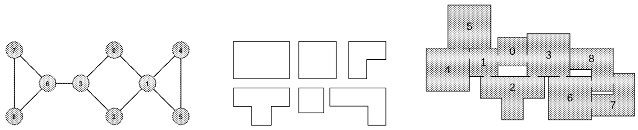
In this paper, we focus on PCG of game levels. One approach to this problem is to use binary space partitioning (Shaker et al. 2016); they start with a rectangular area and recursively split it until there are enough subareas. Some subareas are then chosen to represent rooms and these are then connected by corridors. Another possible approach is so-called agent-based dungeon growing (Shaker et al. 2016); they start with an area that is completely filled with wall cells and an agent is spawned at a specified location. The agent is controlled by a predefined AI and moves through the area, digging corridors and placing rooms.

The problem with these algorithms is that a game designer often loses control over the flow of gameplay, and generated layouts may feel too random and lacking an overall structure (Dormans and Bakkes 2011, Ma et al. 2014). Although this approach may be appropriate in some genres, Dormans & Sanders (Dormans and Bakkes 2011) note that it is not suitable for action-adventure games and propose to generate both missions and spaces of a game using generative grammars. Ma et al. (Ma et al. 2014) propose a different approach. Their method takes a set of room shapes and the level connectivity graph as an input and produces layouts that all follow the defined structure; a game designer thus have complete control over the high-level structure of a level, for example, a control over possible sequences a player can visit respective rooms (graph nodes). In this paper, we describe conceptual and technical improvements to the procedural generation algorithm for dungeon levels developed by Ma et al.

The rest of the paper is structured as follows. We first describe the algorithm in detail, then we show our new features and speed improvements. Finally, we evaluate algorithm's performance and conclude the paper.

ALGORITHM

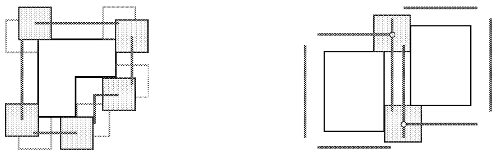
To produce a dungeon level, the algorithm (Ma et al. 2014) takes a set of polygonal building blocks (referred to as room shapes) and a planar level connectivity graph (the level topology) as an input. Nodes in the graph represent rooms, and edges define connectivities between them. The goal of the algorithm is to assign a room shape and a position to each node in the graph so no two room shapes intersect and each pair of neighbouring room shapes share a common boundary segment (Figure 1).



(a) Input graph (b) Building blocks (c) Output

Figure 1: Example output of the algorithm.

Instead of searching through all possible positions and room shape assignments of graph's nodes, the algorithm uses configuration spaces to define valid relative positions of individual room shape pairs. However,



(a) (b)

Figure 2: Configuration spaces. (a) the configuration space (red lines) of the free square with respect to the fixed l-shaped polygon. (b) the intersection (yellow dots) of configuration spaces of the moving square with respect to the two fixed rectangles.

formulating the whole problem as a configuration space computation was shown to be PSPACE-hard (Hopcroft et al. 1984). Therefore, a probabilistic optimization technique is used to efficiently explore the search space. To further speed up the optimization, the input problem is broken down to smaller and easier subproblems. This is done by decomposing the graph into smaller parts (called chains) and laying them out one at a time.

Configuration spaces

For a pair of polygons, one fixed and one free, a configuration space is a set of such positions of the free polygon that the two polygons do not overlap and share common segment(s). With polygons, a configuration space can be represented by a (possibly empty) set of lines (Figure 2).

Because the block geometry is fixed during optimization phase, configuration spaces of all pairs of block shapes are precomputed to speed up the process.

Incremental layout

Algorithm 1 assigns positions and room shapes to graph nodes incrementally; in each step it lays out one chain. A chain is a sub-graph where each node has at most two neighbours. Chains have an advantage that they are relatively easy to lay out. The next chain to connect is always one that is connected to already laid out nodes.

```
Input: planar graph G, building blocks B, layout stack S
procedure IncrementalLayout(c,s)
 Push empty layout into S

 repeat
  s ← S.pop()
  Get the next chain c to add to s
  AddChain(c, s)              // extend the layout to contain c

  if extended partial layouts were generated then
   Push new partial layouts into S
  end if
 until target # of full layouts is generated or S is empty
end procedure
```

Algorithm 1: Incremental layout.

In each iteration, we take the last layout from the stack and try to add the next chain, generating multiple extended layouts and storing them. If this step fails, no new partial layouts are added to the stack and the algorithm has to continue with the last stored partial layout (referred to as *backtracking*). It is usually needed when there is not enough space to connect additional chains to already laid out nodes. The process terminates when enough number of full layouts are generated or if no more distinct layouts can be computed.

To decompose a graph into chains, we first compute a planar embedding of the graph (Chrobak and Payne 1989). The first chain is then formed by the smallest face of the embedding and following faces are added in a breadth-first order. If there are more faces to choose from, we first lay out the smallest one. When there are no faces (no cycles) left, remaining acyclical components are added.

Favoring cycles is quite important as we have empirically confirmed them to be harder to layout and causing the algorithm to backtrack unnecessarily (as the original paper states).

Simulated annealing

The authors of the original algorithm chose simulated annealing (SA) framework to explore the space of possible layouts for individual chains. The reason is that it produces multiple partial layouts in a single run, which is useful in two situations. First, it allows us to backtrack if we are unable to lay out a chain. Second, we are able to quickly generate subsequent full layouts. Instead of starting the generation process all over again from an empty layout, we start with an already computed partial layout that was produced by SA previously.

```
Input: chain c, initial layout s
procedure AddChain(c,s)
 generatedLayouts ← Empty collection of generated layouts
 t ← t0                          // Initial temperature

 for i ← 1,n do                  // n: # of cycles in total
  for j ← 1,m do                 // m: # of trials per cycle
   s' ← PerturbLayout(s, c)

   if s' is valid then

    if s' ∪ c is full layout then output it
    else if s' passes variability test

     Add s' into generatedLayouts
     Return generatedLayouts if enough extended layouts computed
    end if
   end if

   if ΔE < 0 then                // ΔE = E(s') - E(s)
    s ← s'
   else if rand() < e^(-ΔE/(k*t)) then
    s ← s'
   else
    Discard s'
   end if
  end for

  t ← t * ratio                  // Cool down temperature
 end for
end procedure
```

Algorithm 2: Simulated annealing. This pseudocode uses $n = 50$, $m = 500$, $t_0 = 0.6$ and k is computed using ΔE averaging (Hedengren 2013).

SA operates by iteratively considering local perturbations to the current configuration, or layout. The energy function is constructed in a way that it heavily penalizes nodes that intersect and neighbouring nodes that do not touch.

To speed up the process, they try to find an initial configuration with a low energy. To do that, a breadth-first search ordering of nodes from the current chain is constructed, starting from the ones that are adjacent to already laid out nodes. Ordered nodes are placed one at a time, sampling the configuration space with respect to already laid out neighbours, choosing the configuration with the lowest energy.

Tile-based output

We provide an implementation of the algorithm in a context of tile-based maps. Therefore we changed the representation of shape coordinates from original floats to integers and we use only rectilinear polygons instead of arbitrary polygons for room shapes.

NEW FEATURES

Corridors between rooms

In the original paper, it is shown that the method can be used to generate layouts with rooms connected by corridors. To achieve that, a new node is added between every two neighbouring nodes into the input graph and these new nodes get assigned a set of room shapes that was made for corridors.

The problem of this approach is, that we now have almost twice as many nodes than before, which slows the generation down.

In our approach, we use two different instances of configuration spaces. The first is the standard one, in which a position of two rooms is valid when both rooms touch and do not overlap. The second, on the other hand, accepts only positions where the two rooms are exactly a specified distance away from each other (and also do not overlap).

When we perturb a layout, we first use the second type of configuration spaces. By doing so, we should converge to a state where all pairs of non-corridor nodes of the current chain have a space between them. Then we switch to the first type of configuration spaces and try to greedily add all corridors rooms. If we are not able to lay out all corridor rooms this way, we abort the current attempt, remove already added corridors and return to SA.

Explicit door positions

In the original algorithm, it is not easy to specify door positions within individual room shapes. But this can be useful, for example, if we have a boss room which requires the player to enter the room from a specified tile, or if we have a room template with some tiles reserved for furniture, chests, etc.

Therefore, we extended configuration spaces generator to work with door positions directly. It allows us to define door positions of every room shape in a layout explicitly. This modification has no runtime overhead as configuration spaces are generated only once before the algorithm starts. One must be careful though as having too few door positions makes it significantly harder for SA to connect neighbouring rooms and will often cause the algorithm to need more iterations to generate a valid layout.

Custom constraints

The original method enforces two basic constraints on the layout - no two rooms may overlap and all neighbouring rooms must be connected by doors. We decided to make the concept of constraints more general and customizable. We allow to define constraints over the whole layout and constraints over individual nodes.

They can be either hard or soft. All hard constraints must be satisfied before a layout can be accepted whereas soft ones are used to control the evolution by modifying the energy.

This can be useful, for example, if we want to make sure that the whole layout does not exceed some defined boundaries or if we create an obstacle that the layout must not intersect.

SPEED IMPROVEMENTS

Simulated annealing parameters

We observed that SA spends most of the time on runs that either fail to generate anything or do not produce enough partial layouts. Such situations happen mostly if the current chain cannot be laid out because of an unlucky positioning of previous chains. With this in mind, we tried to find ways to terminate non-perspective runs as soon as possible.

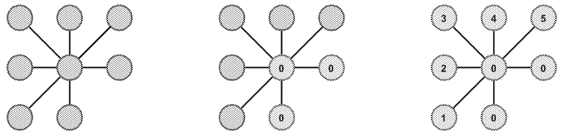
The original algorithm uses a mechanism of random restarts. If we do not accept any state for too long, we quit the current run of SA. The problem is that we can accept a lot of states without producing a single valid layout. Generating valid layouts, however, is our main goal. We tried three different mechanisms that decide if the current iteration of SA is successful or not:

1) is the original approach where we penalize iterations that fail to accept new states; 2) penalizes iterations that fail to produce valid layouts; 3) is the most strict one and penalizes iterations that fail to produce valid layouts that are different enough from already generated layouts.

We benchmarked all three possibilities and the most strict came out as the best one. We also tried various values of parameter m (trials per cycle) and decided to set it to 100 from the original 500.

Chain decomposition

The decomposition of an input graph into chains affects the overall performance of the algorithm. We observed that having a lot of small chains in a decomposition leads to poor performance, especially in situations where we have to backtrack very often. The problem



(a) (b) (c)

Figure 3: Problematic input for chain decomposition. Blue nodes are contained in a chain with a corresponding number. (a) shows the input graph. (b) shows a partial chain decomposition after the first iteration of the algorithm. (c) shows a complete chain decomposition of the graph (6 chains in a graph of 8 nodes).

is that a substantial amount of time is spent when initializing the process of laying out the next chain. For example, it is quite time-consuming to find the best initial configurations for nodes in the current chain.

In Figure 3a we can see an example of a problematic graph. Figure 3b shows how the decomposition may look like after the first iteration of the basic decomposition algorithm. We can see that we have a lot of small acyclic components now. The problem is that the algorithm creates a new chain from every such component. And finally, in Figure 3c, we can see that we will end up with a lot of small chains - which is a situation we want to avoid.

Our solution is quite simple. When we want to add a node to a chain, we check if it does not create acyclical components with only one node. If it does, we add all such components to the current chain. This violates the definition of a chain, but we found out this approach to perform better in practice.

Lazy evaluation

Another technical problem is that the algorithm is trying to generate multiple layouts in each run of simulated annealing in case we need to backtrack later. But what if we are lucky and do not need to backtrack? In that case, we have wasted a lot of time by computing something that is not needed.

The solution is to make the computation lazy. Instead of generating all children layouts at once, we save the state of the current run of the algorithm and resume it later only if it is truly required; as we are using C#, this is easily achievable by using *yield* statement.

EVALUATIONS

All benchmarks presented here are obtained by running our algorithm 100 times on each input graph, with different randomization seeds. We measure the time that is needed to generate a layout and the number of iterations, i.e., how many times we need to perturb a layout to generate a full layout.

In Table 1, we can see a benchmark of our method when used on input graphs presented in Figures 1 and 5. Our method was able to generate all layouts without corridors in under one second. And all layouts with corridors in under two seconds. Our algorithm is therefore quick enough to generate layouts directly in a game.

Input	Time avg/med	Iterations avg/med
Fig. 1	0.00s/0.00s	0.12k/0.02k
Fig. 5, top-left	0.12s/0.08s	4.15k/2.78k
Fig. 5, top-right	0.01s/0.00s	0.29k/0.17k
Fig. 5, bottom-left	0.18s/0.09s	5.50k/3.20k
Fig. 5, bottom-right	0.62s/0.36s	15.28k/10.10k

Table 1: Benchmark of our final implementation of the algorithm. All benchmarks were done with the building blocks from Figure 1b, on a 2.7GHz CPU (the algorithm runs on a single core).

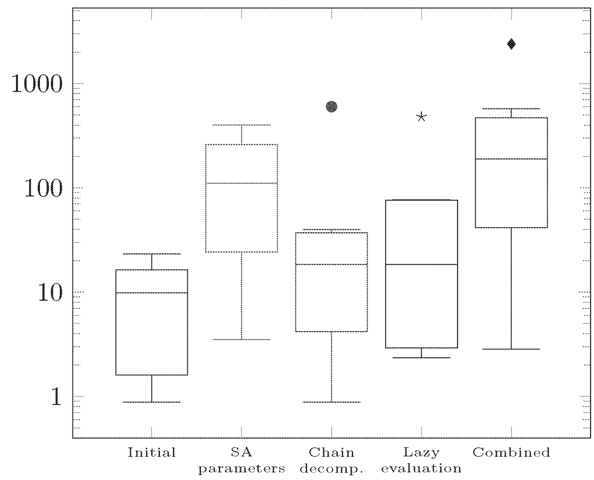


Figure 4: Relative speedup of our performance improvements when compared to the implementation of the original method, available at Github (Ma et al. 2014). The chart shows how different approaches affect the total time needed to generate a layout.

Performance comparison

In the Speed improvements section, we described our most important performance improvements. Figure 4 shows a comparison of how these changes affect the overall speed of the algorithm.

The first bar shows the performance of our initial implementation of the original method in a tile-based context. We can see that this implementation is already faster than the original algorithm. This is mainly caused by the fact that with integer coordinates, we were able to significantly speedup operations with polygons and energy function computation.

The following three bars of the chart demonstrate how the algorithm behaves with only one improvement enabled. And finally, the last bar shows that with all improvements enabled; our algorithm is over 100 times faster than the original method.

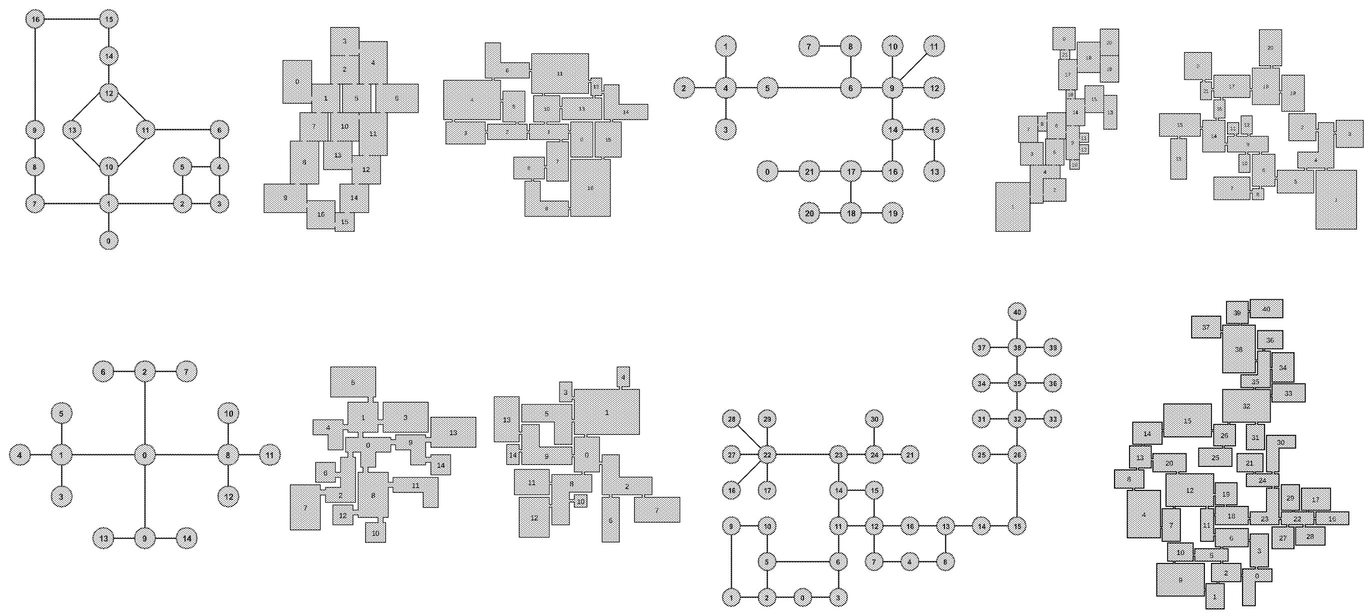


Figure 5: Layouts generated from four different input graphs, using various sets of building blocks.

CONCLUSION

We presented an algorithm for procedural generation of tile-based maps from user-defined building blocks that adapts and improves the previous work of Ma et al.

We proposed several new features and speed improvements. Users can now easily specify door positions and add custom constraints on the layout. We also presented a method to quickly generate layouts with rooms connected by short corridors as usually found in dungeon levels.

We demonstrated that our method can handle various input graphs and building blocks sets. Benchmarks of our method showed that, on average, our algorithm is over 100 times faster then the original one, and is able to generate a layout in under one second for all our inputs in the basic mode without corridors. This makes our algorithm fast enough to be used directly in a game during runtime or as an inspiration for game designers during design time.

Our C# implementation of the algorithm can be downloaded from `https://github.com/OndrejNepozitek/ProceduralLevelGenerator` under the MIT License, which allows the result to be used in commercial games. The repository also contains an extended version of the paper together with an example of a practical use-case of the algorithm.

ACKNOWLEDGEMENTS

This research was funded by the Czech Science Foundation (project no. 17-17125Y).

REFERENCES

Chrobak M. and Payne T., 1989. "A linear-time algorithm for drawing a planar graph on a grid". *Information Processing Letters*, 54, no. 4, 241–246.

Dormans J. and Bakkes S., 2011. "Generating Missions and Spaces for Adaptable Play Experiences". *IEEE Transactions on Computational Intelligence and AI in Games*, 3, no. 3, 216–228.

Hedengren J., 2013. "Simmulated annealing tutorial". `http://apmonitor.com/me575/index.php/Main/SimulatedAnnealing`.

Hopcroft J.; Schwartz J.; and Sharir M., 1984. "On the Complexity of Motion Planning for Multiple Independent Objects; PSPACE- Hardness of the "Warehouseman's Problem"". *International Journal of Robotics Research*, 3, no. 4, 76–88.

Ma C.; Vining N.; Lefebvre S.; and Sheffer A., 2014. "Game Level Layout from Design Specification". *Computer Graphics Forum*, 34, no. 2. `https://github.com/chongyangma/LevelSyn`.

Shaker N.; Togelius J.; and Nelson M.J., 2016. *Procedural Content Generation in Games: A Textbook and an Overview of Current Research*. Springer.

Togelius J.; Yannakakis G.N.; Stanley K.O.; and Browne C., 2010. "Search-based Procedural Content Generation". In *Proceedings of the 2010 International Conference on Applications of Evolutionary Computation - Volume Part I*. Springer-Verlag, Berlin, Heidelberg, EvoApplicatons'10. ISBN 3-642-12238-8, 978-3-642-12238-5, 141–150. doi:10.1007/978-3-642-12239-2_15. URL `http://dx.doi.org/10.1007/978-3-642-12239-2_15`.

CREATING GAMEPLAY MECHANICS WITH DEFORMABLE CHARACTERS

Grant Clarke
School of Design and Informatics
Abertay University
40 Bell Street, Dundee, United Kingdom, DD1 1HG
email: g.clarke@abertay.ac.uk

KEYWORDS

Physics-based Animation, Gameplay Mechanics, Soft Body Simulation

ABSTRACT

This paper presents how soft body simulation can create deformable characters and physics-based game mechanics that result in a more varied gameplay experience. A framework was implemented that allows the creation of a fully deformable soft body character within a games application where the simulation model properties could be altered at runtime to create gameplay mechanics based on varying the deformation of the character. The simulation model was augmented to allow appropriate methods of player control that complemented the character design and its ability to deform. It was found that while the implementation of deformation-based mechanics created a more varied gameplay experience, the underlying simulation model allowed for a limited amount of deformation before becoming unstable. The effectiveness of the framework is demonstrated by the resulting mechanics that are not possible through the use of previous methods.

INTRODUCTION

Many computer games use physics simulation as an underlying system to create real-world-like motion in objects. Whether it's to create the feel of driving vehicles over undulating terrain or have characters fall, ragdoll-like, down a staircase. In terms of game design, titles such as Marble Madness (Atari Games 1984) and Angry Birds (Rovio Entertainment 2009) use physics-based gameplay to create their core game mechanics. This can result in dynamic, but predictable behaviour that can provide players with a varied and engaging user experience.
These games are underpinned by rigid body physics simulation where the shape of the player character within game remains constant throughout the simulation. While this doesn't necessarily limit the different physics-based gameplay mechanics that can be created, it does suggest that the additional use of soft body simulation to create deformable characters could lead to some innovate player experiences in the area of physics-based gameplay.

Soft body dynamics in games isn't new, but its use is often limited to improving the visual aesthetic of digital characters rather than form a core part of the gameplay mechanics. For example, simulated cloth on character garments or providing secondary animation in the form of belly jiggle to make a character's movement appear more dynamic.

To create deformable characters there are several methods of soft body simulation, and the choice of which to use is dependent on application requirements in terms of performance, stability, deformation properties and the means of control.
Mass-spring models have been popular in games to create elastic deformable materials due to the relative simplicity and efficiency of the underlying implementation. The elasticity is controlled by manually adjusting the spring stiffness and damping constants to create the desired amount of deformation across the body.
Finite element methods can be used to create a more physically accurate model of deformable materials that contain both elastic and plastic properties. While more computationally expensive than a mass-spring model, interactive rates are possible. Values that define the deformation properties of real-world materials can be used with this model potentially reducing the amount of manual adjustment required. (Müller and Gross 2004)
Parker and O'Brien (2009) demonstrates the application of a FEM-based model to create realistic deformation and fragmentation in a game environment. The deformation of objects is solely used to improve the visual aesthetics, but nevertheless, this shows this method can be used in a fully featured commercial game.
Position based dynamics allows for more control as the position of the particles that form the soft body can be manipulated directly without compromising the integrity of the simulation model. It is also unconditionally stable and efficient while creating plausible, but not wholly accurate deformations.(Müller et al. 2007)

In terms of soft body control, Liu et al. (2013) proposes a skeleton-driven approach that drives the motion of an FEM-based soft body. Skeletal animation allows for creative control of the overall motion of the soft body character in cases where procedurally generating a similar motion would be too complex. The two-way coupling between the soft body skin and rigid body skeleton does allow for applied forces on the soft body to affect the underlying skeleton, but the rigid skeleton does limit the amount of deformation possible. Tan et al. (2012) proposes the use of simulated muscle fibres to control the motion of FEM-based soft bodies. This allows for the creation of more cartoon-like motion that exhibits a greater range of squash-and-stretch when compared to skeleton-based methods as the character motion is not restricted by the rigidity of an underlying skeleton. However, results show that this method is too computationally expensive for interactive rates required for game applications.

There are some notable game applications that use character deformation as part of their core mechanics. Gish (Cryptic Sea 2003) and Cats are Liquid (Last Quarter Studios 2015) are 2D platform games that let the player increase the elasticity of their main characters so they can squeeze through small gaps in the level.
Deformers (Ready At Dawn 2017) is a fast-paced multiplayer battle game where the elasticity of the characters helps to give the characters a more cartoon-like quality and their ability to change shape from a ball to a cube is used in the games blocking mechanic.
Claybook (Second Order 2017) uses a custom position based dynamics system to create a fully deformable environment where the player can use the shape-changing character to alter and interact with the world to solve puzzles.

Reviewing related work has shown that creating and controlling deformable characters is an active area of research in computer graphics, but has also shown there is a lack of literature into the design and implementation of soft body physics-based game mechanics.

By focusing on the unique behaviours that can be created through the use of soft body dynamics this paper aims to explore how deformable characters can be used to create a wider range of physics-based game mechanics.

METHODOLOGY

Two gameplay prototypes were created with different character representations. The following approach was taken when creating deformation-based game mechanics in each prototype:

- Identify the characteristics of the simulation model and the properties that can be manipulated to alter its behaviour.
- Define a method of controlling the basic movement of the character.
- Formulate deformation-based character behaviours that can be created by interacting with the underlying simulation model.
- Using one or more behaviours, create game mechanics that are generally not feasible via a more traditional rigid body character representation.

Identifying Characteristics of the Simulation Model

A FEM-based model was chosen because of its predominant use in existing research in creating deformable characters. The soft body is represented by a mesh of tetrahedral elements and it is the simulated movement of these elements that cause the deformation. The model is characterised by the following properties: elasticity, plasticity and fracturing.

Elasticity allows the body to deform when under stress. Stress is applied in the form of external forces to the individual elements that describe the body. Removing these external forces result in the body returning to its undeformed state. The elasticity of the body is controlled by two parameters, Young's modulus which describes the stiffness and Poisson's ratio which describes how a solid material tends to expand or contract in a direction perpendicular to any longitudinal stress the soft body is under.

Plasticity allows deformations to remain when external forces are removed. This is achieved by storing the plastic strain caused by the internal elastic strain on the elements that form the soft body. It is this plastic strain that retains the deformation. As it is a stored value within the model, it can be removed at any time. The plasticity is controlled by three parameters, $plastic_{yield}$, the threshold amount of elastic strain that when exceeded, results in the storing of plastic strain. $plastic_{creep}$, the rate at which plastic strain is applied and $plastic_{max}$, the maximum amount of elastic strain that is permitted.

Fracturing allows tearing of a soft body when the internal stress of elements within the body exceeds a fracture threshold. It was decided that for these initial prototypes, fracturing would not be used to avoid the complexity of implementing a character that consisted of multiple separate parts.

The individual parameters from each property were combined to describe what was referred to as a physics material. The implementation of the FEM model was amended to allow the different elements within the model to have their own material applied to them. This allowed for different parts of the model to have different deformation characteristics to one another and produces a much more flexible simulation model.

Prototype 1

To explore the initial challenges in creating deformable characters and the subsequent game mechanics, a ball-shaped character consisting of a single physics material was created.

Character Physics Model

A tetrahedral mesh defines the character shape in the soft body simulation and mesh coupling is used to deform the high-resolution graphical mesh seen in-game.

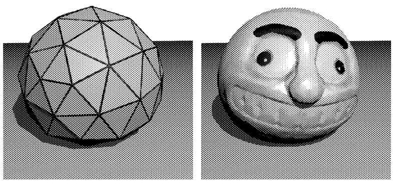


Figure 1: Ball-shaped Character.

Table 1: Material settings for spherical character.

Tetrahedra	118
Surface triangles	118
Density $[kg/m^3]$	1000
Young's modulus $[N/m^2]$	100000
Poisson's ratio	0.99

The refinement of the tetrahedral mesh was chosen so that it was high enough to produce a level of deformation that looked visually plausible when the character collided with objects in the environment, but not cause a significant drop in frame rate.

Control Method for Basic Movement

The ball-shaped character's overall shape is virtually identical to a comparable spherical rigid body. It was therefore decided to mimic the movement mechanic seen in Marble Madness-style games where the direction the character moves in is controlled by 2D axis user input (forward and back, left and right). In order to achieve this, a frame of reference is needed for the soft body. In a rigid body, this is simply the position and rotation of the body. As a soft body consists of a collection of interconnected elements that can be moving in different directions as a result of varying forces being applied across the body, there is no explicit position and rotation for the body as a whole. To achieve this, a fixed node at the centre of the soft body was added during the generation of the tetrahedral mesh.

Because the motion of the body is controlled by applying a force to the body in the desired direction of travel irrespective of the overall orientation of the ball, there is no need to explicitly calculate the overall orientation of the soft body ball. To determine the current direction of travel for the ball, the position of the origin node is sampled over time to calculate its velocity. This velocity is a useful attribute for the player model to expose. For example, it can be used to implement a camera the orients itself with the direction the player is moving in. With the addition of a frame of reference, any mechanic (e.g. jumping) that is straightforward to implement on a rigid body can be transferred to the soft body simply by applying the appropriate uniform force to all elements within the soft body.

Formulating Behaviours and Mechanics

"Softening" The Character

By decreasing the stiffness of the soft body, the increased amount of deformation enables the character to be squeeze through gaps in the environment they would not normally be able to pass if they were implemented as rigid bodies.

This can be achieved simply by lowering Young's modulus in the physics material over time to create a more malleable character.

It's worth noting that with the FEM simulation model chosen here, changing Young's modulus in the physics material requires parts of the underlying simulation model to be reinitialised. The amount of process time this takes is proportional to the number of elements present in the tetrahedral mesh and is, therefore, another place the refinement of the mesh can have an impact on performance.

The player can now consider how this "softness" can be used to traverse the game environment. Existing games, like Gish and Cats are Liquid, enable this ability through user control, but it doesn't have to be limited to this type of use. Environmental objects can be used to change the state of the player providing a range of ways to interact with this mechanic. For example, simulated heat can be used to temporarily melt the character and as it moves from the heat source, it can return its default stiffness over time.

Stretching The Character

In order to explore how stretching can be used as a mechanic, positional constraints were used to temporarily hold a part of the character in place. In essence, a sticky surface was created. Without these constraints, any external forces used to stretch a part of the character would result in dragging the entire shape. The initial concept was to apply stretch forces to the unconstrained side of the character in a direction perpendicular to the sticky surface and see what effect this could have in terms of gameplay. For this character, the overall shape and topology of the underlying tetrahedral mesh prohibited plausible deformations taking place. This was due to the relative coarseness of the mesh and the rolling nature of the character movement. When the character rolled into the sticky surface, depending on its orientation, a varying number of nodes in the tetrahedral mesh would become constrained to the block and in the case where this may have been a low number of contact points, the stretching would cause unwanted irregular spikes in the deformation. This could be potentially be counteracted by increasing the refinement of the mesh, but this then had a significant detrimental on performance. This approach was therefore abandoned.

Prototype 2

Here a cylindrically shaped character was used for a number of reasons. The shape allows more potential to create stretch based mechanics due to its shape and topology. Its shape is very similar to the capsule shape that is commonly used to represent characters in physics simulations for games. This would allow the creation of deformable characters whose basic movement is implemented in a way familiar to the player.

Character Physics Model

To create a cylinder-based character that exhibits elasticity in its default state, multiple physics materials were used to create the tetrahedral model. The bottom section was created with a physics material with a higher density and stiffness to the middle and top sections so the character wouldnt topple easily. The stiffness of the remaining sections was set to create a natural sway in the character as it moved to emphasise it's deformable nature.

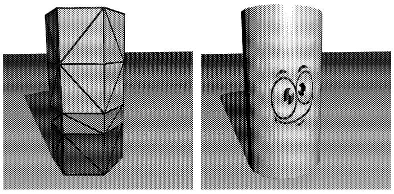


Figure 2: Cylinder-shaped Character.

Table 2: Material settings for cylindrical character.

Tetrahedra	104
Surface triangles	72
Density $[kg/m^3]$	750
Poisson's ratio	0.99
Bottom: Young's modulus $[N/m^2]$	500000
Mid: Young's modulus $[N/m^2]$	300000
Top: Young's modulus $[N/m^2]$	10000

Control Method for Basic Movement

Character controllers in game engines such Unity (Uni 2008) allow for more precise control by setting the desired linear and angular velocity of the controller rather than through the application of forces when momentum based acceleration/deceleration is not wanted. To duplicate this in the deformable character, a reference coordinate frame is needed that provides both position and orientation. A fixed node was added in the tetrahedral mesh generation in the bottom region to define the origin of the character, similar to the ball-shaped character. As orientation is also needed, additional nodes were added to define an orthogonal coordinate frame within the bottom region.

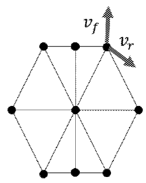


Figure 3: Linear forces acting on core nodes.

To create the controllable movement in the character the bottom region was treated as a rigid core that would essentially pull the rest of the character along as it moved. To achieve this, the linear velocity of each node in the core region was set after the soft body simulation was run each frame.

The total linear velocity of each node within the core region is defined as:

$$v_{total} = v_f + v_r \tag{1}$$

where v_f is the desired linear velocity and v_r is the linear velocity resulting from the desired angular velocity, which is subsequently defined as:

$$v_r = \omega \times r \tag{2}$$

where ω is the desired angular velocity and r is the position of the core node relative to the core origin node. This essentially created a controllable rigid lower region coupled with soft deformable upper regions but without the need to couple a separate rigid body simulation with the soft body simulation model.

Rigid body based mechanics like jumping can now be created to the cylinder-based character by applying uniform forces to all nodes in the core region or through the manipulation of the desired linear and rotational velocities of the character.

Formulating Behaviours and Mechanics

Stretching Up

A stretch mechanic was created to pull the character into a longer shape. This was achieved by constraining the top and bottom of the character with anchors and then raising the top anchor over a period.

During this time, the plasticity parameters were altered to allow the deformation to remain once the anchor was moved. At this point, the plastic creep would be reset to zero to prevent further deformations to remain permanent. This was required so the default elastic movement in the character would return. In this elongated state, the character could now reach items or locations in the level which were too high without the change in shape.

Table 3: Plasticity settings for stretching

.

$plastic_{yield}$	0.02
$plastic_{creep}$	20
$plastic_{max}$	0.613

Squashing Down

A squash mechanic was created using a similar method to the stretch mechanic, but with the top anchor moving downward. This allows the character to shrink down to a smaller size and fit under obstacles. This shows how the deformation can be used to achieve something similar to existing crouch mechanics.

Table 4: Plasticity settings for squashing

.

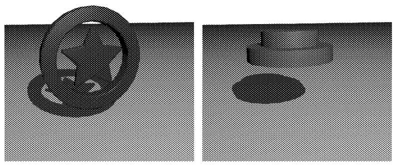
$plastic_{yield}$	0.02
$plastic_{creep}$	20
$plastic_{max}$	0.35

Swinging

To create a more novel means of traversing gaps within a game level, a swing mechanic was created. The top surface of the character was constrained to an anchor within the level. The core based character movement was disabled at this point and a linear acceleration was applied based on the forward user input. This linear acceleration, combined with the centre of mass in the bottom region of the character and the elasticity of the upper regions created a natural swinging motion in the character. The player is then able to build up the swing momentum by alternating the forward user input in time with the swing and cover a greater distance when released from the anchor point. Additionally, when the character is anchored at the top it can also be pulled down to stretch its shape. This is achieved by lowering Youngs modulus to 9000 in the physics material for the top region and apply a downward force to the origin node in the core region. This allows for the character to reach items or locations in the level which were too low to reach without the change in shape.

RESULTS

Figure 4 shows objects used in the prototypes to represent the goal to reach for and the anchor used to constrain the character when creating squash and stretch. Figures 5 - 8 show the visual deformation of the character and demonstrates how the implemented mechanics allow the character to reach an item that would be unattainable if the character was unable to deform.



(a) Goal object (b) Anchor object

Figure 4: Prototype objects.

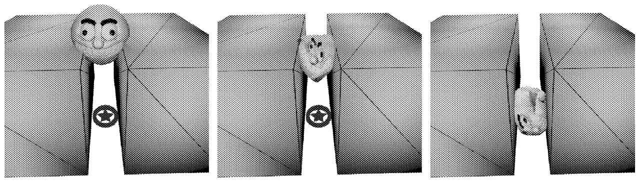


Figure 5: Softening mechanic.

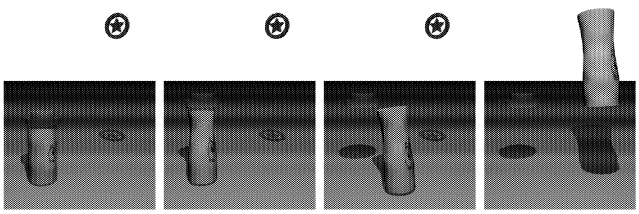


Figure 6: Stretching mechanic.

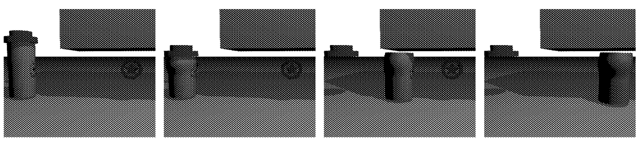


Figure 7: Squashing mechanic.

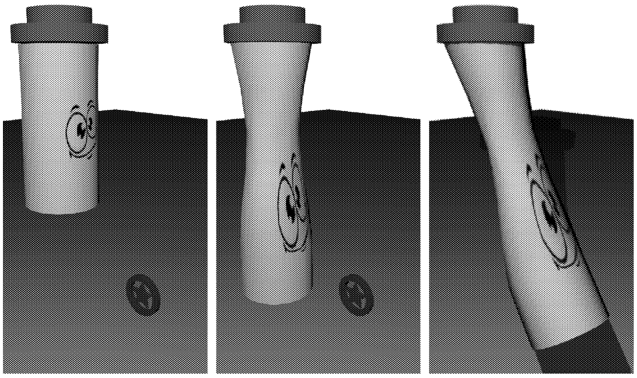


Figure 8: Swinging mechanic.

DISCUSSION

Qualitative Evaluation

The prototype with the ball-shaped character was successful as a proof of concept in terms of testing the pipeline and creating a deformable character like those seen in existing titles. It also demonstrated that beyond the ability to change the overall elastic characteristics of the body to enable the character to squeeze through gaps, limitations based on the mesh topology and the rolling nature of the movement made it difficult to implement other deformation based behaviours. This may indicate why existing games with this style of deformable character predominately use this mechanic and nothing beyond it.

The formation of the tetrahedral mesh topology in the cylinder-based character allows for a more varied use of the characteristics of the simulation model. Constraining specific nodes to anchors that were in line with the stretch direction created deformation in keeping with the overall shape of the character when compared to the deformation problems seen in the ball-shaped character.

The implementation of the rigid core region in the cylinder-shaped character allows for this style of deformable character to use similar methods of motion control to those found using rigid body-based character controllers. By avoiding the coupling of a rigid body to create this core, allows for this region to interchange between rigid and soft body behaviour when used to implement different mechanics.
The elastic and plastic properties of the FEM simulation model enabled the creation of novel deformation-based game mechanics in this relatively small set of experiments. The limitations of the implementation lead to restrictions when balancing the refinement of the generated tetrahedral mesh with runtime performance. Large node displacements did result in invalid model configurations for the soft body simulation. This was particularly evident in the creation of the squashing mechanic where only a relatively small downward displacement was possible before the model folded in on itself. This could potentially be solved by implementing collision response between internal nodes in the tetrahedral mesh. In addition, changing material parameters often required an expensive reinitialise step. While the properties of this simulation are appropriate for creating deformation-based mechanics a more stable simulation model should be considered.

Future Work

The shapes used in the experiments are basic primitive shapes, but the techniques used here does allow for the creation of more complex character shapes. The addition of limbs, for example, can create characters representing creatures or humanoids. This does, however, mean that implementing the basic movement of the character may become more complex. References showed how skeleton animation can be combined with soft body deformation to create deformable characters. This work could be extended to create stretchable bones to allow external forces to alter the overall skeletal shape rather than just the deformation of the soft body skin. This would allow for more creative control in the movement of the character that is augmented by the elastic properties of the soft body rather than relying on procedurally generated animation by the soft body simulation alone.
Using alternate simulation models can not only help eliminate the stability issues but with a different set of physical properties can provide the potential for a more varied set of gameplay mechanics. Macklin et al. (2014) demonstrates soft bodies that interact with rigid bodies, fluid and gases in the same simulation model that could lead to some interesting mechanics where a character could change between these states.

CONCLUSIONS

This paper presents a set of game mechanics that are made possible by the use of soft body simulation to create deformable game characters. The findings showed that even when a somewhat limited set of mechanics, the gameplay prototypes produced demonstrate the potential for more varied and unique gameplay experiences for the player. The deformation behaviour of the character is fully dynamic and controllable. The use of volume regions within the simulation model allows for parts of the characters to deform differently. This was used to create a rigid core within the simulation model that controls the overall motion of the character without the need to couple with a rigid body system. This facilitates a method of control similar to rigid body character controllers that allow players to direct the overall motion of the characters in an intuitive manner while still being able to use the deformation properties of the character as a means to interact with the environment to achieve goals within a game. The topology of the tetrahedral mesh, the configurable properties of the underlying simulation model and the extent these properties can be affected in real-time directly impacts the range of gameplay mechanics possible.

In the current FEM-based implementation, plausible deformations are shown when the character is subjected to external forces, but the integrity of the model can be lost if the displacement of the internal elements of the simulation model is too large which negatively impacts the player experience. This can be overcome by exploring more robust simulation models that have different characteristics and hence lead to the creation of more deformation based game mechanics.

REFERENCES

Atari Games, 1984. *Marble Madness.* Atari Games [Vide Game].

Cryptic Sea, 2003. *Gish.* Chronic Logic [Video Game].

Last Quarter Studios, 2015. *Cats Are Liquid.* Last Quarter Studios [Video Game].

Liu L.; Yin K.; Wang B.; and Guo B., 2013. *Simulation and Control of Skeleton-Driven Soft Body Characters. ACM Transactions on Graphics*, 32, no. 6, 1–8. ISSN 07300301. doi:10.1145/2508363.2508427.

Macklin M.; Müller M.; Chentanez N.; and Kim T.Y., 2014. *Unified Particle Physics for Real-Time Applications. ACM Transactions on Graphics*, 33, no. 4, 1–12. ISSN 07300301. doi:10.1145/2601097.2601152.

Müller M. and Gross M., 2004. *Interactive Virtual Materials.* In *Proceedings of Graphics Interface 2004*. Canadian Human-Computer Communications Society, 239–246.

Müller M.; Heidelberger B.; Hennix M.; and Ratcliff J., 2007. *Position Based Dynamics. Journal of Visual Communication and Image Representation*, 18, no. 2, 109–118. ISSN 10473203. doi:10.1016/j.jvcir.2007.01.005.

Parker E.G. and O'Brien J.F., 2009. *Real-Time Deformation and Fracture in a Game Environment.* In *Proceedings of the 2009 ACM SIGGRAPH/Eurographics Symposium on Computer Animation.* ACM, 165–175.

Ready At Dawn, 2017. *Deformers.* GameTrust [Video Game].

Rovio Entertainment, 2009. *Angry Birds.* Chillingo [Video Game].

Second Order, 2017. *Claybook.* Second Order [Video Game].

Tan J.; Turk G.; and Liu C.K., 2012. *Soft Body Locomotion. ACM Transactions on Graphics (TOG)*, 31, no. 4, 26.

WEB REFERENCES

2008. *Unity.* https://unity3d.com. [Accessed June 2018].

BIOGRAPHY

Grant Clarke is a lecturer in Computer Games Development and programme leader of the Computer Games Technology degree course at Abertay University.

AUGMENTED PLAY

Beyond Pokemon Go
Advances in Augmented Reality for Games

Ken Newman
Tina Gates
Bilal Ahmed
Diviision of CISAC
Higher Colleges of Technology
United Arab Emirates
E-mail: knewman@hct.ac.ae

KEYWORDS
Augmented Reality, Virtual Reality, Mixed Reality, Tango

ABSTRACT

2016 saw the release of Pokemon Go, an Augmented Reality (AR) Game built on Smart Phone Location Services. A new generation of mobile devices is emerging with spatial awareness capability provided by motion tracking, area learning, and depth perception. The first of these new devices are the Tango-enabled phones from Lenovo and Google. This paper reviews the capabilities of these new devices in terms of the traditional AR problems and reviews the current and next generation of AR Games. We choose to use the term Augmented Reality reather than Mixed Reality in this paper. The reasons for this are discussed in Section 4.

POKEMON GO AND LOCATION SERVICES

In 2016 Pokemon Go became for a short time a global phenomenon and introduced a large number of people to the idea of location-based Augmented Reality gaming. The origins of location-based games are found in the 1990s as handheld GPS systems became commercially available to the general public. Early location based game-like activities such as Geo-caching 5/14/17 7:47:00 AM were practiced by enthusiasts with handheld GPS devices but never became mainstream. From 2000 onwards 'location services' have become standard on Smart Phones expanding on the traditional GPS functionality. Location services use triangulation from three separate sources for determining location.

1. Phone towers – accurate to about 100m.
2. WiFi hubs – accurate to 15m.
3. GPS Satelites – accurate to about 2 m.

Source ("Google Location Services Accuracy.")

Pokemon Go players in outdoor locations using multiple devices could expect to see the same Pokemon character on their displays within about 2-20m of each other in real world measurement. This is because the Pokemons are not registered to their real environment with a marker such as a QR Code marker. They use only the location service and hence the registration limits of the location services. The Pokemon graphic is simply placed into the center of the viewport once the device is in the approximate location so they will appear as an obvious video overlay with no depth awareness - no object passing in front of it is able to obscure (occlude) it - demonstrated in figure 1.

It is a limitation of the location services of the current Smart Phones that there is no spatial awareness and therefore occlusion and tight video registration are simply not possible. Some researchers even draw a distinction between location-based games and 'real' AR games (Perlin, K. in Greenemeier, 2016).

Figure 1: Pokemon Go – no Occlussion

This limitation has been at the core of the Augmented Reality research from the beginning. In Section 2 we will go back to one of the seminal papers of AR research (Azuma, 1997) to review his summary of the problems of Augmented Reality and see how little progress was made on these problems until 2015.

In 2015 Google entered the AR space with the release of their first Tango device and SDK (Araújo et al., 2016). In section 3 we look at the Tango platform and ask "which of Azumas (1997) AR problems have now been solved?"

Tango enabled devices entered the consumer Market in 2016 with a first wave of games and apps. In Section 4 we review what we have seen so far, Tango games in development and we review the potential of the new capabilities in spatial awareness for AR games and other applications.

THE HISTORIC PROBLEMS OF AR

In 1997 Ronald T. Azuma first articulated the problems of Augemented Reality when he published an early survey on the field (Azuma, 1997) – a survey that has been one of the most cited papers in the AR field over the last 20 years. The intent of his 1997 paper was to describe the applications in which to integrate 3-D virtual objects into a 3-D real world environment placing the viewer in a realistic context. Azuma's study consisted of understanding the dynamics behind the field of augmented reality and described many of the challenges and problems of integrating the technology (Behringer, 1999). Many researchers have faced the challenges of analyzing the principles contrasting the variations between augmented reality and virtual environments (Nincarean, Alia, Halim, & Rahman, 2013). Through these dialogues and explorations, the research has created windows of opportunities to enhance technologies but progress has been slow.

The central problems of AR, registration and sensing errors, have until very recently remained unsolved as evident with other researchers whose concerns addressed the capability of the proper alignment of real objects and virtual objects to merge two worlds as coexisting objects (Ohta & Tamura, 1999). There are complexities with these problems as the user will experience errors as the system is unable to accurately perceive the dynamic environment. Without the capability of good registration, errors create a "visual – visual" conflict which does not mimic depth of perception and connection with cognitive ability to re-create the model world existing within augmented reality. Augmented reality systems must also be able to respond within milliseconds as the user's point of view changes (Azuma, 1997). Objects can be added or removed from the real environment; however, achieving believable realism is a challenege. AR systems were also deficient at early stages in applying all senses especially inclusiveness of sound (Azuma, 1997). augmented systems need to be able to detect incoming sound as to increase the stimuli and sensing response to the altered reality.

To summarize the five AR problems as articulated by Azuma (1997):

1. Realism: Merging photo realistic 3D objects with real environments. Realism here includes texture, lighting and shadow.

2. Registration - Registration errors must be kept to a fraction of a degree 0.2°– 0.5°.

3. Tracking – tracking the position and orientation of the viewport in 3D space – objects swim around and lag behind.

4. Portability – the ability of the viewer to move within a real environment

5. View parameters (field of view, depth of field, orientation, inter-pupilary distance, occlusion)

Other researchers pointed out, building on Azuma's study, AR systems lacked the ability to visually track or denote true eye location (Takagi et. al, 2000). The stimulus of touch nor the reflexes to feel the real forces of the environment presented a cue to understand location of a virtual object. This is consistent with Azuma's study distinguishing that the user's perception of movement is limited and needs to amend to adapting tracking methodologies within various environmental applications.

GOOGLE TANGO: WHAT GOT SOLVED?

Augmented Reality literature in the last thirty years has consistently articulated the unsolved problems of AR registration and sensing (Azuma, 2002). With the recent release of the first commercial devices incorporating Google Tango it seems as though some of ARs perennial problems have not only been suddenly solved but the solution has left the experimental labs and become available to a mass market. This section takes a look at Tango in terms of ARs big problems and asks what has been solved and what is left?

The Tango area-learning capability builds on the SLAM problem which has emerged from research in the autonomous vehicle field. The Simultaneous Localization and Map-building (SLAM) problem asks the question "is it possible for an autonomous vehicle to start in an unknown location in an unknown environment and then to incrementally build a map of this environment while simultaneously using this map to compute absolute vehicle location?"(Dissanayake et al, 2001, p. 229)

The SLAM problem in AR viewports, as opposed to vehicles, contains additional complexity as the device must track it's own orientation as well as movement relative to an origin point in a 3D coordinate system. This orientation is usually represented by a four part quaternion (x,y,z,w) and often refered to as six degrees of freedom as shown in figure 2. A car by comparison has only one axis of rotation (**Wy**) and two axes of movements (**x, z**), hence three degrees of freedom. Tango devices use visual odometry (analyzing variations in infrared camera images over time) supplemented by the device's gyroscope, compasss and accelerometer sensors. The result is a platform for registering AR objects to a real environmental point that has not been possible before on a mobile device.

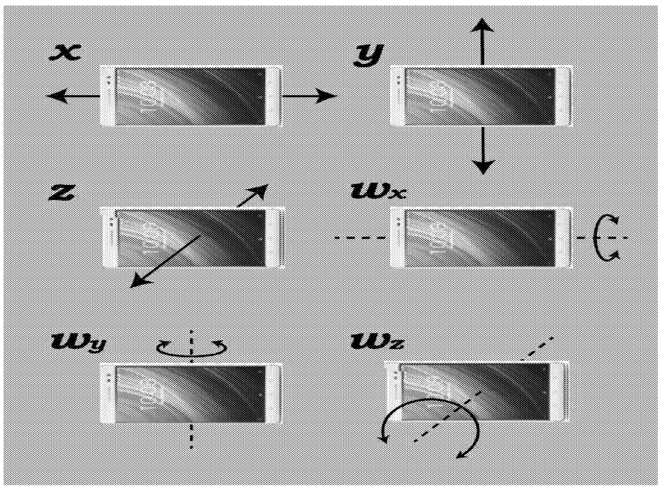


Figure 2: Device Six Degrees of Freedom

The limits of Tango motion tracking, area learning and depth-sensing accuracy have been tested in Roberto et al (2016). They found average Tango registration errors of 3-8cm for distance of 1-1.5m which translates to between 2°-5° of vision. The Roberto tests were done with the first generation Tango tablet. The authors replicated the tests with the Lenovo Phab 2 Tango enabled phone and found sensing errors improved - around 0.3cm at 1.5m distance.

While this may not still be accurate enough for surgical AR applications it is sufficient for many new visualization and gaming applications and it greatly improves the VR object registration capability offered by location services. Google Tango attempts to give the mobile devices spatial awareness similar to human perception of space and movement, and tries to unify how people and devices view the space around them.

As previously noted, Google Tango works on 3 major principles of Motion Tracking, Depth Perception, and Area Learning. On the software side, the required Tango APIs are available when Tango-enabled devices are connected.

Depth Sensing

Depth sensing techniques generate point cloud data from returned infrared light, which enables a device to differentiate real objects and surfaces in the real environment. This facilitates placing virtual objects inside real-world environment in a more natural manner.

Occlusion

Occlusion happens when a real world object is present between the virtual object and the users' viewport. Google Tango uses area mapping (meshing) to intelligently place the real world object between the virtual object by detecting the edges of the virtual object and the real object in a seamless manner. It performs this functionality using depth sensing to see the world in 3D and mapping the environment to know where the real object is present compared to the virtual object.

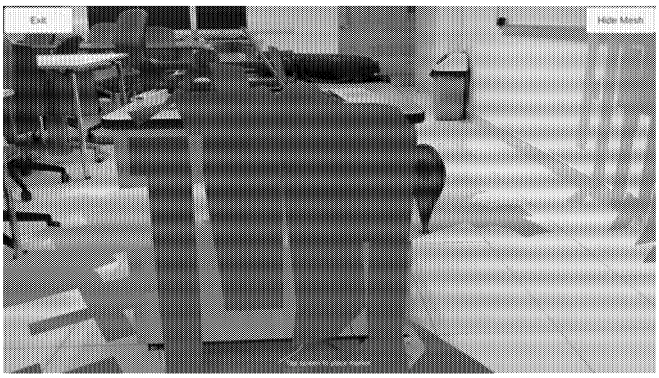


Figure 3: Tango area mapping (meshing)

Meshing

Tango meshes the real environment by bouncing inra-red light into the environment to create a point cloud and mapping those as texture tiles onto a virtual 3D mesh of the environment. It still cannot perfectly scan the whole environment, few mesh holes can be expected in the areas which are very close to each other and where it cannot determine the shape of the object due to poor light, poor contrast or its proximity to another object. In figure 3 the red virtual marker is partially occluded by the desk with the help of the area mesh generated at run time. In this example the mesh is left visible to demonstrate that though it is far from complete, it is sufficient to generate the occlusion image for the scene.

All this works well with still room environments. If there are moving objects in the environment, then the results are not meant for meshing, but those tracking reults still can be used for mapping cluttered environments.

To summarize the progress of the Tango platform in terms of Azuma's AR problems.

1. Realism

3D modelling, texturing and lighting have come a long way since Azuma articulated this problem in 1997. While there is still work to be done in dynamically matching lighting to real world environments some work has been done with casting shadows. In Figure 4 Tröster (2017) demonstrates how a virtual object can throw a shadow which follows real world contours using a pre-loaded Tango area-mesh and adding a Unity 3D shadow fragment shader

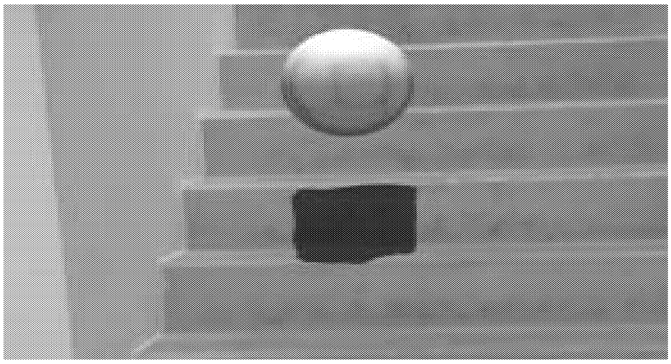


Figure 4: Realistic Shadows

2. Registration

The previously mentioned Roberto study (Roberto et al., 2016) found registration errors of between 2°-5° of vision. Azuma (1997) set a registration target of 0.2°– 0.5°. The authors own tests using a Lenovo Tango-enabled Phab 2 device show considerably better results than the Roberto study, although this needs to be tested further. Figure 5 shows a measure of 30cm for an A4 page of 29.7cm. This represents a registration error of > 0.2°.

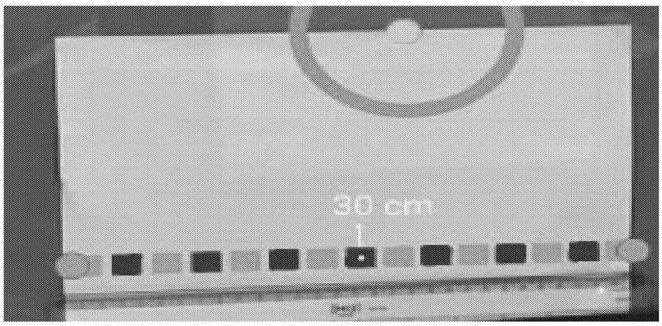


Figure 5: Registration Error > 0.2° vision

3. Tracking

Various shake tests have been used in eye-tracker research (Mardanbegi, Hansen, & Pederson, 2012). For this study we devised a simple shake test consisting of four shakes through a minimum of 30° in 1 second. The video of the shake test was analyzed to find the Maximum track error as shown in figure 6. In the this case the error was 3cm at a distance to the object of 1 m. This translastes to >2° of vision (Newman, 2017).

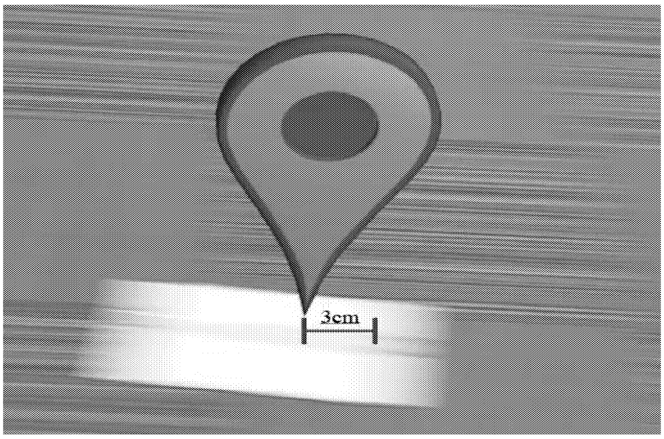


Figure 6: Tracking Shake Test

4. Portability

The VR/AR space is currently evolving between the moblie vs tethered platforms. Tango is the first implementation of a completely mobile spatially aware AR platform. The Tango area learning functionality is theoretically unlimited although as the device travels further from the point of origin the device's own calculation of it's position and orientation may have drifted from it's real location. By returning to the origin point or any previously seen location the Tango device will perform a 'drift correction' to improve the accuracy of the area map. Area maps may be saved and loaded in which case the Tango device will look for a familiar location and 'localize' (locate and orient) itself, to the origin point of the pre-loaded area map.

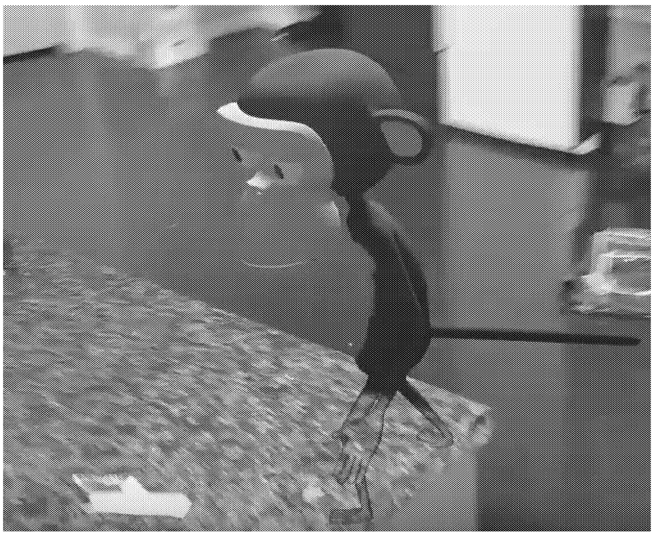


Figure 7: Occlusion With Tango

5. View parameters

Of all the view parameters identified by Azuma (field of view, depth of field, orientation, inter-pupilary distance, occlusion) occlusion remains the most important in terms of it's impact upon AR believability. Although still experimental occlusion is becoming a reality with the Tango Platform. In figure 7 the virtual monkey is partially occluded by the bench using a pre-loaded area mesh (Petrie, 2017).

TANGO GAMES AND THE VR/AR/MR SPACE

Since the Tango platform was first demonstrated in 2015 (Lee, 2015) a number of notable Tango games and other applications have been released. In this section we will broaden the view to consider how the new spatial awareness technology is developing and how it is being perceived by developers of entertainment and other commercial applications.

The first Tango game "Woorld" was released in September 2016 and at the time of writing in May 2017, there are 34 Tango apps listed in the Google Playstore. Figure 8 shows one of these games: "Towers for Tango" an AR version of the old SimCity genre.

Figure 8: Towers for Tango

Tango makes it easy to play AR games. There is no need for AR trackers or markers to place throughout the play space to help orient the viewing device. Instead, the device always knows where it is in relation to the AR objects. Because Tango devices have spatial awareness and know about the game play area, enemies can shoot directly at the player, and players have to physically move in order to change their location or dodge enemy attacks.

Other Tango applications to date include visualization apps for furniture companies. The user can load virtual furniture 3D models into their room environments in order to take a look at those in more details before placing the final order. Another notable application of Tango are the museum and Real Estate AR tours. The Detroit Institute of Arts developed a Tango app called 'Lumin' - a mobile AR tour of the museum. Figure 9 shows a scene from Luma where a user can view an X-ray of an Egyptian sarcophagus.

There are various other notable developments in the new VR/AR/ MR (Mixed Reality) space and this is prabaly a good time to make an observation about the terminology. In this paper we have elected to use the term Augmented Reality (AR) rather than the term Mixed Reality (MR) which is gaining traction from Microsoft. The term MR tries to draw a line under the first generation of AR which in practice was mostly limited to pop up text panels or 3D objects registed to QR Codes. MR describes the new 3D object rendering and spatial awareness capabilities. Others, the authors included, feel that Microsoft is muddying the water with the term MR (Schwarz, 2017).

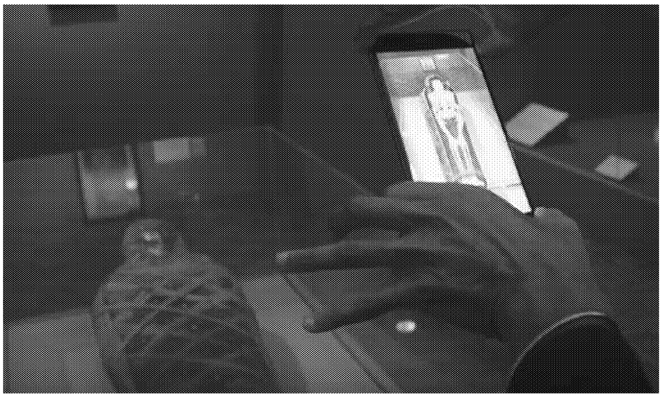


Figure 9: 'Lumin' AR Tour

They (we) prefer to see the new capabilities as part of a continuum that finally solves some of the problems that were clearly articulated from the earliest beginnings of the AR journey.

When VR/AR/MR is being discussed by business and economics forecasters they are often simply referred to as different 'flavours' of VR (Obrien, 2016) and this may be the best way to proceed as the traditional VR and AR technologies merge. In May 2016 Google announced the merger of the Tango AR and Daydream VR teams. The two technologies together integrates the whole VR 'All-flavours" spectrum and leads the way for low-end/mobile VR. On the high-end/tethered side of VR spectrum Occulus, Microsoft HoloLens, HTC Vive and Magic Leap are key players with proprietary platforms. The Superdata Research Report of the Virtuality Market 2016 (Superdata Research, 2016) reports VR and AR revenues for 2016 as $2.7 billion and $1 billion respectively and predicts 500% growth in these figures over the next four years. Tim Sweeny CEO of Epic (Unreal Game Engine) says,

"I would project that in six years or so, almost all of our designers will spend almost all of their time immersed in the VR editing environment." (Sweeny, 2017)

As Google Tango, Microsoft HoloLens, HTC Vive and Magic Leap all develop towards the next generation of spatially aware AR/VR/MR (Mixed Reality) technologies the future is loaded with possibility. In November 2016 the first standalone Virtual Reality Developers Conference took place with a focus on 'Mixed Reality' games focusing on the Google Tango and Microsoft Hololens platforms.

HCT Vive is leading the push towards enterprise applications in education, medicine and almost every conceivable form of remote control, but the future is far from clear. Alex Schwarz, VR/AR game developer and CEO of AR Games development company Owlchemy says,

"When I show people good VR, there's like 10, 20, 50 ideas of amazing things that could be built or industries that could be changed... AR just seems to be more of a blue ocean of possibility where people don't really even know what will be the form factor, the types of apps you would need or want," (Schwarz, 2017)

Welding simulators (e.g. figure 10) are an interesting case study on what makes a successful commercial AR training application. The requirement for registration and tracking accuracy for this simulation are already sufficient. To date these applications are based on QR Codes for their registration and it will be interesting to see if the new AR spatial awareness is adapted by these applications.

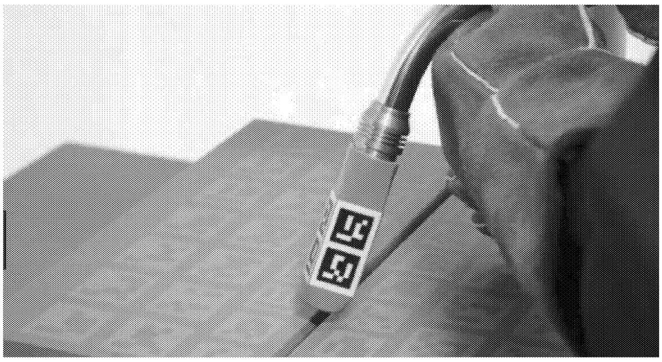


Figure 10: AR Welding simulator – QR Code Registration

With Smart Cities and integrated-everything systems increasingly becoming the norm, the case for VR/AR/MR remote control applications also becomes clearer. In 2016 DEWA, the Dubai Electricity and Water Authority announced it was using Microsift Hololens to visualize their Smart Power Plant, provide interactive 3D models for the plants' equipment, and remote expert assistance; including access to maintenance job cards, equipment, training manuals, and other operating procedures (DEWA, 2017).

As this paper is going to press four American Congress Representatives have announced a Congressional Caucus on Virtual, Augmented and Mixed Reality with a goal of supporting these technologies which have "shown tremendous potential for innovation in the fields of entertainment, education and healthcare" (Delbene, 2017).

SUMMARY

In this paper we have reviewed the development from the existing AR Games based on location services (Pokemon Go) to the new AR games and apps based on spatially aware technologies (Tango). We looked at the evolution of location-based games from geo-chaching to Pokemon Go and found the close interplay between these games and the enabling technologies, in particular the 'location services'.
We then looked at Augmented Reality and reviewed the Azuma's summary of the AR problems which have remained persistent and in many ways we see that failed previous attempts to produce AR and VR commercial products have

been premature because the Azuma problems have not been solved. Specifially we have reviewed Google Tango and examined to what extent it solves some of the persistent original problems of AR. Tango is of particular interest because it is a mobile/unthered technology. Rather than follow the route of the high-end/tethered platforms such as Occulus and HoloLens it trades the limitations of device processor/graphics/battery, for full 6 degrees of freedom in unlimited 3D space. We have done a short general review of the present state of Tango game and application development. We find a small (growing) number of games available and anticipate more as the number of Tango enabled consumer devices grows. Finally we took a wider look at the place of AR games and apps in the broad VR/AR/MR spectrum and found that the new spatial awareness technologies are driving a re-newed interest in this area.

As researchers we are watching the VR/AR/MR spectrum and Tango in particular with great interest and plan future studies to test other AR platforms and the user experience of the next wave of AR games and applications.

REFERENCES

Araújo, T., Roberto, R., Teixeira, J. M., Simões, F., Teichrieb, V., Lima, J. P., & Arruda, E. (2016). Life Cycle of a SLAM System: Implementation, Evaluation and Port to the Project Tango Device. In 2016 XVIII Symposium on Virtual and Augmented Reality (SVR) (pp. 10–19). https://doi.org/10.1109/SVR.2016.13

Azuma, R. T. (1997). A Survey of Augmented Reality. Presence, 6(4), 355–385. https://doi.org/10.1162/pres.1997.6.4.355

Azuma, R. T. (2002). Recent Advances in Augmented Reality. IEEE Computer Graphics and Applications, 21(6), 34–47. https://doi.org/10.1109/38.963459

Behringer, R. (1999). Registration for outdoor augmented reality applications using computer vision techniques and hybrid sensors. In Proceedings IEEE Virtual Reality (Cat. No. 99CB36316) (pp. 244–251). Los Alamitos. Calif. https://doi.org/10.1109/VR.1999.756958

Dissanayake, M. W. M. G., Newman, P., Clark, S., Durrant-Whyte, H. F., & Csorba, M. (2001). A solution to the simultaneous localization and map building (SLAM) problem. IEEE Transactions on Robotics and Automation, 17(3), 229–241. https://doi.org/10.1109/70.938381

Mardanbegi, D., Hansen, D. W., & Pederson, T. (2012). Eye-based Head Gestures. In Proceedings of the Symposium on Eye Tracking Research and Applications (pp. 139–146). New York, NY, USA: ACM. https://doi.org/10.1145/2168556.2168578

Nincarean, D., Alia, M. B., Halim, N. D. A., & Rahman, M. H. A. (2013). Mobile Augmented Reality: The Potential for Education. Procedia - Social and Behavioral Sciences, 103, 657–664. https://doi.org/10.1016/j.sbspro.2013.10.385

Obrien, J. (2016, May). The Race to Make Virtual Reality an Actual (Business) Reality. Fortune. Retrieved from http://fortune.com/virtual-reality-business/

Roberto, R., Lima, J. P., Araújo, T., & Teichrieb, V. (2016). Evaluation of Motion Tracking and Depth Sensing Accuracy of the Tango Tablet. In 2016 IEEE International Symposium on Mixed and Augmented Reality (ISMAR-Adjunct) (pp. 231–234). https://doi.org/10.1109/ISMAR-Adjunct.2016.0082

Sherman, E. (2004). Geocaching: Hike and Seek with Your GPS (Softcover reprint of the original 1st ed. edition). Berkeley, Calif: Apress.

WEB REFERENCES

Delbene, S. (2017, May 3). Reps. DelBene, Clarke, Flores, Issa and Lieu form Reality Caucus. Congresswoman Suzan DelBene. Retrieved May 4, 2017, from https://delbene.house.gov/media-center/press-releases/reps-delbene-clarke-flores-issa-and-lieu-form-reality-caucus

DEWA. (2017, January 7). DEWA adopts Microsoft HoloLens technology to enhance electricity and water services in Dubai. Retrieved May 7, 2017, from https://www.dewa.gov.ae/en/about-dewa/news-and-media/press-and-news/latest-news/2017/01/dewa-adopts-microsoft-hololens-technology-to-enhance-electricity-and-water-services-in-dubai

Google Location Services Accuracy." Android Forums. Accessed April 10, 2017. https://androidforums.com/threads/google-location-services-accuracy.924312/.

Greenemeier, L. (2016, July). Is Pokémon GO Really Augmented Reality? Scientific American, (July). Retrieved from https://www.scientificamerican.com/article/is-pokemon-go-really-augmented-reality/

Petrie, R. (2017). Google Tango Mesh Occlusion. Retrieved from https://www.youtube.com/watch?v=EpDhaM7ZhZs

Newman K (2017). Tracktest2. Retrieved from https://www.youtube.com/watch?v=070PRYaLh8E&feature=youtu.be

Lee, J. (2015). Google I/O 2015 - Project Tango - Mobile 3D tracking and perception. Retrieved from https://www.youtube.com/watch?v=iP9m9a2KEN4

Schwarz, A. (2017, May 1). VR exclusivity should be a last resort - Owlchemy. Retrieved from http://www.gamesindustry.biz/articles/2017-04-28-vr-exclusivity-should-be-a-last-resort-owlchemy

Steffen Tröster. (2017). Project Tango - Unity Mesh Builder Occlusion. Retrieved from https://www.youtube.com/watch?v=M8tiDMsmDyI

Superdata Research. (2016). Market Brief — Year in Review 2016 (Games data and market research). Superdata Research. Retrieved from https://www.superdataresearch.com/market-data/market-brief-year-in-review/

Sweeny, T. (2017, March). For Tim Sweeney, advancing Epic means racing into AR and VR. Retrieved from http://www.gamasutra.com/view/news/292573/For_Tim_Sweeney_advancing_Epic_means_racing_into_AR_and_VR.php

IMMERSION ISSUES IN HAPTIC EXPERIENCES

Helena Barbas
Faculdade de Ciências Sociais e Humanas – Universidade Nova de Lisboa
1069-061 Lisbon,
Portugal
E-mail: hebarbas@fcsh.unl.pt

KEYWORDS
Immersion; brain studies; special relativity theory; games; human computer interaction; digital humanities; theoretico-practical terminologies; haptics;

ABSTRACT

As a work in progress this paper intends to address actual immersion issues concerning mismatches in VR and Mixed Reality haptic experiences, in the relation between real/virtual space.

It considers the more recent developments in Brain Studies, Cognitive Sciences, and Physics and how these are affecting the concepts of Man, space, and human-space interaction.

The focus was on touch and haptic issues, namely mismatching of places and shapes.

In some examples these problems seem to be attributable to the resource to outdated theoretical practice frames (Euclidian-Newtonian), and the suggestion is that some of the issues could be solved by resourcing to Minkowki-Einsteinian space-time algorithms.

INTRODUCTION

This paper – as a work in progress – intends to address actual immersion issues concerning mismatches in VR and Mixed Reality haptic experiences, in the relation between real/virtual space.

It will initially address conceptual vocabulary problems, as the terminology used, adopted from other sciences, is theory-laden and in some cases obsolete.

New developments in Brain Studies, Cognitive Sciences, and Physics have affected the concepts of Man, space, and human-space interaction. So, and considering that Immersion presupposes a user/body with senses, acting/reacting to an environment (real or virtual), these concepts demand an updated re-definition in a new Minkowski-Einsteinian world.

Special Relativity theory replaces 3D movement by weaving patterns of complex 4D geometry, or sculpture, at the neural as well as the cosmic scale. Gravity as a force is abolished, to be replaced by pure geometry. Minkowski complex Geometry is already being used by game developers, namely for the creation of volumes, or movement.

The most recent Human Brain Projects are redesigning new brain atlas, redefining its characteristics, areas and functions.

The impact of all this new information is overwhelming. And as well as in the scientific fields, it will demand new research in all other human related areas.

Having in mind the new sensory/motor Homunculus map, the 5 senses will be quickly addressed to concentrate on Touch.

Considering the case-study experiences, the focus will be on the type of haptic gadgets used (wearable or external), resistance, mapping, as well as mismatches and warping.

In some examples these problems seem to be attributable to the resource to outdated theoretical practice frames (Euclidian-Newtonian), and the suggestion is that some of the issues could be solved by resource to Minkowki-Einsteinian space-time algorithms.

These haptic issues in the relation between user-body and the real or VR world need clarification in order to establish an operational basis that can become useful for further and posterior game, storytelling and cultural heritage practices.

CONCEPTUAL PROBLEMS

The word «**Immersion**», from the late 15th century, comes from the Latin *immersio*(*n*-), from *immergere* 'dip into'. Its senses are: 1. the action of immersing someone or something in a liquid; 2. deep mental involvement in something; or 3. (from Astronomy), the disappearance of a celestial body in the shadow of or behind another (O.E.D.).

The term has been updated and transformed to describe new experiences related to the internet world and virtual reality, and gave basis to multifarious definitions: the Saussurian "signifié"/signified acquired, anarchically, many "signifiants"/signifiers to encompass the needs of new realities in the computer world.

This option to return to dictionary definitions – here and in subsequent words/designations – is due to the fact that all these terms have been arbitrarily contaminated by the disciplines that chose to use them and became theory-laden – namely in/by a Euclidian-Newtonian universe – and so are also always right and always wrong.

The current definitions of **Immersion** – into a virtual reality – allude to the **perception** of being **physically** present in a non-physical **world**. This 'perception' is fashioned by surrounding the VR system **user** with **images**, **sounds** or other **senses stimuli** that provide an engrossing total **environment** (bold mine).

From the state-of-the art, either theoretical or practical, even the more recent speculations resort to specialists whose models are old. Immersion has been considered and related to the artistic experience (Frank Rose 2011), a feature of mental operations associated with narrative engagement (Marie Laure Ryan 1992) and go the par with the literary 'suspension of disbelief', a state of consciousness (Maurice Benayoun 2008) or separated into main practical categories (Ernest W. Adams, 2003/2012).

In all cases, the name-word is a metaphor used to describe a user (a body, a subject, a consciousness) in relation with a physical world (cosmos) experiencing (sensing / feeling) a relation with the environment - real or artificial - which can be more (or less) verisimilar.

All the definitions and terminologies – so far – have not considered the latest developments either in Brain Studies and Cognitive Sciences, Philosophical representation theories or Physics. They are operating in a Euclidian-Newtonian world ignoring the progress in Quantum Physics, disregarding Minkowski's Geometry, or Einstein's Special Relativity theory. The situation is more complex because in, and from, Games (making-of, mechanics or as an artefact) the Minkowski-Einstein theories and universe are already the rule.

All have as working material obsolete images of the world, of the human being, and so, of the human interaction with that world.

CONCERNING THE WORLD

«In spite of the fact that SR [Special Relativity] has been known for a century and Newtonian theory is only an approximation, all the theories of biology, evolution, and neuroscience today still remain resolutely rooted in the Newtonian System. For example, evolutionary theory today describes 3D organisms competing in a 3D space in a separate time and changing their structures dynamically as they do so. These organisms only appear to be moving because they are observed by the Observer on its time travel along their common time dimension. Whereas all these dynamic changes are described in SR as features of the static 4D sculpture of the world lines of these organisms.» (Smythies 2018:8).

Newtonian vs. Einsteinian worlds

The world is described by Newtonian theory as a collection of 3D objects extended in a 3D space. They exist in a separate time that flows independently.

Their movement in space during time is determined by Newton's laws of motion – gravity as an attractive force between objects.

The new 3D experiences – probably due to the influence of games – try to adapt the linear (1D) or 2D theories, with diverse results, as if 3D where an extension of both.

Concerning time and space, they root themselves on a Newtonian world, not considering the most recent theories about the universe 4D.

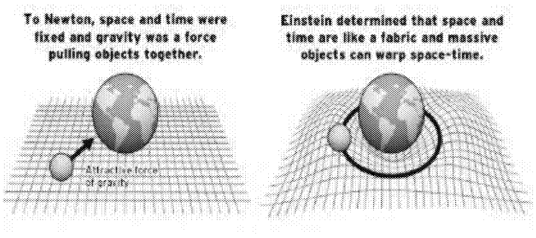


Figure 1: Newton vs. Einstein worlds

For Penrose: «what we perceive as moving 3D objects are really successive cross sections of immobile 4D objects past which our field of observation is sweeping.» (Smithies 2018:6).

Here, the Observer is not the physical body of the scientist, but a Self or Observer that can move in time in the way described.

SR replaces 3D movement by weaving patterns of complex 4D geometry, or sculpture, at the neural as well as the cosmic scale. Gravity as a force is abolished, to be replaced by pure geometry.

From Euclidean to Minkowski Geometry

In 3D space, the differential of distance (line element) is defined by coordinates that are the differentials of the three spatial dimensions.

By 1907 Minkowski, based on the previous work of Lorentz and Poincaré, and considering SR introduced by his former student Albert Einstein (1905), advanced that time and space are not separated entities but intermingled in a four dimensional space–time.

Minkowski space-time appears to be very similar to the standard 3D Euclidean space, but there is a crucial difference in regards to time.

In "Minkowski space-time" geometry there is an extra dimension (coordinate X0) derived from time, such that the distance differential fulfils where dX = (dX0, dX1, dX2, dX3) are the differentials of the four space-time dimensions.

This suggests a deep theoretical insight: special relativity is, analogous to the rotational symmetry of Euclidean space, a rotational symmetry of our space-time. Just as Euclidean space uses a Euclidean metric, so space-time uses a Minkowski metric:

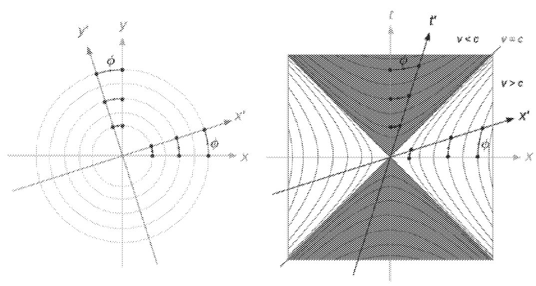


Figure 2: Orthogonality and rotation of coordinate systems compared - Maschen

Basically, SR can be stated as the invariance of any space-time interval (that is the 4D distance between any two events) when viewed from any inertial reference frame.

However, if this distance is transferred from a static to a dynamic perspective, it has to consider a mutual interference: the angled x' line in the image follows the line of simultaneity at event 0 for worldline t'.:

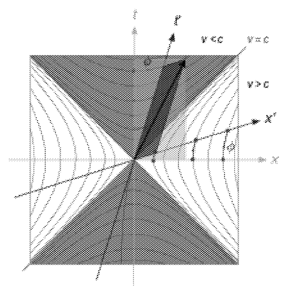


Figure 3: Wordline's variances - Tom Slijkerman

So, if the relation between two events/objects is considered from a mobile standpoint, it has to reflect the interference of the speed of the object in respect to the Observer, and that its own line of simultaneity is not horizontal, but is angled in the same but opposite angle as its worldline.

All equations and effects of SR can be derived from this rotational symmetry (the Poincaré group) of Minkowski space-time.

Minkowski complex Geometry is already being used by game developers, namely for the creation of volumes, or movement.

The Hyperbolic Games 2.0 4+ (2018 Jeff Weeks) – for PC and mobile apps – combine a multi-connected topology with a non-Euclidean geometry.

Mathematically they illustrate the hyperbolic plane, as a live scrollable object; the «under-appreciated fact that the two traditional models of the hyperbolic plane are simply different views of the same fixed-radius surface in Minkowski space…»

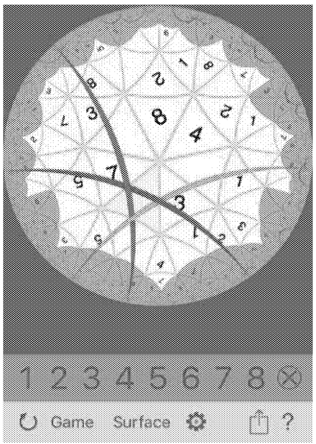


Figure 4: The Hyperbolic Games 2.0 - Jeff Weeks

Einsteinian worlds

Special Relativity uses a 'flat' 4-dimensional Minkowski space – an example of a space-time.

Organisms do not evolve in a block universe – rather the Observer sees successive cross sections of the organisms 4D physical structure that is simply more complex the further up the time dimension the Observers travel.

MIT Game Lab provides *OpenRelativity* (2012) an open-source Unity toolkit, designed to let developers integrate space-time-bending and accurately simulate SR effects of Einstein's special relativity, such as: «Lorentz contraction, time dilation, Doppler shift and the searchlight effect»:

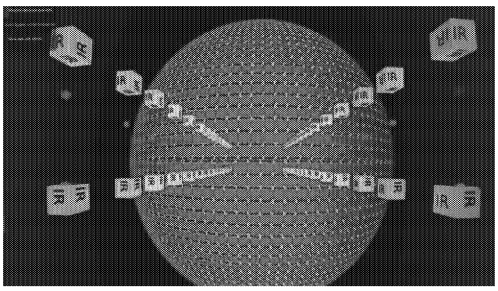


Figure 5: The IR/UV spectrum effect in OpenRelativity – MIT GameLab 2012.ewton vs. Einstein worlds

They also created two games with this software. *A Slower Speed of Light* (2013) – a first-person game prototype in which players navigate a 3D space while picking up orbs that reduce the speed of light in increments; and Einstein Playground (2016) – now a immersive SR VR experience at The Charles Hayden Planetarium.

Besides the incipient *Velocity Raptor* (TestTube 2011) there is available on STEAM since 2015 *Relativity Wars* – A Science Space RTS, a strategy/action game, set in a universe advertised as obeying Einstein's theories of relativity.

Figure 6: Relativity Wars - A Science Space RTS - FunGameCo

The need to consider the SR picture only comes in effect when the deeper questions as to the nature of reality are addressed (Smythies 2018:8): «This realization also requires radical changes in our ideas of what brain "events" actual consist of in a Minkowskian block universe».

Brain "events" are not composed of moving 3D atoms but of successive 3D cross sections of the static 4D world lines of these atoms. (Smithies 2018:8)

However, this does not require any changes in the practical day to day experimental neuroscience for which Newtonian terminology seems quite sufficient.

CONCERNING THE SUBJECTS BODY

– Brain Studies

Here the body will also be regarded in its etymological, dictionary terms: «1. The physical structure, including the bones, flesh, and organs, of a person or an animal.» from Old English *bodig*.

Among the organs, the brain will be considered.

The most recent studies – The Human Brain Project Home, the Brain Initiative, The Human Connectome Project, and The Functional Connectome Organization – are sketching a new brain atlas redefining its characteristics, areas and functions.

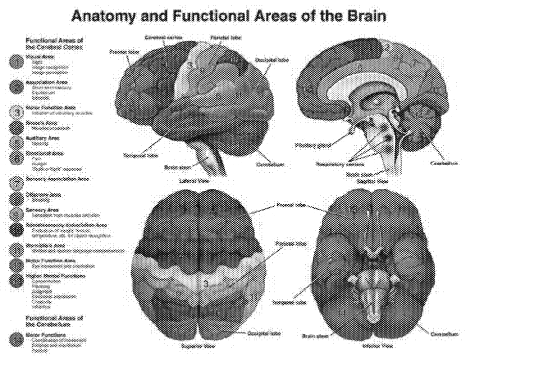


Figure 7: Anatomy and functional areas of the brain

To this now 'traditional' representation of the brain, an old known part (Sir Francis Crick; Christof Koch) – *The Claustrum* – has suddenly acquired new importance (Ramachandran 2013). It is a thin sheet of neurons laying between the insular cortex and the *striatum*, reciprocally connected with almost all cortical areas (motor, somatosensory, visual, auditory, limbic, associative, and prefrontal), presently proposed as the 'seat of consciousness':

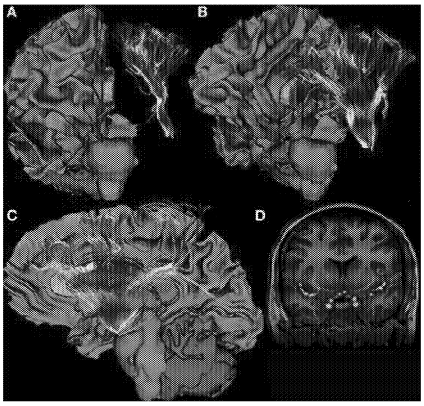


Figure 8: (A–C) Scanner showing white matter pathways emanating from the region of the human *claustrum*. (D) Binary label mask drawn of the left and right *claustra*.

Studies from the Functional Connectome Organization, based on single-scanner sampled 5216 UK Biobank participants, mapped «sex differences in brain volume, surface area, cortical thickness, diffusion parameters, and functional connectivity between adult males and females in the range between middle- and older-age.» (Ritchie 2018). Gender differences are shown in blue-pink:

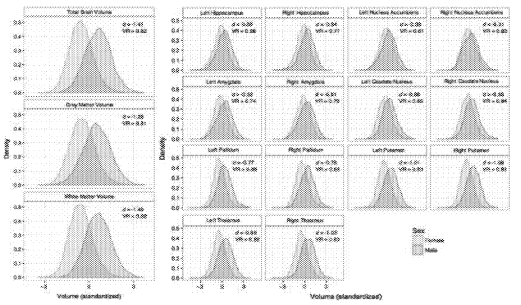


Figure 9: Sex Differences in the Adult Human Brain – Cereb Cortex

Discrepancies in Volume, Surface area and Cortical Thickness are shown below:

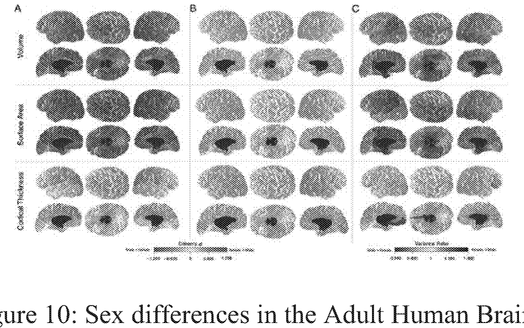


Figure 10: Sex differences in the Adult Human Brain – Cereb Cortex

These gender differences are biological, and consequently will have functional consequences at neurodevelopment.

This study is accompanied by cognitive tests at a very rudimentary level (Verbal-numerical reasoning/fluid intelligence; Reaction Time - modelled on the game of 'snap').

The authors explain: «Functional connectome organization showed stronger connectivity for males in unimodal sensorimotor cortices, and stronger connectivity for females in the default mode network. This large-scale study provides a foundation for attempts to understand the causes and consequences of gender differences in adult brain structure and function». And conclude: «providing a clear characterization of neurobiological sex differences is a step towards understanding patterns of differential prevalence in neurodevelopmental disorders» (Richtie 2018).

The impact of this all new information is overwhelming. And as in Sciences fields, will demand new research in all the human related areas.

Experiences have been made mapping the cerebral zones reacting to words. There is a Brain dictionary – (https://www.youtube.com/watch?time_continue=1&v=k61nJkx5aDQ) and IMR scans show the different areas awakened by game-playing, or during storytelling:

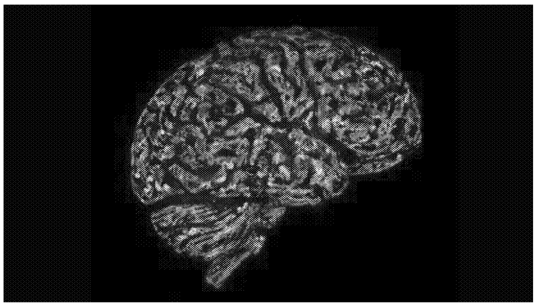


Figure 11: Brain illuminated during Storytelling

Theoretical neuroscientists are working to develop a multi-scale theory of the brain that synthesizes top-down and data-driven bottom-up approaches.

CONCERNING THE SUBJECTS BODY

– consciousness – perception – feelings - sensations

These areas are beings studied by Cognitive Sciences, Neuro philosophy, with admirable new results and due to the vastness of their implications – either for game-playing either any kind of storytelling or cultural heritage – are being considered for a future work.

Within that area are included arguments about perception, its controversies about 'direct' and 'indirect realism' – either if the immediate objects of perception are distal physical objects (as in naturalistic metaphysics, common sense and ordinary language) or if conscious experiences consist of reconstructions from information encoded in neural states and is hence indirect (a brain computation - neuroscientists).

– the five senses

As stated above, body will be considered in its etymological sense. This option to return to the dictionary is also due to the contamination the term has suffered either from Biology, Philosophy, New Media, Interactive art or Post-humanities studies (i.e., culturally constructed body; embodiment; body-image; body-schema; relational-process).

The issue with these classifications and ontologies is the same as with Immersion – both are based on out-of-date theories, and either consider the body as a robot, or a relational process.

Here, the brain will be addressed, among the organs.

The Sensory-Motor Homunculus

For the moment, what can directly affect kinetic and haptic experiences has to do with the so called Sensory/Motor Homunculus, a new atlas of the relationships between body and brain:

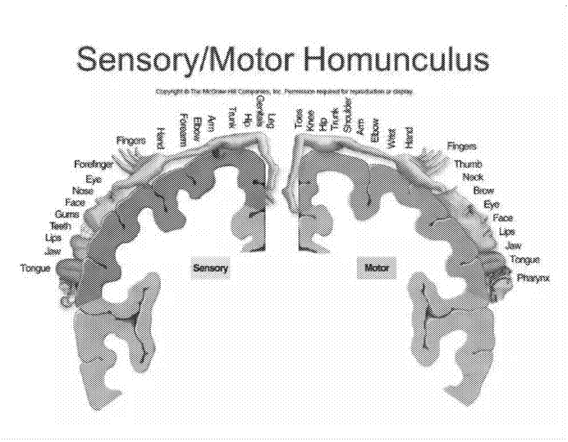


Figure 12: The sensory Motor Homunculus

This new mapping of the relationships between body and brain shows the bi-partition of activities – sensory (feeling) /motor (action) – but more importantly, exhibits considerable difference in intensity, a hierarchy of the prominence that each of the several limbs and parts of the human body have in brain activity.

– some of the five senses issues

Sense – Is a faculty by which the body perceives an external stimulus; one of the faculties of sight, smell, hearing, taste, and touch. Origin Late Middle English (as a noun in the sense 'meaning'): from Latin *sensus* 'faculty of feeling, thought, meaning', from *sentire* 'feel'. The verb dates from the mid 16th century.

The traditional definition of the 'five senses' – bearing in mind the Homunculus – might also need a readjustment, but will here be considered as is.

– Sight

The main mechanism of visual brain is information compression – the incoming picture via retina is divided into successive frames, first in the lower visual cortex (V1), then superimposed and transmitted to the higher visual cortex – saving computational resources.

The picture is built from information provided directly by the retina together with the memories stored in the higher visual cortex (Smithie 2018: 2)

Recent studies show that vision is dispersed by at least three different brain areas. So, the processing of colour, shape and motion of visual stimuli is carried out in quite different locations, and at different speeds – colour being the fastest, followed by shape and motion. Somehow at the highest level all these separate computations are amalgamated into the unified visual object that we see.

Visual neurons, when stimulated into a conscious visual experience, can be changed by epigenetic manipulations into functioning auditory neurons – resulting in a conscious auditory experience.

The same neurons, stimulated by differently ingresses, can give different conscious visual experiences (i.e. geometrical patterns, or oily swirls).

The new brain research and neuroscience theories are studying vision quite in depth. They show that the act of perceiving visually is a complex composite computation.

– Hearing

Concerning this sense, the only useful remark here is the existence of music-colour synesthesia, and that there are visual stimuli that can be changed into hearing experiences, and vice-versa. (Curwen 2018:96).

– Taste and smell

Taste and/or smell have been subject to some avant-garde artistic experiences – namely Tate Gallery (London), Robotarium (Bruges), etc., but are not relevant for this study.

– Touch – Haptic

Haptic perception (Greek: "suitable for touch") literally means the ability "to grasp something". Awareness, in this case, is achieved through the active exploration of surfaces and objects by a moving subject, as opposed to passive contact by a static subject during tactile perception (Hackfelt 2018).

The term Haptic was coined by the German Psychologist Max Dessoir in 1892, when suggesting a name for academic research into the sense of touch in the style of that in "acoustics" and "optics" (Hackfelt 2018).

Gibson (1966) defined the haptic system as «the sensibility of the individual to the world adjacent to his body by use of his body». Gibson and others further emphasized what Weber had already realized in 1851: the close link between haptic perception and body movement, and that haptic perception is active exploration.

Haptic or kinesthetic communication recreates the sense of touch by applying forces, vibrations, or motions to the user – directly or indirectly

CONCERNING THE SUBJECTS' BODY

– the relationship with space

So, leaving perception issues for future work, we will be addressed Proprioception

Proprioception - from Latin *proprius*, meaning "one's own", "individual", and *capio*, *capere*, to take or grasp, is the sense of the relative position of one's own parts of the body and strength of effort being employed in movement [3].

It is the ability to sense stimuli arising within the body regarding position, motion, and equilibrium.

Even if a person is blindfolded, he or she knows through proprioception if an arm is above the head or hanging by the side of the body. The sense of proprioception is disturbed in many neurological disorders.

In humans, it is provided by proprioceptors in skeletal striated muscles (muscle spindles) and tendons (Golgi tendon organ) and the fibrous capsules in joints.

It is distinguished from exteroception, by which one perceives the outside world, and interoception, by which one perceives pain, hunger, etc., and the movement of internal organs. This suggests that while these components may well be related in a cognitive manner, they may in fact be physiologically separate. (Hakfelt, A. 2018).

Haptic technology

Under this category can be included all types of gadgets that enhance/replace bodily functions, either as a prosthesis directly applied to the human body, or as an external accessory – wearable, or belonging to the environment. The former can include head mounted displays, gloves, coats or shoes, i.e.. The latter can include objects providing haptic EMS effects as walls, gates, sliders, boxes, and projectiles.

Here as well are being created ontologies of haptic systems and controllers, i.e. considered 'active', 'passive' or 'Encounter-type' (Zhao 2018) – but all gadgets have to be active and deliver some kind of interaction and feedback. Either wearable or external, static or dynamical, they can provide muscular stimulation to create an effort / resistance effect – which can go from a single point kinesthesia to larger parts of the body.

These gadgets are employed to allow the human body to interact with Augmented, Mixed or Virtual Reality environments (Raptis 2018). Their aim is said to be to fool the human 5 senses (sight, sound, touch, smell, taste) by creating a sensation of Presence or Full Immersion – that would allow the user to perceive the digital environment as being physically real.

This perspective is ignoring two elements that are vital for any good VR experience – one is the adoption/embodiment of the position of the Omniscient Narrator, the other the aesthetic factor. As an example of these two features, that transport the VR experience to another 'brain' level not yet systematized is the film #DoWhatYouCant Samsung Ostrich Commercial (2017) - https://www.youtube.com/watch?v=wdL3zfxzueQ

Figure 13: #DoWhatYouCant Samsung Ostrich Commercial (2017)

Besides these, and as a unique intentionally aesthetical experience, created by Studio Swine in Milan during the 2017 international design week, is *New Spring*. It consists of a tree-like sculpture in the centre of a dark room, featuring cascading, scented, mist-filled blossoms that burst and evaporate upon contact with the skin, but live for a few moments when met with textured fabrics (http://edition.cnn.com/style/article/milan-design-week-cos-studio-swine/index.html).

Mismatching

In more down to earth experiences of haptic props there are issues in attaining a perfect coincidence of places and shapes between the real and virtual object – mismatches and warping – «which may lead to poorer sense of presence» and, what can be really important: «to decreased manipulation performance during the interaction» (Zhao 2018).

Mismatching can lead to loss of verisimilitude. It can happen in shape, alignment, and speed of action. Its consequences are space warping.

This issue has demanded to be approached by the full 3D mapping of one, or pairs, of complex surfaces in 3D spaces. The original legend to the image below says: «We present a new optimization based technique for 3D haptic retargeting of complex shapes. From left to right, a) A user interacts with a tracked physical prop (a coffee mug) while wearing an HMD, b) the virtual view of the user, showing a tea cup and retargeted hand position, c) the outline of the hand shows its position in the physical world, and the transparent coffee mug shows the shape and position of the real physical prop super imposed over the user's view, d) a 2D example of our method which shows how space is warped to retarget a square to a circle and move the position and orientation of another rectangle. The blue vectors represent the direction and magnitude of the spatial warping.» (Zhao 2018):

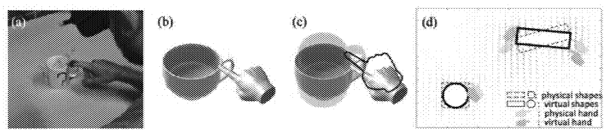


Figure 14: 3D haptic retargeting of complex shapes

Mismatching consequences are space warping and pseudo-haptic feedback, approached by the full 3D mapping of one or pairs of complex surfaces in 3D spaces – in a Newtonian world. Probably some of the above issues could be solved by recourse to Minkowki-Einsteinian space-time algorithms and avoid retargeting needs. More so due to the fact that the magnitude of the 3D spatial warping angle seems – just from an optical perspective – very similar to the 4D dynamic representation of wordline variances as in Figure 3.

Visual Dominance in Haptic experiences

When there is a conflict between an observers' sense of vision and touch, vision becomes dominant. In the experiences the users «preferred to think that the object was most similar to the distorted visual image, rather than the actual physical shape that they felt» – this is an issue that will have to be addressed via Brain Studies.

Mastering tactile interaction at close range is the first step to advance to more complex haptic interfaces – holograms, distant objects – in order to widen its applications in storytelling, gaming, cultural heritage, manufacturing, medical, and other industries.

CONCLUSION

The newest advances in Brain Studies and constellation areas are imposing a paradigm change in theory and practice, and affecting the concepts of man, space, human-space and human-computer interactions – to be updated and made more conform to a Minkowski-Einsteinian world.

The aim of this paper – a work in progress – was to hierarchize the principal issues to be addressed and establish some guidelines for future work.

The focus was on touch and haptic issues – probably the easiest of the 'senses' to be addressed in such a context. The dis-connections between gadget users and AR/VR have to be disentangled without human help or interference from the observer.

To solve mismatching and warping issues without the presence of the physical body of the Observer can contribute to the development of wide spread everyday use gadgets – from mobiles, oculus, prosthetic limbs or devices, i.e. – to probably future more complex applications spreading from cultural heritage tours, augmented training, to no-manned apparatus.

REFERENCES

B Arnaldi B, Cotin S, Couture N, et al (2018) New Applications. In: Virtual Reality and Augmented Reality. John Wiley & Sons, Inc., Hoboken, NJ, USA, pp 1–71

Baños RM, Botella C, Alcañiz M, et al (2004) Immersion and Emotion: Their Impact on the Sense of Presence. CyberPsychology Behav 7:734–741. doi: 10.1089/cpb.2004.7.734

Beery A K, Zucker I (2011) Sex bias in neuroscience and biomedical research. Neurosci. Biobehav. Rev. 35:565–572.

Bell A, Ensslin A, Bom I van der, Smith J (2018) Immersion in Digital Fiction. Int J Lit Linguist. doi: 10.15462/IJLL.V7I1.105

Berken A (2015) Focus: For 100th anniversary of Einstein's theory of relativity, why it still matters today – Orange County Register. https://www.ocregister.com/2015/11/29/focus-for-100th-anniversary-of-einsteins-theory-of-relativity-why-it-still-matters-today/. Accessed 10 Jun 2018

Biswal B B, Mennes M, Zuo X-N, et al (2010) Toward discovery science of human brain function. Proc Natl Acad Sci 107:4734–4739. doi: 10.1073/pnas.0911855107

Bowman D A, Mcmahan RP, Tech V (2007) Virtual Reality: How Much Immersion Is Enough? (Cover story). Computer (Long Beach Calif) 40:36–43. doi: 10.1109/MC.2007.257

Burgaleta M, Head K, Álvarez-Linera J, et al (2012) Sex differences in brain volume are related to specific skills, not to general intelligence. Intelligence 40:60–68. doi: 10.1016/j.intell.2011.10.006

Carr D, Bossomaier T, Lodge K (2007) Designing a computer game to teach Einstein's theory of relativity. In: Computer Graphics, Imaging and Visualisation: New Advances, CGIV 2007. pp 109–114

Cheng L-P, Marwecki S, Baudisch P (2017) Mutual Human Actuation. Proc 30th Annu ACM Symp User Interface Softw Technol - UIST '17 797–805. doi: 10.1145/3126594.3126667

Cruz-Neira C, Fernández M, Portalés C (2018) Virtual Reality and Games. Multimodal Technol Interact 2:8. doi: 10.3390/mti2010008

Curran N (2017) Factors of Immersion. Wiley Handb Hum Comput Interact 1:239–254. doi: 10.1002/9781118976005.ch13

Curwen C (2018) Music-colour synaesthesia: Concept, context and qualia. Conscious Cogn 61:94–106. doi: 10.1016/j.concog.2018.04.005

Elmenzeny A, Edenhofer N, Wimmer J (2018) Immersive storytelling in 360-degree videos: An analysis of interplay between narrative and technical immersion. J Virtual Worlds Res 11:1–13. doi: 10.4101/jvwr.v11i1.7298

Feix T, Romero J, Schmiedmayer HB, et al (2016) The GRASP Taxonomy of Human Grasp Types. IEEE Trans Human-Machine Syst 46:66–77. doi: 10.1109/THMS.2015.2470657

Formosa N J, Morrison B W, Hill G, Stone D (2018) Testing the efficacy of a virtual reality-based simulation in enhancing users' knowledge, attitudes, and empathy relating to psychosis. Aust J Psychol 70:57–65. doi: 10.1111/ajpy.12167

Goll Y, Atlan G, Citri A (2015) Attention: The claustrum. Trends Neurosci. 38:486–495.

Hakalisto S (2017) Immersion In Interactive Documentaries : A Game-Studies-Driven Approach In A Case Study Of Bear 71.

Hakfelt A Haptics (2018) The power of touch in immersive storytelling - Ultrahaptics. https://www.ultrahaptics.com/news/blog/haptics-storytelling/. Accessed 12 May 2018

Hoff van, Arthur (2017) Virtual Reality and the Future of Immersive Entertainment. Proc 2017 ACM Int Conf Interact Exp TV Online Video 129–129. doi: 10.1145/3077548.3080529

Howell E (2017) Einstein's Theory of Special Relativity. In: Space.com. https://www.space.com/36273-theory-special-relativity.html. Accessed 9 Jun 2018

Joel D, Berman Z, Tavor I, et al (2015) Sex beyond the genitalia: The human brain mosaic. Proc Natl Acad Sci 112:15468–15473. doi: 10.1073/pnas.1509654112

Kim T, Cooperstock JR (2018) Enhanced Pressure-Based Multimodal Immersive Experiences. In: Proceedings of the 9th Augmented Human International Conference on - AH '18. ACM Press, New York, New York, USA, pp 1–3

Landsberg P T (1968) Special theory of relativity. Nature 220:1182–1183. doi: 10.1038/2201182a0

Lopes P, You S, Cheng L-P, et al (2017) Providing Haptics to Walls & Heavy Objects in Virtual Reality by Means of Electrical Muscle Stimulation. In: Proceedings of the 2017 CHI Conference on Human Factors in Computing Systems - CHI '17. ACM Press, New York, New York, USA, pp 1471–1482

Martin-Brevet S, Jarrassé N, Burdet E, Roby-Brami A (2017) Taxonomy based analysis of force exchanges during object grasping and manipulation. PLoS One 12:e0178185. doi: 10.1371/journal.pone.0178185

McGrath D, Wegener M, McIntyre TJ, et al (2009) Student experiences of virtual reality - a case study in learning special relativity. Am J Phys 78:862–868. doi: 10.1119/1.3431565

Mendes D, Caputo FM, Giachetti A, et al (2018) A Survey on 3D Virtual Object Manipulation: From the Desktop to Immersive Virtual Environments. Comput. Graph. Forum

Parvinen P, Hart A, Griswold M, et al (2018) Introduction to Minitrack : Mixed , Augmented and Virtual Reality. 1395–1396.

Rau R, Weiskopf D, Ruder H (1998) Special Relativity in Virtual Reality. In: Mathematical Visualization SE - 20. Springer Berlin Heidelberg, Berlin, Heidelberg, pp 269–279

Ritchie SJ, Cox SR, Shen X, et al (2018) Sex Differences in the Adult Human Brain: Evidence from 5216 UK Biobank Participants. Cereb Cortex. doi: 10.1093/cercor/bhy109

Rose F (2011) The Art of Immersion. W. W. Nort. Co.

Sherin Z W, Cheu R, Tan P, Kortemeyer G (2016) Visualizing relativity: The OpenRelativity project. Am J Phys 84:369–374. doi: 10.1119/1.4938057

Shin D (2018) Empathy and embodied experience in virtual environment: To what extent can virtual reality stimulate empathy and embodied experience? Comput Human Behav 78:64–73. doi: 10.1016/j.chb.2017.09.012

Slijkerman, T Corpuscles of energy - https://www.quora.com/What-is-Minkowski-space-time

Smythies J R, John R, French R E (2018) Direct versus indirect realism : a neurophilosophical debate on consciousness.

Smythies J R, Edelstein L, Ramachandran VS (2013) Claustrum Vilayanur S Ramachandran (9780124047228) | Bokus.

Tilot A K, Kucera KS, Vino A, et al (2018) Rare variants in axonogenesis genes connect three families with sound–color synesthesia. Proc Natl Acad Sci 115:201715492. doi: 10.1073/pnas.1715492115

Torgerson CM, Van Horn JD (2014) A case study in connectomics: the history, mapping, and connectivity of the claustrum. Front Neuroinform 8:83. doi: 10.3389/fninf.2014.00083

Zhao Y, Follmer S (2018) A Functional Optimization Based Approach for Continuous 3D Retargeted Touch of Arbitrary, Complex Boundaries in Haptic Virtual Reality. Proc 2018 CHI Conf Hum Factors Comput Syst - CHI '18 1–12. doi: 10.1145/3173574.3174118

WEB REFERENCES

A Space-Time Cocktail: Minkowski Space and Special Relativity. http://www.thephysicsmill.com/2013/02/10/a-space-time-cocktail-minkowski-space-and-special-relativity/. Accessed 9 Jun 2018

A wearable system for VR haptics - Haptic. https://www.haptic.ro/wearable-system-vr-haptics/. Accessed 10 Jun 2018

'Brain Atlas' Charts How We Navigate Language | Nat Geo Education Blog. https://blog.education.nationalgeographic.org/2016/04/28/brain-atlas-charts-how-we-navigate-language/. Accessed 8 Jun 2018

Brain Research through Advancing Innovative Neurotechnologies (BRAIN) - National Institutes of Health (NIH). https://www.braininitiative.nih.gov/?AspxAutoDetectCookieSupport=1. Accessed 29 Aug 2017

"Einstein's Playground" at the Museum of Science. http://gamelab.mit.edu/event/einsteins-playground-at-the-museum-of-science/. Accessed 10 Jun 2018

Geometry Wars (Franchise) - Giant Bomb. https://www.giantbomb.com/geometry-wars/3025-237/. Accessed 10 Jun 2018

Haptic and Immersion - Virtual Reality Pulse. http://www.virtualrealitypulse.com/haptic/immersion/?open-article-id=7691060&article-title=tactical-haptics-is-debuting-its-vr-haptic-controllers-at-ces-2018&blog-domain=catsandvr.com&blog-title=cats-and-vr. Accessed 12 May 2018

HBP (2015) Human Brain Project. https://www.humanbrainproject.eu/en/. Accessed 8 Jun 2018

Hyperbolic Games. http://www.geometrygames.org/HyperbolicGames/. Accessed 10 Jun 2018

Maurice Benayoun | Art After Technology. http://benayoun.com/moben/2008/07/10/art-after-technology/. Accessed 9 Jun 2018

OpenRelativity: free toolkit from MIT Game Lab lets Unity developers play with time and space | PC Gamer. https://www.pcgamer.com/openrelativity-free-toolkit-from-mit-game-lab-lets-unity-developers-play-with-time-and-space/. Accessed 9 Jun 2018

Playing with Einstein | MIT News. http://news.mit.edu/2013/playing-with-einstein. Accessed 10 Jun 2018

String Theory – part 2: Special Relativity – the Picture of Space and Time | SoMA. https://societyofmodernastronomy.wordpress.com/2015/06/18/string-theory-part-2-special-relativity-the-picture-of-space-and-time/. Accessed 9 Jun 2018

The Motor Homunculus - impremedia.net. https://impremedia.net/the-motor-homunculus/. Accessed 13 May 2018

Video gameplay haptics. (2014)

Velocity Raptor | Relativity Game | TestTubeGames. http://www.testtubegames.com/velocityraptor.html. Accessed 10 Jun 2018

BIOGRAPHY

HELENA BARBAS (1951) is Professor-Lecturer of the Departament of Portuguese Studies – Faculdade de Ciências Sociais e Humanas – Universidade Nova de Lisboa. She holds a MA (1990) and a PhD (1998) in Comparative Literature – Literature and the Arts, and gained her "Habilitation" (2008) in Literature and Cyberarts. Research interests: Digital humanities, storytelling, hyper-fiction, avatars, serious games, cognitive-sciences, human computer interaction.

URL: http://www.helenabarbas.net.

MOBILE GAMING

ROLE AND EXPERIENCES OF TUTORIAL IN LOCATION-BASED GAME

Kaisa Könnölä[1], Tuomas Ranti[2], Tapani N. Joelsson[1], Tuomas Mäkilä[1]
[1]Department of Future Technologies, [2]Development Services, Brahea Centre
20014, University of Turku
Finland
E-mail: kaisa.konnola@utu.fi, tuomas.ranti@utu.fi, taneli@utu.fi, tuomas.makila@utu.fi

KEYWORDS
Location-based game, game design, tutorial.

ABSTRACT

Location-based games, such as Geocaching and Pokémon Go, have gained wide popularity during recent years. Since playing a location-based game requires time and willingness to go outside, convincing players to play these games can be more challenging than other types of mobile games. In this article, we research how different tutorials, a location-based (which requires moving around) and a sedentary (which is played without moving physically) could help to overcome this barrier. A location-based game called "Treasure hunt" was developed and used for collecting data from two test periods: first one with a location-based tutorial and second one with a sedentary tutorial. On one hand, the sedentary tutorial increased significantly the proportion of the players who completed the tutorial. On the other hand, it did not increase the proportion of the players who continued playing while moving. Thus, a sedentary tutorial can be considered as a good way to introduce the idea of the game, but does not appear to increase long-term engagement with the game.

INTRODUCTION

Location-based games (LPGs) have gained wide-scale developer interest upon the incorporation of positioning technologies in mobile devices. The success of games like Geocaching and Pokémon Go shows that GPS provides a means of developing fun location-based games that people are eager to play.

The nature of location-based games creates various design challenges. Hardware limitations, such as the inaccuracy of GPS signal (Benford et al 2003), data transfer (Jacob and Coelho 2011) and battery lifetime are things to be taken into consideration. The information availability about places of interest can be limited, creating challenges to the creation of a world-wide playable game (Jacob and Coelho 2011).

The experiencies of the players reveal also non-technical design choices and challenges. The usage of the player's actual physical environment brings the game as a part of the player's everyday life: players experience a greater sense of immersion when the ordinary space and play are not distinct (Saker and Evans 2016; Papangelis et al. 2017). As a downside, this can create safety issues which need to be taken into account (Jacob and Coelho 2011).

Social interactions seem to be a crucial part in many popular location-based games. Even though a game may not have in-game communication, like in the case of Pokémon Go, going outdoors creates opportunities to encounter other players (Paasovaara et al. 2017). The positive effects on advancement in the game that sharing information provides as well as the usage of existing points of interests in real world to enable the encounters encourage social interaction (Paasovaara et al. 2017). According to Kari et al. (2017), playing Pokémon Go made the players more social, gave more meaning to their routines, made them express more positive emotions, and motivated them to explore their surroundings. In addition to the social activity, there is evidence on increase of physical activity with location-based games (Althoff 2016; Fountaine 2018).

Getting people to download the game is not enough, but in order to enjoy the game, they need to understand how to play the game. A tutorial can be used for this matter. In addition to giving understanding about the game mechanics, tutorials are the first tools for engaging users, especially if they create curiosity (Wavro 2015; Järvinen 2010).

In their research, Andersen et al. (2012) suggest that tutorials are to a lesser extent necessary in simpler games, in which players can easily get acquainted with the gaming mechanics through experimentation. However, in most complex games, tutorials would increase playing time as much as 29% and player progress as much as 75%. They also noticed that in the case of a complex game, playtime and progress were increased when instructions were given in close connection to when they were needed, instead of providing them up front at the beginning. However, in the simpler games of their experiment, Andersen et al. did not find context-sensitive instructions beneficial, or even found them resulting in a lower return rate of users.

In a research concerning MMORPGs, usability data did not show significant difference in the overall error rate between such players that had read a manual and such that had not done so (Cornett 2004). It could be thus argued that the benefit of separate manuals in games may be questionable. However, the results suggested that in-game tutorials and context-sensitive help would be the best way to provide players with the information they require to play the game.

When considering the importance of the tutorial, also the type of the player can have an effect. Morin et al. (2016) made a comparison between casual and hardcore players, and their perceptions of a game with or without a tutorial. In their research, they noted that a tutorial can have positive consequences on purchase and continuous use intentions. This is especially valid with casual players and confirms that they need the tutorials.

In Pagulayan et al. (2003), player response to the first mission of a game was measured in a game in which the first missions were intended to act as a tutorial. A third of the players thought that the mission did not give proper challenge, and consequently satisfaction, and caused them to receive a poor initial impression of the game. Many players would complain that they were being taught everything. After changes were incorporated in the game, e.g. increasing the difficulty of the game and creating a separate optional set of tutorial missions, the proportion of players reporting that the game was not exciting reduced from approximately a third of the participants to 3%, without a reduction in the comfort rating associated with the tutorial of game controls. (Pagulayan et al. 2003). This suggests that it is preferable to grant players with a possibility to choose whether they like to carry out the tutorial or not.

According to game developers, the type of the game and its audience affect the creation of the tutorial, and precise advice is thus difficult to give (Wavro 2015). Crumlish and Malone (2009) provide advice on several onboarding techniques for web designers that can be directly used for (social) games (Järvinen 2010): limit the user focus, train the user for the game and take into account the existing user information. Even though tutorial levels can create frustration (Järvinen 2010, Smith 2017), good ones can be designed. A tutorial level should be part of the game, not necessarily easy to win, but easy to learn (Smith 2017). Also attention should be given to ensure that the player is not patronized, forced to complete the whole tutorial if starting the game again, or overwhelmed with too much information e.g. in textual form (Adams 2011; Wavro 2015).

With location-based games, getting people to understand the game mechanics usually requires moving outdoors with the game, which creates an additional barrier to adoption, compared with traditional mobile games. We define *a sedentary tutorial*, which is a tutorial that can be completed without actually going from one place to another physically. It can provide a solution to get past this barrier and get more people to try out a location-based game.

In this article, we will concentrate on experiences of utilizing a sedentary tutorial for a location-based game, and compare it with *a location-based tutorial*, i.e. a tutorial which requires moving physically in the real world from one place to another. We aim to find understanding on whether the sedentary tutorial will attract more players than a location-based tutorial.

RESEARCH METHOD

Our case study consisted of the development of a mobile location-based game and carrying out two test periods after the release of the game. In order to help the players to acquaint themselves with the game, two different tutorial types were experimented during the subsequent test periods. In the first period *a location-based tutorial* was utilized, whereas in the second period it was changed to *a sedentary tutorial*. Our hypothesis was that in location-based games, a sedentary tutorial will engage players and get them to play the actual game more efficiently than a location-based tutorial.

The research questions are:

1. Does a sedentary tutorial get more people to try out the game?
2. Does a sedentary tutorial increase long-term engagement?

Game Description

A location-based game called *The Treasure Hunt* (Figure 1) was created with the Unity game engine to conduct the study. The idea of the game is to walk in the surrounding area and visit a set of virtual fortress islands to find a route to a pirate treasure. After finding enough treasures, the rank of the player increases. The highest rank is "Admiral", which the player is granted after having found the treasures of all nine pirate captains in the game.

Figure 1: a) Basic View of the Game, b) Conquering a Fortress, c) Shop

The basic gameview has a the treasure map (upmost in Figure 1a) and a compass showing the approximate direction (east/west/north/south) of the next fortress to be found and the distance shown above the compass. When the player is close enough to the fortress island (i.e. within 50 meters), the player taps the island to conquering it by tapping the target rapidly (Figure 1b). In the battle, the player's field is damaged. In other islands than fortress islands, the player can find cargo boxes filled with food items to be sold in an in-game store (Figure 1c). A better cannon, a better telescope (increasing find radius to 100m), an accurate compass and fix of player's shield are sold in the store. As a reward from finding a treasure, the player gets money, and either a better telescope, an accurate compass or even more money.

Description of Tutorial Types

In the first test period, *the location-based tutorial* consisted of a treasure map, similar to other maps in the game. The fortress islands were created in the vicinity of the player so that the player had to walk to find the islands. The only difference to the non-tutorial maps was the context-sensitive advice for the player. The tutorial guided the player throughout the game, starting from giving advice of the roles of the treasure map, the compass and the distance. When the player found the first fortress island advice on how to conquer the islands (i.e. tapping) was given and the role of the in-game store was explained. This way, the player played their first map similarly to later maps – the only difference was the context-sensitive advice.

In the second test period, *a sedentary tutorial* was used. New islands were created close to the player so that that the player could reach them without walking anywhere. Then the player was advised to find (and tap) the fortress island, conquer the fortress island (by tapping rapidly) and repeat this with the treasure island as well. This way, the player could get a grasp on the idea of the game without moving physically. Only after playing the sedentary tutorial, the player started to play the game by walking in the surrounding area.

Research Settings

The aim of the game development was to create a game, which would interest the players for a short period of about two weeks. The core of the game development was iterative and feedback was received from researchers, marketing professionals and from a consumer web forum.

There were two test periods, each of which lasted for about two weeks. The first one was at the end of June 2017 and the second one was at the beginning of September 2017. Potential players were contacted in several ways: 1) e-mail lists of both personnel and students of the university were utilized, 2) posters of the game were displayed in the vicinity of the university, 3) social media accounts of the university were utilized and 4) in total about 200 flyers were distributed in the streets nearby. In all contact channels, the possibility to win a gift certificate (50 euros) to Steam, Google Play or Spotify was advertised. At the end of each test period, one gift certificate was given to the best player of the test period and another gift certificate was given through a lottery among the top 60% of the players.

In the first test period, the game was installed 131 times, out of which 105 users also registered to the game. In the second test period, there were 65 installations and 65 registrations. When registering to the game, e-mail address was given and according to this information, only one player continued playing the game in the second round. In-game data, such as player names, and points for the high score list was collected to the server, and this information is used as the first data source in this research.

As another data source, a questionnaire about the game and the way it was experienced was sent to the players after each test period. In order to be able to win a gift certificate, an answer to the questionnaire was required. In the first round, 26 answers were received to the questionnaire and in the second round 29 answers.

RESULTS

The high score list of the game was maintained in the server. After each test period, the situation of the high score list was recorded. According to the game logic, the player would get about 20 points from playing each of the first two maps.

Based on this information, we were able to divide the players into four groups:

- players, who did not even complete the first map, i.e. the tutorial (<20 points),
- players, who completed about one map, i.e. the tutorial (20– 40 points),
- players, who completed about one map after the tutorial (40– 60 points), and
- players, who completed more than one map after the tutorial (over 60 points) and can be therefore considered as active players.

Figure 2 presents how many treasure maps people completed, as a percentage of the total amount of players who signed into the game in each round. As can be seen, the proportional amount of players who completed the tutorial, i.e. the first map, increased significantly between the two test periods: from one third, to two thirds. According to Chi-squared test ($p=0.000013<0.05$), the difference of the proportion of people completing the tutorial, is significant.

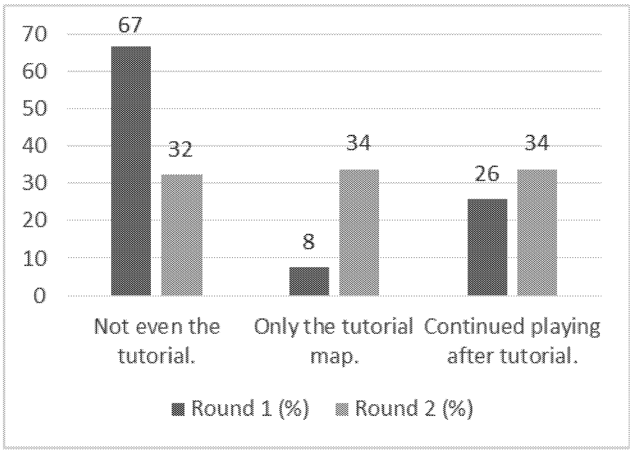


Figure 2: The Proportional Distribution of Number of Treasure Maps Played

To gain insight for our second research question on whether a sedentary tutorial encouraged the players to continue the gameplay also after the tutorial, we compare the players who completed location-based treasure maps. In Figure 3, the first two bars of each round represent the players, who did not complete even a single location-based treasure map. Two latter bars represent the players who completed one or more location-based treasure maps. The percentages of the players who did not complete even one location-based treasure map remained about the same in each round. When we compare the amount of people, who continued to play after the first location-based treasure map, i.e. the long term players, there is a slight, but not statistically significant (Chi-squared test $p=0.47>0.05$), increase from 26% to 31%.

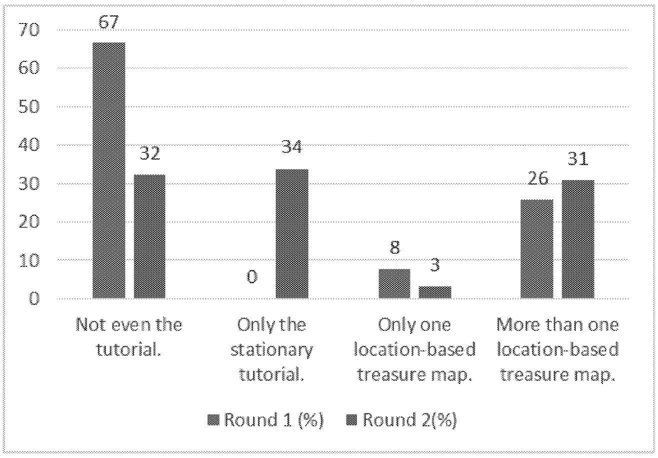


Figure 3: Proportion of Location-Based and Stationary Treasure Maps Played

In the questionnaire to the players, possible reasons for not playing at all or ending the gameplay were asked, as can be seen from Figure 4. The major reason was the time the game requires. Also some people found that the game was not versatile enough. Several technical issues were reported as well, and even though a tutorial was present, it did not give enough guidance to all the players.

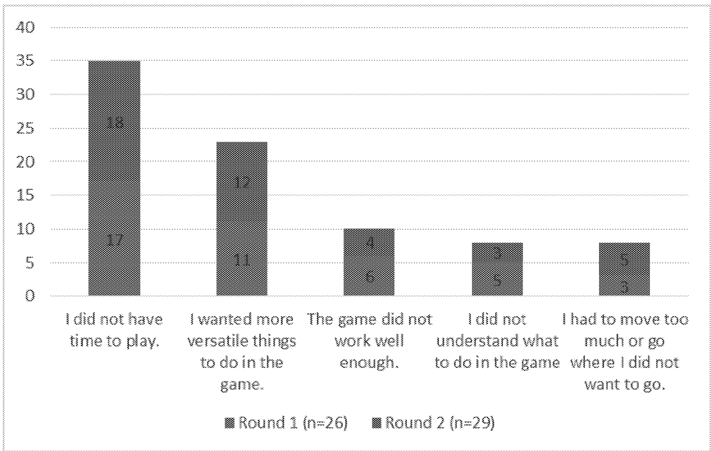


Figure 4. Reasons for not Playing at All or Ending Gameplay

DISCUSSION

In location-based games, there can be a) context-sensitive instructions that are both related to the physical surroundings and the in-app game environment, and b) context-sensitive help related merely to the in-app game environment. The sedentary tutorial relates to b), meaning that tutorial covers issues present in the game application, but cannot provide context-sensitivity related to the real world. In the location-based tutorial also context-sensitivity related to the real world was present in form of presenting the meanings of the compass and the distance.

The game logic consists of quests, i.e. treasure maps, to be completed by winning battles. In the battle, the only thing the player has to do to conquer a fortress is to tap the target rapidly. Thus, the game logic can be considered a quite simple one. If the shield is damaged, the player has to collect sellable items from islands by walking in the surroundings. Even though logically this is simple, it is time consuming to collect enough sellable items and can lead to frustration. According to Andersen et al. (2012), tutorials for simpler games are not beneficial. We consider that the perseverance required, and especially the penalty if not understanding what to do in the game, makes location-based games more complicated, regardless of the game logic. Thus we thought that a context-sensitive tutorial was required for the game.

In the tutorial, the complexity required both by scientific research (Andersen et al 2012) and the game designers (Smith 2017) was taken into account: the location-based tutorial was designed to be a part of the game and the difficulty was close to the difficulty of the first real treasure map. The user got context-sensitive information about the essential functions little by little, as they were guided through the gameplay of the first map. The number of textual advice was kept as little as possible (Adams 2011), so that the player would feel like playing instead of going through a tutorial. In the sedentary tutorial, the tutorial was kept simple and small, in order to enable a quick playthrough. This was seen as a way to avoid frustration associated with obligatory tutorial levels (Järvinen 2010, Smith 2017).

Engaging people to play location-based games can be due to the perseverance required. Many people who download the game quit the game before testing its location-based features. This was proven in our research as well, where time was the major reason for not playing or quitting to play. According to our results, the difference of people who completed the tutorial rose from one third to two thirds with a sedentary tutorial compared with a location-based tutorial. Thus, we consider that a sedentary tutorial will help more people to gain an understanding about the game idea than in the case of a location-based tutorial.

Even though a larger proportion of players completed the sedentary tutorial than the location based tutorial, it did not significantly influence the proportion of those players who tested the game by walking in their surroundings. According to our questionnaire, a third thought that the game was not versatile enough. Reflecting on our sedentary tutorial, it might not have been challenging and satisfactory enough to encourage the players to continue the game. Thus our results are in-line with the research by Pagulayan et al. (2003).

As time was one of the major reasons for not playing, we can also consider time as one of the major reasons for a larger proportion of players completing the sedentary tutorial, compared with the location-based tutorial. This will also highlight our conclusion of sedentary tutorial being a good way to introduce the game logic.

CONCLUSIONS AND FUTURE WORK

The aim of this paper was to gain more insight on how different tutorials affect the amount of players playing a mobile location-based game. Our case study included a mobile game which was tested on real players in two distinct test periods: first test period with a location based tutorial and second test period with a sedentary tutorial.

According to our study, the proportion of players who completed the tutorial rose from 33% to 68% with a sedentary tutorial. A sedentary tutorial will significantly increase the amount of players who will complete the tutorial. This will get more players to get a grasp on the idea of the game, before actually diving into the real world to walk and play.

Another even more interesting question is whether a sedentary tutorial will engage the players in the longer term. According to our study, the proportion of people who actually played the game in the real world was around one-third in both test periods, even though a slight increase from 26% to 31% was noted.

Our results show that a sedentary tutorial will work best for presenting the idea of a location-based game to the players. However, using a sedentary tutorial includes a risk of making a too simple tutorial, which can drive the players away.

In the research setting, a possibility was given to win gift certificates in order to attract people to try out the game. These prizes may have had an effect on the way people played – perhaps they were not interested in the game itself, but played only to get the prize.

Also some popular location-based games, for example Pokémon GO, include a sedentary tutorial. It would be interesting to know whether the proportion of people who play only the sedentary tutorial is similar in them.

ACKNOWLEDGEMENTS

The research reported in this article has been conducted as a part of the Co-Creative Snacks project, which is mainly funded by the European Regional Development Fund via Tekes – the Finnish Funding Agency for Technology and Innovation.

REFERENCES

Althoff T.; R.W. White; and E. Horvitz. 2016. “Influence of Pokémon Go on Physical Activity: Study and Implications”. *Journal of Medical Internet Research*. (Dec 6), Vol.18, No.12, e315.

Andersen, E.; E. O’Rourke; Y-E. Liu; R. Snider; J. Lowdermilk; D. Truong; S. Cooper; and Z. Popovic. 2012. “The Impact of Tutorials on Games of Varying Complexity”. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (Austin, Texas, USA, May 5 - 10). ACM, New York, NY, USA, 59-68.

Benford S.; R. Anastasi; M. Flintham; C. Greenhalgh; N. Tandavanitj; M. Adams; and J. Row-Farr. 2003. "Coping with uncertainty in a location-based game". *IEEE Pervasive Computing*. (July-Sept), Vol. 2, No. 3, 34-41. doi: 10.1109/MPRV.2003.1228525

Cornett, S. 2004. “The Usability of Massively Multiplayer Online Roleplaying Games: Designing for New Users”. *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (Vienna, Austria, Apr. 24 - 29). ACM, New York, NY, USA, 703-710.

Crumlish C. and E. Malone. 2009. *Designing Social Interfaces.*O’Reilly Media, Sebastopol, VA (Sep).

Fountaine C.J.; E.J. Springer; and J.R. Sward. 2018. “A Descriptive Study of Objectively Measured Pokémon GO Playtime in College Students”. *International Journal of Exercise Science.* (Jan 2) Vol. 11, No. 7, 526-532.

Jacob J. and A. Coelho. 2011.”Issues in the development of location-based games”. *International Journal of Computer Games Technology*. Vol. 2011, Article ID 495437, 7 pages. https://doi.org/10.1155/2011/495437.

Kari T., Arjoranta, J. and Salo, M. 2017. “Behavior change types with Pokémon GO”. In *Proceedings of the 12th International Conference on the Foundations of Digital Games (FDG '17)* (Hyannis, Massachusetts, Aug. 14-17). ACM, New York, NY, USA, Article 33, 10 pages. DOI: https://doi.org/10.1145/3102071.3102074

Morin R.; P.-M. Léger; S. Senecal; M.-C. Bastarache-Roberge; M. Lefèbrve; and M. Fredette. 2016. “The Effect of Game Tutorial: A Comparison Between Casual and Hardcore Gamers”. *In Proceedings of the 2016 Annual Symposium on Computer-Human Interaction in Play Companion Extended Abstracts (CHI PLAY Companion '16)* (Austin, Texas, USA, Oct. 16-19). ACM, New York, NY, USA, 229-237. DOI: https://doi.org/10.1145/2968120.2987730

Paasovaara S.; P. Jarusriboonchai; and T. Olsson. 2017. “Understanding collocated social interaction between Pokémon GO players”. *In Proceedings of the 16th International Conference on Mobile and Ubiquitous Multimedia (MUM '17)* (Stuttgart, Germany, Nov. 26-29). ACM, New York, NY, USA, 151-163. DOI: https://doi.org/10.1145/3152832.3152854

Pagulayan, R.; K. Keeker; D. Wixon; R. Romero; and T. Fuller. 2003. “User-centered design in games”. In *The Human-Computer Interaction Handbook: Fundamentals, Evolving Techniques and Emerging Applications*, J.A. Jacko and A. Sears (Eds.). Lawrence Erlbaum Associates, Mahwah, New Jersey, 883 - 905.

Papangelis K.; M. Metzger; Y. Sheng; H.-N. Liang; A. Chamberlain; and V.-J. Khan. 2017. “"Get Off My Lawn!": Starting to Understand Territoriality in Location Based Mobile Games”. In *Proceedings of the 2017 CHI Conference Extended Abstracts on Human Factors in Computing Systems (CHI EA '17)* (Denver, Colorado, USA, May 06-11). ACM, New York, NY, USA, 1955-1961. DOI: https://doi.org/10.1145/3027063.3053154

Saker M. and L. Evans. 2016. “Everyday life and locative play: an exploration of Foursquare and playful engagements with space and place”. *Media, Culture & Society 2016.* Vol. 38, No.8, 1169 –1183.

WEB REFERENCES

Adams E. 2011. “The Designer’s Notebook: Eight Ways To Make a Bad Tutorial”. Gamasutra. https://www.gamasutra.com/view/feature/134774/the_designers_notebook_eight_.php (Accessed 24.5.2018).

Järvinen A. 2010. “First Five Minutes: How Tutorials Make or Break Your Social Game”. Gamasutra. https://www.gamasutra.com/view/feature/132715/first_five_minutes_how_tutorials_.php (Accessed 24.5.2018).

Wawro A. 2015. “The art of the tutorial: When to hold a player’s hand, when to let it go”. Gamasutra. https://www.gamasutra.com/view/news/235148/The_art_of_the_tutorial_When_to_hold_a_players_hand_when_to_let_it_go.php (Accessed 24.5.2018).

Smith G. 2017. “We need a revolution in tutorials”. Rock, Paper, Shotgun 2017-1-9. https://www.rockpapershotgun.com/2017/01/06/we-need-a-revolution-in-tutorials/ (Accessed 24.5.2018)

QUALITY ASSURANCE IN A MOBILE GAME PROJECT: A CASE STUDY

Maxim Mozgovoy
The University of Aizu
Tsuruga, Ikki-machi, Aizu-Wakamatsu,
Fukushima, 965-8580 Japan
mozgovoy@u-aizu.ac.jp

KEYWORDS

Quality assurance, mobile games, automated testing.

ABSTRACT

Quality assurance is an integral part of the software development process. Game projects possess their own distinctive traits that affect all stages of work, including quality assurance. The goal of this paper is to share the lessons learned during a three year-long mobile game development project. I will discuss the techniques that turned out to be most efficient for us: manual testing, automated and manual runtime bug reporting, Crashlytics crash analysis, and automated smoke testing. I will outline how these types of testing address typical game-specific issues, and why they can be recommended for a wide range of game projects.

INTRODUCTION

Quality assurance is a complex set of methods, used in all stages of software development, ranging from requirements engineering and software design to coding and testing. Explicit quality assurance measures are found in all widely used software development processes, from traditional waterfall model to modern agile approaches [1].

Still, quality issues are common in resulting software products. Khalid et al. [2] analyzed user reviews of 20 most popular iOS apps of June 2012. They found that 26.68% or user complaints are related to functional errors, and other 10.51% of complaints mention app crashing. Together with “feature request”, they constitute top 3 complaint types.

One may argue that the best way to ensure software quality is to maintain high standards of software development culture. Indeed, poor design and planning, and somewhat relaxed attitude to writing code is often mentioned as the primary reasons for buggy software [3]. Thus, gradual improvement of software development processes is a necessary, but difficult and time-consuming measure.

I will concentrate on relatively simple, but cost-efficient “last resort” measures, aimed to reveal bugs before they creep into the release version, and to facilitate quick fixes of bugs not identified during testing. While all these methods are well-known, they deserve additional discussion within the process of game development, since it has certain distinctive traits that affected our views on quality assurance.

WORLD OF TENNIS: ROARING '20S GAME

The observations discussed in this paper were made during the development of a mobile tennis game *World of Tennis: Roaring '20s*. The most interesting aspect of the game is the presence of machine learning-based AI system that observes players’ behavior to substitute them in player-vs-player matches [4]. This capability allows the players to compete against each other at any time, and mitigates all negative effects of poor internet connection.

From organizational point of view, *World of Tennis* a typical mobile game project, developed by a small team during a time span of three years. The game is written in Unity game engine, and is currently available for iOS, Android, and Universal Windows platforms. The game is free to play (i.e., supported by additional in-app purchases), and requires internet connection for most actions.

GAME DEVELOPMENT-SPECIFIC FACTORS

The nature of a software product we create affects the whole development process, including quality assurance. Game development has its own peculiarities, discussed in literature [5, 6]. The most significant factors that affected our approach to quality assurance were the following.

1. Heavy reliance on unstable 3rd-party libraries and tools.

We have to use specific libraries to integrate with external services (such as ad providers), and to rely on Unity for internal game engine functionality. Some of 3rd-party modules are quite complex, unstable, and may cause app crashes. Often we have to decide whether to use a library that provides a functionality needed for a certain feature, or to cut this feature at all.

In practice, it means that our approach to functional errors and crashes has to be nuanced. For example, we might decide to tolerate a certain level of crashes if it lets us to integrate with an ad provider or enable great-looking cloth simulation.

2. Diversity of hardware and software platforms.

Unity greatly simplifies the process of cross-platform development, encouraging the developers to take advantage of this capability, and to release the game on a wide range of platforms. In turn, it means that the game has to be tested on each platform separately.

Platform-specific errors typically occur in fragments of code appearing in native binary libraries and in procedures calling platform-specific SDKs (e.g., for in-app purchases).

Diversity of hardware and operating systems also imposes challenges. Some distribution channels such as Apple and Google stores allows the developers to specify the types of compatible devices by providing the required OS version and hardware configuration. It leads us again to treat known flaws pragmatically: if the game does not work properly on certain devices, it might be reasonable to consider them incompatible rather than invest efforts into patches.

3. Abundance of visual and sound issues.

A great number of bugs in games can only be revealed with manual testing. For instance, we had situations when shadows were not visible, the colors of clothes were wrong, the characters had their feet below the ground level, some text boxes overlapped with other GUI elements or were too small to contain the corresponding text lines. Similar observations can be made about animation and sound effects.

Therefore, automated testing in game projects is applicable to a relatively narrow set of cases. Ironically, this factor motivated us to automate as many scenarios as we could to give our testing team more time to find nontrivial bugs.

4. Large proportion of high-cost unit testing code.

Literature on agile development speaks in favor of unit testing, but one should note that the associated costs are distributed unevenly. Sanderson [7] identifies two types of code with high cost of unit testing: complex code with many dependencies, and trivial code with many dependences ("coordinators" between other code units). According to Sanderson, complex code with many dependencies should be refactored to separate algorithms from coordination.

Our experience shows that a game project has a large proportion of both types of high-cost unit testing code. I believe the primary reason for it is that the most cost-efficient type of code ("complex code with few dependencies" in Sanderson's scheme) belongs to the game engine such as Unity and 3rd-party libraries. The problem is further aggravated with the fact that "complex code with many dependences" is rarely refactored in practice and thus also cannot be unit-tested efficiently.

It might be tempting to attribute the lack of refactoring and frequently noted substandard design of system architecture in game projects to low culture of development. However, there are objective factors contributing to this situation. In particular, games have to be *entertaining* and provide *excitement* — requirements that can hardly be satisfied with traditional planning methods. Therefore, game programming requires much experimenting, and it is not surprising that the developers tend to view much of their work as "throwaway code", poorly engineered and rarely refactored [5].

5. Deep integration of GUI and animation

Automated tests (especially unit tests) often rely on the possibility to separate entities. One might want to test game physics separately from animation or GUI independently from underlying logic. However, it might be virtually impossible to do in a game. For instance, in Unity animation is an integral part of character motion model. To check the changes in character's coordinates during movement, one has to play the related animation sequence. The notion of "user interface" is also vague in games, as any clickable onscreen object can be considered a part of interface. Furthermore, typical user controls like buttons or edit boxes are often hand-drawn in games and thus inaccessible through standard automation interfaces (such as UI Automator in Android or XCTest in iOS).

QUALITY ASSURANCE IN WORLD OF TENNIS

This section is dedicated to a more detailed discussion of some specific measures we implemented in the project. We consider them useful and cost-efficient, and are willing to adhere to the same practices in the future.

Crashlytics Crash Reporting. As mentioned in the previous section, we take a pragmatic approach to errors. With numerous 3rd-party modules we use, Unity as a game engine, and a variety of supported platforms and devices, malfunctions are inevitable. Our task from the early stages of development was not only to identify faults, but also to assess their severity for the product.

One of our first decisions was to integrate Crashlytics crash reporting service[1]. It embeds special crash reporting code into the application, which sends crash details into a central server. As developers, we can analyze the reasons of crashes and the list of devices where crashes occur.

In particular, Crashlytics helped us to identify devices with inadequate amount of RAM. On mobile platforms, a task scheduler can kill a foreground application if it consumes too much memory, which is practically equivalent to a crash. However, it is hard to decide where exactly one has to draw a line, since numerous devices belong to a "gray area" where crashes are possible, but not certain. Actual statistics from Crashlytics helped us to make a well-grounded decision.

Autobugs and Manual Bugs. Developers widely use *assertions* to check assumptions about certain points in code. Assertions can be seen as a part of "design by contract" approach [8]. There is a general agreement that assertions should be used during development as a method for both in-code documentation and quality assurance, but the practice of keeping assertions in production code is debatable [9]. The arguments often depend on what assertions *actually do*, and the typical presumption is that a failed assertion shows an error message and terminates the application.

In our game, each failed assertion and each raised exception is reported to us. We presume the presence of internet connection on user devices, thus error reporting is easy to automate. Our task and bug tracker Teamwork[2] has a capability to create tasks via email messages, which we use to gather information about failed assertions and raised exceptions. Each report contains basic information about the build, user device, and current user account. It also contains a link to the detailed session report stored on our server.

The same technology is used for reporting "manual bugs". The users marked as beta-testers in the system have an option to pause the game at any moment and send a bug report. It will be posted to Teamwork in the same manner along with the session report and with a user-supplied description. As noted above, massive manual testing in games is inevitable, so we started recruiting beta-testers one year before release.

Manual Testing. Our approach to manual testing is straightforward. As soon as we get a new build that is considered "stable", we ask our testers to play several game sessions, noting any problems they encounter. All game

---

[1] https://crashlytics.com

[2] https://www.teamwork.com

sessions are recorded as video clips, and the testers illustrate their findings with links to particular video fragments. Since our QA team is small (only two people test regularly), we also rely on a professional QA company to check our major release builds on a variety of devices and platforms.

Automated Smoke Testing. Smoke testing is a type of functional testing aimed to reveal failures in a complete system by covering a broad product features with simple automated test scenarios [10]. We automate testing of simple routine actions, such as: 1) create a new user and pass the tutorial; 2) play a league match against the next opponent; 3) upgrade your character's skills using available experience points; 4) link your Facebook account to the game; 5) change current club / character / clothes / equipment. These actions require most subsystems of the game to operate correctly, so it can be expected that such automated testing would identify many critical bugs.

Technically, mobile smoke tests can be set up using an external service, such as Bitbar Testing[3] or AWS Device Farm[4]. However, we found them too expensive for daily use, and set up our own mobile farm of one Windows, three iOS, and four Android devices [11]. The testing farm is fully integrated into our pipeline. When a new build is available on the build machine, the system runs predefined test scripts on all devices in the farm.

The scripts interact with our mobile devices via Appium framework[5] and use image recognition to identify clickable GUI elements. Test logs are available as HTML reports with screenshots, illustrating ongoing actions. If a certain test fails, it is easy to identify the cause in most cases.

These automated tests can also generate autobugs, so even if there are no obvious faults reported by the test, it still might detect errors via the mechanism of assertions and exceptions.

DISCUSSION

Mobile free-to-play games is a special kind of product. They require long-term experiments with game mechanics, monetization techniques and new features, thus exhibiting the traits of both games and non-game applications.

Research shows that game programmers believe there are substantial differences in their work practice comparing to the work practice of non-game developers [5]. In particular, game projects suffer from loosely formulated requirements, frequent changes of core system elements, heavy reliance on manual testing, and little incentive to improve architecture, since much of the work is seen as disposable code. In a sense, a game is like a movie: once it is ready, nobody needs props anymore.

Mobile free-to-play games is not an exception in regards to coding practice, but they require strict and reliable quality assurance process to make sure that regular updates do not break the game. It is incredibly difficult to establish a place in a hyper-competitive environment of modern mobile app stores, and bugs may cause a quick descent.

Therefore, I believe that games would benefit from a more comprehensive approach to testing that takes into account specific issues related to game development. Not all commonly recommended practices are well suitable for game developers, and the right answer to this challenge would be to identify the practices that work best.

CONCLUSION

Numerous objective factors have a negative effect on mobile game projects. However, they cannot serve as an excuse for functional errors and crashes, haunting many games. Instead, they should be seen as challenges for more comprehensive and streamlined quality assurance procedures, based on cost-efficient measures that take into account the distinctive nature of game projects. In our mobile game *World of Tennis: Roaring '20s*, a combination of crash reporting, autobugs and manual bugs, manual testing, and automated smoke testing is used. All these elements work together, providing a clear cumulative effect. Most of these subsystems are easy to setup, and can be implemented in a small team on lean budget.

REFERENCES

[1] M. Huo, J. Verner, L. Zhu, and M. A. Babar, "Software quality and agile methods," in *The 28th IEEE International Conference on Computers, Software & Applications*, 2004, pp. 520–525.

[2] H. Khalid, E. Shihab, M. Nagappan, and A. E. Hassan, "What do mobile app users complain about?," *IEEE Software*, vol. 32, no. 3, pp. 70–77, 2015.

[3] C. C. Mann, "Why software is so bad," *Technology Review*, vol. 105, no. 6, pp. 33–38, 2002.

[4] M. Mozgovoy, M. Purgina, and I. Umarov, "Believable Self-Learning AI for World of Tennis," in *2016 IEEE Conference on Computational Intelligence and Games*, 2016, pp. 1–7.

[5] E. Murphy-Hill, T. Zimmermann, and N. Nagappan, "Cowboys, ankle sprains, and keepers of quality: How is video game development different from software development?," in *Proceedings of the 36th International Conference on Software Engineering*, 2014, pp. 1–11.

[6] L. Pascarella, F. Palomba, M. Di Penta, and A. Bacchelli, "How Is Video Game Development Different from Software Development in Open Source?," in *MSR Conference*, 2018.

[7] S. Sanderson, *Selective Unit Testing – Costs and Benefits.* Available: https://bit.ly/2tVIUcx.

[8] B. Meyer, "Applying 'design by contract'," *Computer*, vol. 25, no. 10, pp. 40–51, 1992.

[9] W. Bright, *Assertions in Production Code.* Available: https://digitalmars.com/articles/b14.html.

[10] C. Kaner, J. Bach, and B. Pettichord, *Lessons learned in software testing: A context-driven approach*. New York: Wiley, 2002.

[11] M. Mozgovoy and E. Pyshkin, "Unity application testing automation with Appium and image recognition," in *International Conference on Tools and Methods for Program Analysis*, 2017, pp. 139–150.

[3] https://bitbar.com/testing

[4] https://aws.amazon.com/device-farm

[5] http://appium.io

ONLINE GAMING

TRADE-OFFS AND CHALLENGES OF CLOUD GAMING IN PRACTISE

Robert J. Grigg
International Game Architecture & Design (IGAD)
Breda University of Applied Sciences
email: grigg.r@buas.nl

René Hexel
Institute for Integrated & Intelligent Systems
Griffith University
email: r.hexel@griffith.edu.au

KEYWORDS

Cloud Gaming, File Streaming, Cloud Game Engine Architecture

ABSTRACT

Offering games in the cloud is something being realised by many operations, but still there are diverse challenges in conducting this successfully. Currently it is expected that graduating game development students will have at least one published game and a chance to experience the release phase and post-production aspects of game development. This led to the inspiration for a project to create the first University cloud games channel in an effort to stop our game development history being lost due to reasons such as storage media failure, along with providing an avenue for students to experience game release, operation and community management.

A study evaluated various technologies and requirements from stake-holders. The criteria created was then used to identify the best approach to undertake and implement. The results includes the selection and development of the University games channel using an asset streaming solution in partnership with *Utomik*, along with the supporting processes for games to be delivered to the platform. This solution makes use of specialised offline asset compression combined with asset streaming prediction. The game requires no installation and the performance of the caching of the game required to start playing was profiled and found to perform at an average of 29 percent of the total game size for our student games, that was then compared to the results of industry games that performed on average at 4 percent.

INTRODUCTION

In the Netherlands there is a game development programme at the University of Breda called International Game Architecture and Design (IGAD). During the four year study, students spend a large percentage of time working in game development teams of artists, programmers, designers, producers, and audio engineers to create at least four or more games before seeking an internship. The results are a variety of interesting games where the end-product is often on a DVD that finds its way to an archive or lecturer's desk draw.

Many games created at our institution were disappearing, in most cases, forever. The later retrieval, review, or demonstration of these games proved to be difficult to impossible to arrange. The challenges include the media they were submitted on perishing, the operating system version not being readily available, or there being a missing 3rd party software component. All these factors prevented the game from being played and brought about the loss of the creative history of students and alumni, as well as meaning an absence of playable game artefacts from students' online portfolios and the University archive.

This led to the question of whether it is feasible for an educational institution to leverage a cloud platform to solve these issues faced. Creating a cloud based games channel is something that is achievable today but not without challenges. Most important are the different approaches for cloud gaming, where the methods and underlying infrastructure used greatly impact the operational costs, benefits and feasibility for the operator, developer and end-users. This paper presents the motivating background, the challenges addressed in the search for a possible solution and the resulting selection criteria for the decisions made. Finally the performance results and processes formed to help make such an endeavour operational are discussed.

BACKGROUND

One challenge of a game development education is giving students the experience, skills and knowledge of each stage of the game development life-cycle. Most students will experience the core pre-production and production aspects of game development up to the creation of a game prototype or minimum viable product. In most cases students will never release a product to any community and, therefore, miss out on all the learning opportunities the later phases of game development present (see Figure 1).

Aiming for release or publishing on existing commercial platforms meant that games needed to be of a scale and quality that involved years of development and commitment from student teams. Examples of games from our students include FRU, a Xbox One Kinect game from Through Games (Through Games 2016), which went through the publishing process with Microsoft's *Xbox*

Figure 1: Game Development Life-cycle and Areas of Student Experience

Live (2016) where the game took two and a half years of development before final release. Feedback from students was that the experience of completing a game with this level of polish, meeting stringent technical requirements and later managing the game issues and player community post release was something valuable to the game development learning experience.

In later years of the programme some student teams would work hard enough to submit and go through Valve's *Steam* (Newell 2003) greenlight process. Kabounce is an example of an arcade game for the PC from Stitch Heads (Stitch Heads 2016) that is progressing through the *Steam* process and now is eighteen months into development and planning to release within the two year mark. Again the scale and quality of the product is a major hurdle in meeting the requirements for release on this platform.

Although both examples of FRU and Kabounce are beneficial learning experiences, sometimes the commitment required by an assembled group of students can prove to be too much when the timeline is two or more years. Implementing a public relations campaign, executing marketing plans and community management are aspects that go even further and require even more time and effort.

Annually our education's industry board meets with large developers, such as *Sony* (2016) and *Ubisoft* (2016), through to smaller local developers like *CodeGlue* (2016). Members of the industry board agreed that today a student approaching them for an internship should have at least one published game to their name. Being able to present a game that has been released is a differentiator for a student in comparison to other graduates, personally demonstrating the ability to finish a game and understand the bigger picture of the game development life-cycle.

In a larger set of circumstances student teams want to develop a game and release it in a form similar to that of pre-game release demos, this being where only one or two levels are complete, but done to a professional level representative of the final game quality aimed for. The release of this would no longer be applicable on services such as *Xbox Live* or *Steam* as they would expect the full product to follow at a later stage. The result is a need for a platform where games can be released to communities for students to experience the complete development cycle including post release activities.

Another concern was the growing scale of even small student games. The main contributing factor was the growing size and abilities of the selected game engines used. A game called *Blank Slate* was developed last year using the *Unreal Engine 4.11* (2016) ended up being well over 3 GB in size. Most games aiming for larger game worlds require well above 1 GB in storage space.

The impact of the size of commercial games was illustrated in a series of workshops with the *BigWorld* (2015) engine where students would develop a vertical slice of a game concept for the Massive Multiplayer Online (MMO) genre. Part of this series included using an existing MMO title Star Trek Online (STO), developed by *Cryptic Studios* (2015), to download, install and play. The game required an initial download size of 4.7 GB for the whole game. Before being able to play a patch approaching another 1.5 GB needed to be downloaded, with the patch approaching 5 GB on the *Steam* service. The download required a day, and once up and running, it was delayed another two hours due to server sign-in problems. Once operational, one of the workshops used the *Kalydo* toolset to package the game STO for local servers to simulate the distribution of the game. The result was that the game, via a single in web-browser click, would stream the assets to the point of where 250 MB was downloaded and then the game would be playable, this taking only minutes. Students then were able to apply this to their own games. From these experiences the need to employ smart distribution technologies and leverage cloud services was seen as the way forward in helping effectively deliver these ever-growing games of the future.

CLOUD GAMING APPROACHES

The core underlying architecture for a cloud gaming platform has a number of different approaches in delivering the game from the cloud cluster to the gaming client.

A lot of current work has leveraged interactive remote rendering systems of the past and is covered in a survey by Shi *et al.* (Shi and Hsu 2015). Cai *et al.* (Cai et al. 2016) has completed a broad study defining cloud gaming and the ongoing research and challenges. Prior work surveyed various cloud gaming architectures in both research and industry contexts to further define opportunities and challenges in cloud gaming (Grigg and Hexel 2017). The main approaches identified include:

1. *Streaming of Graphics Commands:* The streaming of graphics commands, that is also referred to as *Graphics Streaming*, involves the Application Programming Interface (API) level intercept of graphics commands such as OpenGL or DirectX, that are then compressed and streamed to the client to be executed. This is an advantage for a *cloud cluster* because it reduces the amount of server-side processing and thus allows each server to support more users (Jurgelionis et al. 2009). The disadvantage is that bandwidth is subject to frame-rate (Stegmaier et al. 2002). High quality games present more of a challenge with large numbers of draw commands per frame requiring methods of compressing and caching to reduce network throughput (Eisert and Fechteler 2008; Liao et al. 2016).

2. *Streaming of the Video Image:* This approach is about the streaming of a video game, or Video-Streamed Games On Demand (VSGOD), where the cloud cluster renders the 3D world to a 2D image that is then captured, encoded, and compressed into a video format before transmission. The client receives the stream and decompresses it using a decoder to replay the video. The client captures input from the user to return to the server. The advantage of this is that *thin clients* do not require special hardware, such as GPUs, although they may have cheap decoder chips that support quick video decoding (Cheng et al. 2004; Holthe et al. 2009; Huang et al. 2013a,b).

3. *Streaming of Files:* The streaming of files utilises smart distributed file systems that intelligently transmit files in a timely manner to be available for the game client-side. Files may have individual compression approaches and the transmission of file chunks may help when delivering game patches where only small parts of large files change. This solution avoids problems with in game latency, has no network limitations where most modern day connections are all that is required and playing of games while offline is possible.

4. *Streaming of Memory Pages:* The streaming at the memory page level of an application is done by sending memory pages as required by the user application. The approach uses a statistical method to help predict what pages are required next by the client application. The handling of the memory pages is done by a virtual machine layer on the client-side which communicates with the server to retrieve the previously virtualized memory page making it possible to play after under 10 percent of the game has been downloaded. Users can play offline, based on what is cached already, and peers within local area networks can form proxies for faster downloads. Given that a platform has a cloud paging virtual machine available, then the same games can be played across different hardware devices. This process was called *Cloudpaging* by Numecent (Ahiska 2015) and brought to gaming by a spin-off company Approxy (Ahiska 2012) that was later reacquired by Numecent.

5. *Hybrid Approaches:* These include combinations of the above approaches, sometimes adding client-side post-processing techniques. As an example addressing frame-rate in a video streaming solution, the use of image warping within the framework of the H.264 encoder delivers increased performance of up to twice the frame-rate upon the client and is a suggested improvement through use of a custom *codec* (Giesen et al. 2008).

A Typical Cloud Gaming Platform View

A typical view of a *cloud gaming system* is one that renders the game scene on the cloud cluster and then transmits the encoded scenes to an end-user device via the internet or similar broadband connection. User input is taken from the end-user device and transmitted to the *cloud gaming cluster* so it can update the game state and render the next frame as illustrated in Figure 2. This architecture means that game-code is stored and executed on the server with the benefit of being able to deliver the game in a generic fashion suitable for a wider selection of end-user devices while minimising the chance of piracy.

In the *cloud cluster*, an important differentiating factor is whether the system is created generically, so it can plug in as many game products as possible. Alternatively the cloud gaming platform can be built to leverage a selected game engine internally where the engine information and functionality may enable better performance but tie the solution to a particular game engine and, therefore, introduce the need for porting of a game product to the cloud gaming platform (Marpe et al. 2003). The creation of a generic plug-in system that allows for a wide range of games to be *plugged-in* negatively impacts performance due to the additional steps required to capture game output as a *black box* process.

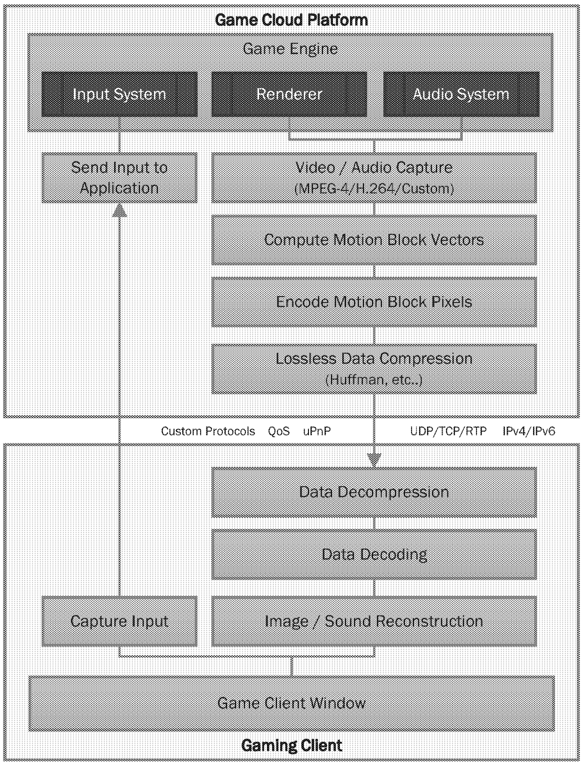


Figure 2: An Architectural View of Cloud Gaming and Video Streaming

CLOUD REQUIREMENTS

The first step in setting up a cloud gaming channel is the selection of the type of cloud platform. This presented a large number of challenges for a small non-profit, educational institution. It is important to consider the processes that need to be created to work with the cloud platform and the way students will handle this. Finally the day-to-day running costs and administration required are the final hurdle.

From the motivations discussed in the prior section, surveys were done amongst students and staff, ranking the requirements of a set of features found on cloud platforms and capturing comments on their wider views of what cloud gaming could offer game development students. This survey included management who had to consider the feasibility from a financial perspective. The results are presented in Table 1, where each requirement is given an importance of zero (not important) to ten (extremely important).

From the surveys, it was found that both the developer and player want the bandwidth used to be as small as possible. Other aspects from the survey, specific to the developer and player, included:

1. *Developer Priority:* to be able to upload the game onto the service with minimal work. Once live, be capable of testing with closed user-groups and obtain detailed results when game problems occur.
2. *Developer Secondary:* to perform patching of the game with the ability to switch between game versions without issuing a new patch. Once a game version is selected then players are all moved onto that version automatically. In learning more about the success of the game design and level layout, easily capture gameplay metrics is also important.
3. *Player Priority:* The game plays in under a minute, preferably with gameplay that has little to no latency over that of a natively installed game. Being able to launch a game without installation, or from within a web-page, is necessary for students to embed games in their online portfolios.
4. *Player Secondary:* While having save games stored in the cloud is thought of as important, being able to play the same game across multiple different platforms was not viewed as crucial, mainly due to user interface and performance limitations that different devices present. Respondents found the ability to play while offline less important given that their devices are constantly online and people viewing their portfolios are assumed to be also.

When further inquiry was made around the minimal bandwidth use for game-players, the feedback was that most had internet connections that are servicing a number of people within a household, where several video streams for viewing movies would compete with activities such as streaming games where bit-rates required are high and constant.

These results were also used to help source funding for the project. This led to success in the form of a grant from the marketing department, due to the visibility it would help give game artefacts to the outside world. The project started in 2013 and commenced with the exploration of which particular cloud architecture would best suit our requirements.

Cloud Gaming Platform Selection

Various options were evaluated, including some developed in-house. Two of the most prominent challenges were the scalability and on-going cost of operation. For these reasons, the use of a cloud-gaming video-streaming solution was ruled out. Although video-streaming a game can make it possible to play the game on many different types of devices (from your mobile phone to your Smart-TV), it was found that often the interface difference between devices made the gameplay experience not as good as on the original device it was designed for. The introduction of extra latency would have caused games that are real-time and sensitive to latency approach 100 ms or more, such as a First-Person Shooter (FPS), to become less responsive, and thus the player experience suffers (Beigbeder et al. 2004).

Table 1: Cloud Requirements

Requirement	*Who*	*Importance (0-10)*
R1. Porting Game to Cloud is Simple	Developer	9
R2. Game Bug Reporting	Developer	9
R3. Game Metrics Reporting	Developer	7
R4. Game Patching in the Cloud	Developer	8
R5. Simple Game Submission Process	Developer	9
R6. Management of Alpha/Beta Release Groups	Developer	9
R7. Gameplay Latency is Low	Player	8
R8. Plays on Multiple Platforms	Player	5
R9. Game Plays without Installation	Player	8
R10. Game Plays Immediately	Player	5
R11. Game Plays in under a Minute	Player	9
R12. Game Launchable from Web-page	Player	8
R13. Save Games are in the Cloud	Player	7
R14. Game is Playable Offline	Player	5
R15. Bandwidth Used is Minimal	Both	9

Creating a Cloud Game Channel

One of the aims was to create the first cloud-based University games channel. The motivation was to make the history of games created at the University easily available for everyone to access and play. Before this, the history of games created were archived, not easily retrieved, and faced issues with not being playable due to dependencies on legacy hardware or the failure of the storage media such as CD-ROMs and DVDs.

The current University infrastructure was explored, but both the in-house inter-department costing for server space, time and bandwidth were prohibitive especially for an ongoing video streaming setup. Even when open source technologies are employed, such as *GamingAnywhere*, the results of studies find it challenging to set them up to work effectively (Xue et al. 2015). Furthermore, the security requirements also created some challenges in our tightly firewalled and protected environment. This meant leveraging cloud infrastructure from service providers such as Amazon's *AWS* (2016) or Microsoft's *Azure* (2016) for example. Again, costings for a video streaming solution were projected, and even with conservative usage it was not sustainable for the University, while the use of file streaming was feasible.

Given that the requirements favoured minimising a continual high use of bandwidth, the need for low latency and a willingness to accept some game caching time before a game starts, then the best choice was the use of asset or file streaming, as discussed next.

Streaming File Systems

The transfer of files can be defined by the developer, determined by the levels of a game, anticipated statistically or via prediction methods on what assets or files are required and when. The aim for all these approaches is to minimise the size of the initial download before being able to play, as well as effectively stream in the background without negatively impacting the game (see Figure 3). Sometimes file streaming is done at the block level of a file, where chunks of the file are sent and not all of a large file is sent if it is not required, which can be especially useful when it comes to efficient patching. The streamed game then runs natively, or with a minimal layer that handles the file requests, and therefore is not impacted by latency, but is restricted to the end-user device and operating system the game was originally designed for.

Some services will create a Virtual Machine (VM) environment for the client side, which supports file streaming. This approach may introduce a level of hopefully unnoticeable latency, but can enable playback of the game on different devices. Nowadays there are a number of platforms in the cloud gaming space (see Table 2) that utilise asset streaming technologies and techniques in providing their service.

Table 2: Cloud Gaming Services (File Streaming)

Platform	*Stream Type*
AirConsole, airconsole.com	Files
AWOMO (GDI) [b]	Files
Exent, exent.com	Files
Happy Cloud, thehappycloud.com	Files
InstantAction (GarageGames)[b]	Files
Kalydo (Eximion)[b]	Files
Numecent (Approxy), numecent.com	Memory/Files
Steam (Valve), steampowered.com	Files
SuperStreamer	Files
Triton/Game xStream (DiStream)[b]	Files
Utomik, utomik.com	Files

[b]Service discontinued.

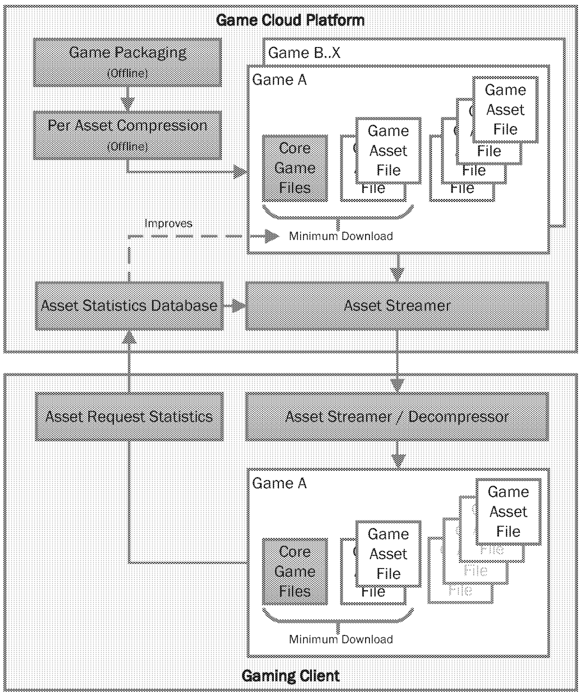


Figure 3: An Architectural View of Cloud Gaming and Asset/File Streaming

Each file streaming service develops custom features to help differentiate, focus and support the desired target markets. Services such as *AWOMO*, *InstantAction*, and *Triton* are file streaming game services that are no longer in operation today. Game Domain International (GDI) used a statistical approach for a *A World Of My Own* (AWOMO) (Walkden 2009) where games would be provided in a Second-Life (2011) like world that intended to be the best place to buy and sell games (The Steam Review 2007). *InstantAction* focused on being able to deliver games directly within a web-browser from websites such as Facebook for example (DiStream 2004; Jurgelionis et al. 2009; Walkden 2009). *Exent*, which had been involved with the *Games@Large* cloud gaming research (Jurgelionis et al. 2009; Nave et al. 2008), primarily relied on file streaming technology layer for delivering in-game advertisements in making the games freely available.

Eximion provided the cloud gaming tool-set called *Kalydo* (Eximion 2006), allowing the packaging of an existing game so that it could be progressively downloaded, enabling players to commence gameplay as soon as the minimum number of files required were available. Access to the game could be provided directly through a user's web browser using a small plug-in. Games, such as the Massively Multiplayer Online Role-Playing Game (MMORPG) *Runes of Magic*, normally required a download of over 8 GB, but with this technology the game started playing after downloading only an initial 200 MB.

SuperStreamer was a funded project that was prototyped in the open-source Unreal Engine 4 to allow developers to select a minimum download set of assets to start the game, and then stream assets predicting when they are required based on the spatial distance and number of times the texture is used. The fallback for not having the texture on time is the use of a low resolution version that is part of the initial minimum download. Using *SuperStreamer* a game of 1340 MB (688 MB compressed) required a minimum initial download of 139 MB (434 MB uncompressed) to commence playing, this being five times smaller achieving an approximate 30 percent saving on the game loading time (Eu et al. 2016).

Valve's *Steam* depends on file/asset streaming at the core. Using a Distributed File System (DFS), the structure includes manifest objects that determine the order of when files are required to be cached locally. If multiple players share a device then *Steam* will share the assets of the game as well. Also a move to deliver games, not just to the PC, but also to devices such as Smart TVs required that the developer's game be compatible only with a selected game controller (Newell 2003; Valve 2011).

An even more effective way to deploy a file streaming solution is to use a *player prediction* based approach for file streaming as done by *Utomik* (Tops 2015). This captures player behaviour to build a predictive model for what files are going to be required when. The aim is to minimise the download size through better prediction given the results from more and more players having played the game.

AirConsole is a browser based service that started in 2015 that uses your smartphone as your game controller and turns your Smart TV, or web-browser, into the console. Due to the system relying on web-browser technologies, these games are written with tools like Flash, HTML5, Construct 2 or Unity3D, using their WebGL build option where game content is streamed as files to the web-browser cache. The service has games that range from 1 to 8 players with a console like party game focus, where a subscriber's privileges are shared with other players in the same session (N-Dream 2015).

AirConsole has been active with the *Global Game Jam* and with competitions encouraging content creators to create or port games to the platform. The technique is the smart use of existing technologies, although feedback on how modern day smartphones feel as a controller are mixed due to the current lack of haptic feedback these devices currently offer and some games may demand. *AirConsole's* use of mobile devices as controllers presents a latency challenge that they approach in reliable and latency minimising steps of first using *long polling* where controllers keep open web requests answered when the server has new information to send, and if possible change to *WebSockets* or the least latent option *WebRTC*.

Streaming File System Features

The success of the platform depends on being able to attract both a strong community of game players as well as a variety of good game content from providers. For developers and players, the supporting eco-system around platforms like *Utomik* and *Steam* presents cloud features that makes these platforms more attractive. These features can include:

- Error capturing, reporting, core dump and video replay,
- Version management and automatic patching/updating,
- Player save games handled and available across devices,
- Downloadable Content (DLC) support in and out of game,
- Player profile to capture achievements, define avatar and compete on leaderboards,
- Player network features for friends, sharing events and activities,
- Community news and game promotion services,
- Support for game or service issues, both in and out of the game,
- Staging secure release areas such as closed Alpha/Beta test groups,
- Gameplay metrics on who and how the game is played,
- Matchmaking for multiplayer games,
- Tools to manage bad player behaviour and capture cheating,
- Cross-platform delivery of the game with minimal or no developer changes,
- Game communications for friend voice/video communication, and
- Handling of sales, in-game micro-transactions and reporting of the data.

File streaming solutions aim to run natively and perform as though the user has installed the game once past the initial download requirement. The transfer of game assets at the file level can utilise specialist offline compression approaches that best suit each file-type and maximise the compression ratio. The use of a program to manage the streaming can vary from a lightweight application to manage the delivery of the files, through to a complete virtual machine which impacts performance, but also may allow for better security and support of games on a wider variety end-user devices and operating systems.

Discussion with several service providers at the start of the project led to a partnership created with *Utomik*. They were found to be open and supportive to the idea of a University channel upon the game service. This led to the deployment of eight games from our archive that passed our developed delivery process and packaging for the service.

The year after implementation larger titles were deployed onto the service from our yearlong projects, that including *Blank Slate* and *Heroes & Heretics*, in comparison to our existing smaller arcade like titles such as *Spaceship 724* using *Unity 5* (2016). With the completed launch of the first University cloud games channel the results from the performance of the underlying cloud architecture is presented in comparison with commercial titles also on the same service.

RESULTS

The result of our evaluation and selection was to choose an approach that streams game assets by using a prediction system to determine which of the assets are required and when. This means that a new game on the cloud platform will first download the entire compressed game to the initial players. Once a few people have played the game the information on what asset is required is accumulated and can start to be predicted. The more people play, the more this improves the minimum download size to start the game. The information on the internal operation of the *Utomik's* prediction system remains proprietary.

Results - General Operation

The deployment of a game with the *Utomik* tools allows for the selection of the mandatory game components, such as the executable, and separate packaging for content that can be streamed later, such as level content. The tools allow for private testing of the game packaging both offline and online.

Operationally, the intention is for a group of test users to play the game to verify the quality of the game and also provide the initial player data for the minimum download to be established. The prediction system can quickly obtain a minimum download size that is close to optimum after a single play session of the game, this dependent on the game having a single path through the experience. When the game has multiple paths this may take longer, especially in a large MMO where the directions taken vary and the starting points may differ. Once released to the general public, the information from all the game sessions helps improve the minimum download size further. The results will be presented from the performance of student games on the cloud platform followed by an evaluation of how the commercial titles, that are larger and possibly better crafted, performed in comparison.

Results - Student Games

The final results for the tested games varied in download time from five minutes for *Blank Slate* down to just over ten seconds for the smallest game *Spaceship 729*. It was found that once the minimum download size moved beyond 240 MB, the download time more likely exceeded the one minute time deemed acceptable in our survey. This impacted *Blank Slate* and *Heroes & Heretics* (see Table 3).

The worst-case amount to download improved greatly after compression, from a best case of 31 percent of the original size of the game *Spaceship 729* through to 54 percent on the game *Chameleon Cancellation*, done by the use of specialised, per file type, compression approaches (see Figure 4). The overall average compression of the student games came out at 41 percent of the original game size.

The prediction system improved the minimum download size reducing it to an average of 73 percent of the compressed asset size of the games. The difference between the best case to worst case performer in the prediction stage is large, with the compressed size of the game *Chameleon Cancellation* reduced to 56 percent of the compressed size, while the game *Heroes & Heretics* reduced to only 95 percent of the compressed size.

In total it was found that the minimum download size for each game tested is an average of 29 percent compared to the original game installation size, this including both the compression and prediction stages. The range of the minimum download sizes varied per game from the best-case of 19 percent with the game *Spaceship 729* down to worst-case 35 percent on the game *Heroes & Heretics*.

Results - Commercial Games

The results from the evaluation of the performance of commercial games on *Utomik* involved analysing six games from different developers and publishers that each use a variety of unique engine technology. Nearly all games came close to being within a minute to download and start playing where 1.5 minutes was the longest wait time for a game to start. There was one exception with the game *Unreal Tournament 3 Black* taking approximately 16 minutes to cache file assets to be ready for gameplay (see Table 4).

The use of specialised per file type compression approaches achieved an average compression of 80 percent of the original size of the game (see Figure 5). The best case compression of 40 percent of the original size on *Evoland 2*, while the worst-case compression amount of 99 percent, on both *Darksiders* and *Red Faction*, performed poorly due to assets already being efficiently compressed in these games.

The prediction system on the commercial games operated at highly efficient levels where the improvement on the compressed version to the minimum playable download size averaged at 13 percent, with the best case examples being 2 percent on *Red Faction* and 4 percent on *Darksiders*. The prediction worst case of 48 percent on *Unreal Tournament 3 Black*, an outlier given that the other five games all achieved 11 percent and under.

The combined results of both the specialised compression and the prediction system resulted in the minimum download for a playable game to be approximately 4 percent (this being 11 percent if including the *Unreal Engine* title) of the original full game size. The best case examples of 2 percent on *Red Faction* and 4 percent on both *Darksiders* and *Evoland 2*. The worst case of 41 percent on *Unreal Tournament 3 Black*, an outlier given that the other five games came in at 6 percent and under.

DISCUSSION

The results indicate that there is a gap between the compression and prediction performance on the student games when compared to commercial games. The commercial games manage to achieve over 25 percent better minimum download size than that of the student games when we exclude the games made with *Unreal Engine*.

Some of the reasoning behind this is that student games are found to only have one to two levels where, if there are two levels, one is most likely a training level. Typically, there is no restriction on a player going into either level from the start and therefore the minimum download needs to have everything included and ready. One consideration, which needs further investigation, is the possible need for more student training for how the game packaging system is best used. This system highlights the mandatory assets required for the game that must be downloaded and therefore is excluded from the prediction stage. The remaining assets are then monitored by the prediction system to build a minimum download subset, as well as determine what to stream next, for the game to continue without interruption. Importantly, the actual structure of the game itself and design choices about when and how the player will progress through the content appears to have by far the biggest impact on the minimum download for some games.

With the widening eco-system and opening of the entry onto Valve's *Steam*, some of the original challenges highlighted at the start of the project may no longer be as much of a barrier as it was at the beginning. That said, however, the ability for the creation of a University branded games channel is something that would not have been achievable on such a service even today. With the partnership with *Utomik* we have ended up with a result that works and meets the marketing goals of being

Table 3: Asset Streaming Evaluation - Student Games

Criteria\Game	Blank Slate	Heroes & Heretics	See Me	Chameleon Cancellation	Spaceship 729
Engine	Unreal 4.11	Unity 5.0	Unity 4.6	Unity 4.5	Unity 5.0
Genre	Stealth	RTS	Adventure	Platformer	Shoot'em Up
Perspective	3rd Person	1st Person	3rd Person	3rd Person	3rd Person
Download Time[a]	~255 secs	~100 secs	~60 secs	~12 secs	~9 secs

[a]Time to cache and start playing at a download speed of 4 MB/s.

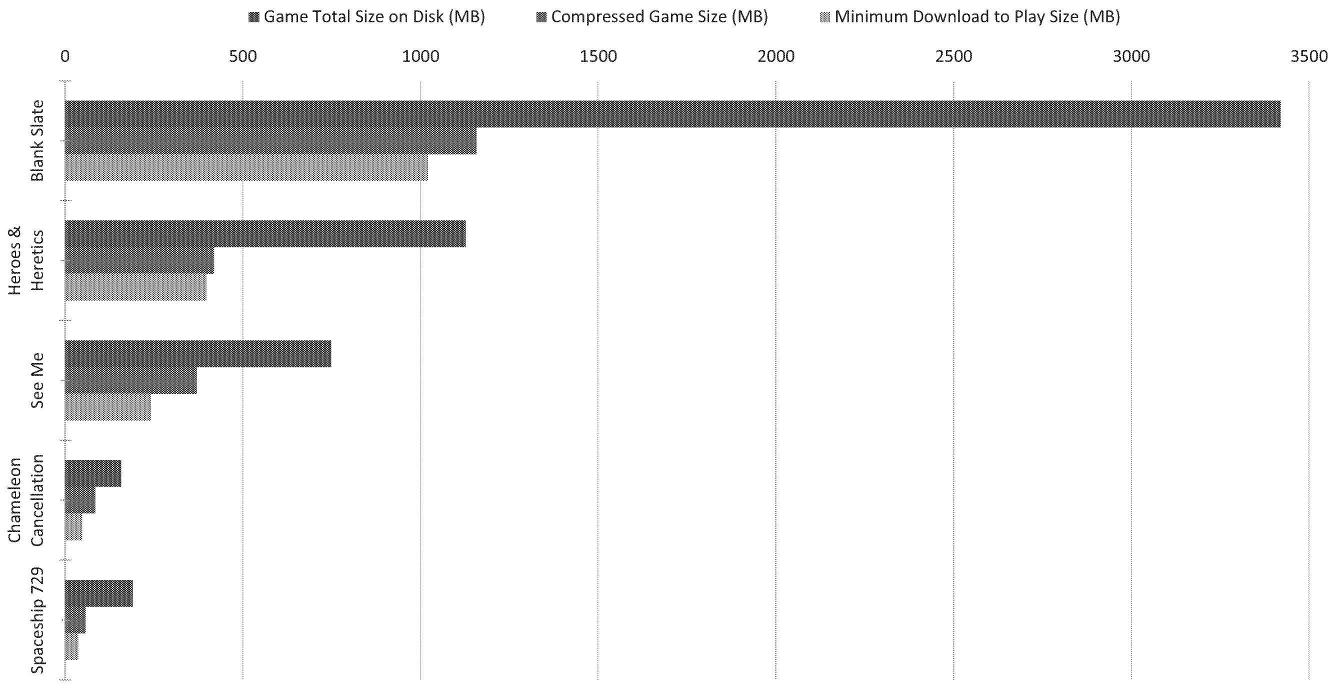


Figure 4: Original, Compressed and Minimum Download Size of Student Games Evaluated

highly visible (see Figure 6) while also meeting a lot of the requirements identified in the survey. The student-created games on the channel are available free to all members of the service.
The selection of a file streaming approach allowed for games of a certain quality threshold to be selected and added to the cloud games channel. Games play just like they were installed locally, therefore latency eliminated as a concern during gameplay. The resulting operation from the first year of operation resulted in the delivery of two major games with one winning awards.

Comparison to Commercial Games

Due to the size of the student games, where engines used were *Unreal Engine 4* or *Unity*, and the structure of the game is one to two levels at most, we wanted to compare the performance of larger commercial games already on the *Utomik* service to gain further insights into the performance of the compression and prediction systems.

For some engines, such as *Unreal Engine*, assets are already compressed within large files, these being called *pak* files. As a result the compression stage has less of an impact on these files. Interestingly, this engine organisation and compression of game assets could be disabled and compared to the resulting compression of the deployment system instead. It would also be interesting to investigate if this would have any benefit to the prediction system or not by placing them outside of the *pak* file.

Games from Indie developers are less likely to have assets packed into fewer files and tightly compressed. With these games the compression stage is highly effective. This highlighted a key to the system in that, with file streaming, not only does it make use of existing Content Delivery Network (CDN) infrastructure, but also allows for significant savings in bandwidth by the use of custom, finely tuned compression algorithms and codecs

Table 4: Asset Streaming Evaluation - Commercial Games

Criteria\Game	Darksiders	Unreal Tournament 3 Black	Red Faction: Armageddon	ArcaniA: Gothic 4	Overlord: Raising Hell	Evoland 2
Developer	Vigil Games	Epic	Volition	Spellbound	Triumph	Shiro Games
Publisher	THQ	Midway Games	THQ	JoWoord	Codemasters	Shiro Games
Engine	Internal	Unreal Engine 3	Geo-Mod 2.5	Vision 7	Internal	Internal
Genre	Action RPG	Shooter	Shooter	Action RPG	Action RPG	Action RPG
Perspective	3rd Person	1st Person	3rd Person	3rd Person	3rd Person	3rd Person
Download Time[a]	~89 secs	~963 secs	~50 secs	~91 secs	~73 secs	~8 secs

[a]Time to cache and start playing at a download speed of 4 MB/s.

that can process offline. This means that the time invested into maximising compression will not have a direct impact on gameplay latency, as they would for video streaming.

Minimum Download Size and Bandwidth Usage

The main advantage of the solution is the impact on minimising bandwidth usage where once assets are streamed there is no longer a continual high bit-rate stream of data as found in a video streaming solution. The solution also avoids the demanding requirement to run each game instance on our own server farm. Perhaps surprisingly, there were significant bandwidth savings just with the file-level offline compression.

The comparison between the commercial titles and student titles indicates that although the commercial games were on average twelve times the physical install size of the student games, the minimum download size average for a commercial game to be playable was only 68 MB larger than that of a student game. This result indicates the effectiveness of the file streaming prediction methods becoming more evident as a game title becomes larger.

The prediction also had challenges with some of the commercial games which could also be seen with the smaller student games. For games that allow a player to jump into any level from the start, in other words there is no strict linear progression through the game levels or locking of certain levels before others are played, will benefit least from any smart approach to ordering the delivery of assets to minimise the initial download.

A multiplayer game where any level can be immediately selected next to play is an example of such a game where, for performance reasons, the engine is constructed to have all level assets loaded before playing. For these games all levels need to be immediately accessible and therefore makes the minimum download required larger than that of other games examined. For multiplayers where the server joined contains a defined level play order, there is then the possibility for further advances to the prediction system in using this information to order content download accordingly. The prediction system will also perform well with large open world games, such as MMO games where the engine streams assets in and out as needed, especially when player paths through the world follow similar patterns.

Given that student titles tend to be smaller in size, have only a couple of levels to play (where sometimes one of those is a training level) and all levels are immediately playable with no unlocking of levels in a sequence, then not much ends up being left for the prediction system to do in reducing the minimum download further beyond that of the specialised compression (see Figure 7). Similiarly student-developed games tend to use a lot of unoptimized assets in the games created, poor use of compilation options, as well as the inclusion of components that are not required. The result is that the game ends up considerably larger than it needs to be in the first place. This highlights that attention needs to be paid to these particular developer skills and knowledge, which can be of benefit to all areas of game creation and especially useful in the development of mobile games.

Traditionally, with full game download and install, game-players may quit even before they have completely downloaded the game. Once downloaded and installed, there may be only minutes of gameplay before a game-player takes a dislike to the game and quits to never play it again. In both cases this results in a waste of downloaded content where 90 percent or more of the game is never seen or used. With the student games, even though they tend to be a lot smaller than 10 GB of STO, the bigger student game example *Blank Slate* would potentially save 1 GB to 2 GB of download each time this might occur.

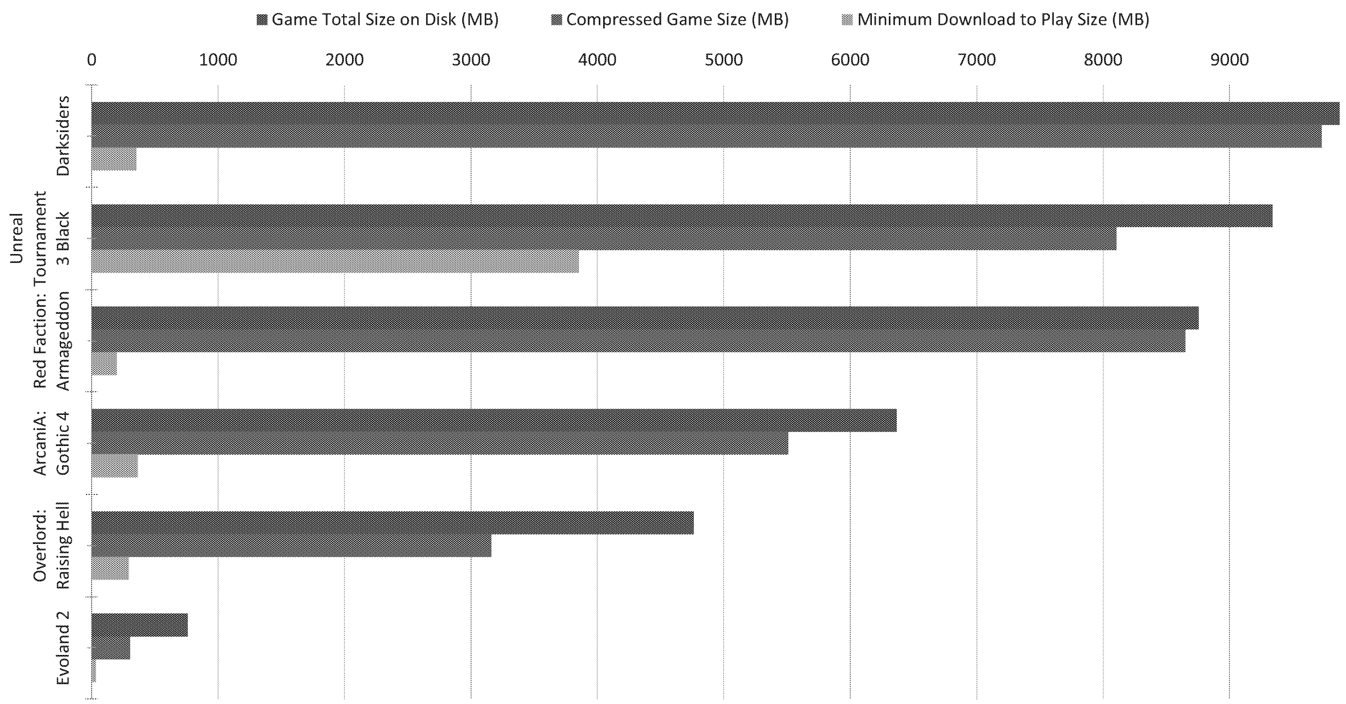


Figure 5: Original, Compressed and Minimum Download Size of Commercial Games Evaluated

Figure 6: The University Games Channel Presenting Student Titles for Selection

Download Timing and Automatic Installation

The final download timings for two of the five student games tested fell under the one minute caching time thought important by survey respondents for not having players turn away. For the three larger games one fell just over a minute to commence playing while the other two games needed two and five minutes to be ready to play. Out of the commercial titles, nearly all of them came in under one and a half minutes with two games playable well under a minute, with the exception of the game *Unreal Tournament 3 Black* which requires all levels be available immediately.

With larger game titles requiring more than a minute to download, there are still ways to help minimise this time through considerations such as how levels are constructed, aspects like streaming the assets in during the player's progression through the level, and knowing that the player must pass through certain checkpoints in sequence would form the most ideal situation for the prediction model to effectively reduce the minimum download further. This will not always be possible depending on whether the design of the game allows players to follow many different paths, or for performance reasons needs to load all assets into memory before starting to play a level may be the design choices made, and therefore be unavoidable.

The automatic installation management worked well and completely removed the need to go through installation steps and management of games. At least 30 seconds to a couple of minutes were saved by avoiding manual installation steps. This time saving on top of the minimum download time saving, in total, becomes substantial. Being able to free space on the client device is also easy with a section to review what is stored, the space currently in use, and if required a quick one-click removal of games from the client device.

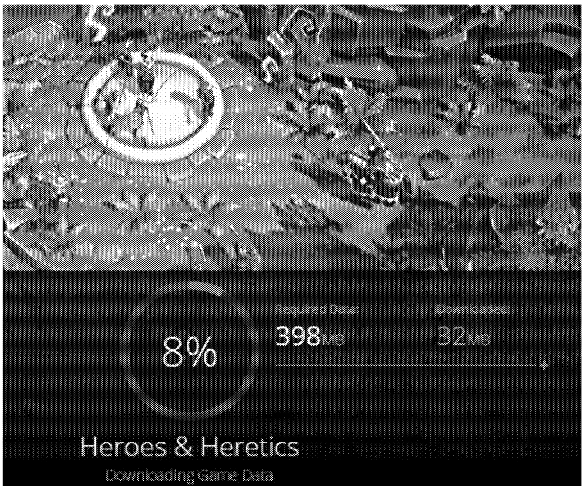


Figure 7: The Student Game *Heroes and Heretics* Streaming the Minimum Download to Start Playing

Version Management and Patching

One feature that worked effectively was the version management system. Through the web console it is straightforward to enable a particular build as the active version of the game. In the background, the system smartly streams only the required assets to match the version if the client does not already have them. For the student games this is quite an important feature, useful in closed group and beta testing, to help roll-out new improved versions without having to deliver large patches and rely on users to install them. This also helps safeguard that all games are in sync and therefore not subject to problems from players running incorrect versions of the game.
The prediction system is aware of the updates and patches delivered. The system will always include the history of statistical information collected from older versions in the way it performs. That said, there is a preference to the prediction system gives to the most current version of assets and the game behaviour that comes with it.

Production Life-Cycle

For students who released on the *Utomik* platform, they achieved experiencing a larger segment of the game development life-cycle. Ultimately, in the first year using this approach, the milestones of these year long 3rd year projects left not as much time as we would have liked to manage the game post release. It was found, however, that this had more to do with teacher planning than students' milestones.

Porting to the Cloud

The tool, process and supporting documentation made the steps of preparing the game for the cloud an uncomplicated experience. The process was not so much porting but just packaging the game. Important steps were done to tag files that are the mandatory components that must always be contained in the minimum download. The staging steps allowed the student developers to then test the packages in an offline test environment. With students assuming that developing and releasing 64-bit builds is the direction to go, students encountered a challenge. At the time the cloud platform worked with only 32-bit builds which supports a wider game playing audience in being able to execute the game. To address this, an immediate adjustment on the cloud platform to also support 64-bit builds resolved this situation. That done, it would remain expected that student teams provide both 32 and 64-bit builds so as to support players on 32-bit systems (see Figure 8).
Some challenges had been created from students not packaging the game properly. Part of this was addressed by having hands-on workshops aimed at helping students who were new to the tools which was positively received when run. For future games, the submission approach will always include a review of the packaging done by performing test streaming locally before uploading.

Figure 8: Student Game Blank-Slate (64-bit) after Launching

Submission Process

An important part, largely underestimated at the start of the project, involved the setup of a submission process for students to deliver games. At first, old games retrievable from the archive and playable were then evaluated. This meant that the game product played with a level of robustness, included the correct logos, and credits for all team members involved. Promotional items such as screenshots and a game trailer, that meet a set of requirements, all needed to be supplied. This helped us refine our own submission process for students to later follow.
This led to a formalised submission process needed to check that the game met a list of requirements and is running robustly enough for packaging and deployment to the cloud. As a step in the process, the collection of

information about the title is important for the proper description and credits made for the game and all who helped develop the game (see Figure 9).

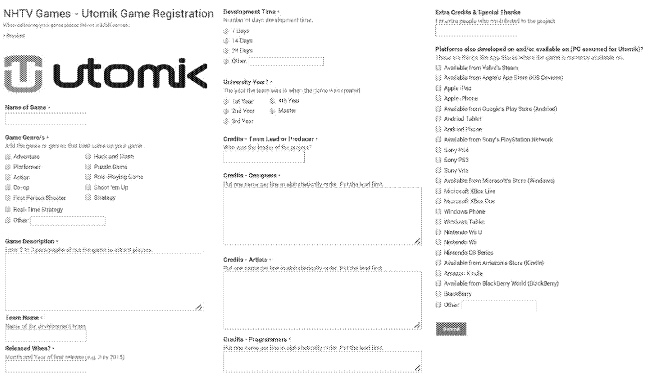


Figure 9: Part of the Submission Process for Games on the Channel

Although the aim is to create a submission process that had a lower bar to that of, say, the console marketplace, we still wanted students to experience the challenges of a submission process nonetheless. The aim remained to emulate, as close as possible, a process in industry but at a scale suited to the development time and product size feasible for the student teams. The output was to validate that games are correctly specified and credited so that game players and possible industry visitors understand who was involved and how it was developed, which is especially important when it comes to the short development time had for these games (see Figure 10).

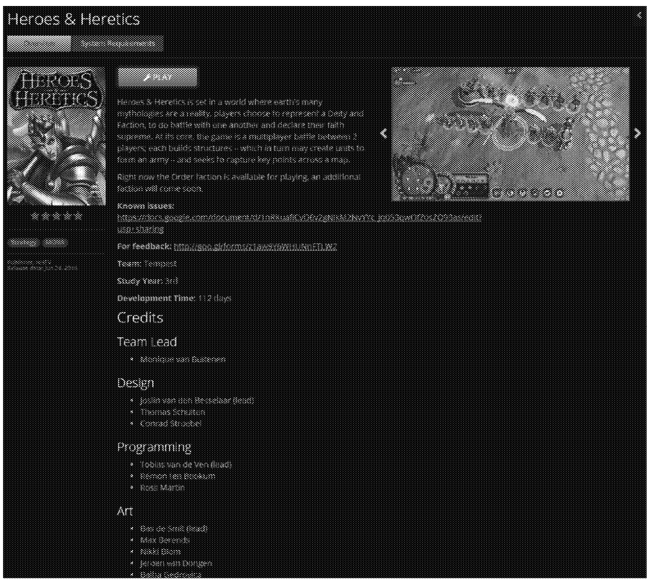


Figure 10: Game Information Giving Credits and Details About the Game

Reporting in the Cloud

Once the game is live then students and developers want to know as much as possible about what is working and what is not. The game crash capture and reporting worked well, with statistics presented on the frequency, to quickly identify priority of what needs to be addressed. This is important especially for student games that tend to be less robust.

Game metrics for who is playing, the player profile and how long they played are on the cloud platform. The students found this information useful in seeing whether games were hitting the targeted demographics and also achieving the expected playtimes and replayability aimed for. The game metrics from the actions of players in-game is an aspect not done yet purely on the cloud platform. Instead, students created their own server services to capture game events and data. From this, they would then produce their own game analytic tools, such as player activity heat map visualisation for example. Some student teams would also use more conventional feedback methods, such as an online survey linked from the game description page.

CONCLUSIONS

The results indicated that the file streaming solution is free of the latency issues during gameplay and is bandwidth effective, where the minimum download to begin playing on this platform is reduced by a factor of 25 to 1 for commercial games and 5 to 1 for student games. The download size of the commercial games, that were on average eleven times the size of the students' games, contributed an average of only 68 MB more to the minimum download for the commercial games. This demonstrated how well the two staged approach of specialist offline compression followed by the use of a prediction system helped in ensuring players were playing as soon as possible.

Importantly, the packaging of games for the *Utomik* service is a straightforward process requiring no development or porting, this being one of our requirements, although having an understanding of how the service operates helps to obtain the best results. This can be seen with the student game performance in comparison to that of commercial games on the service. Some of this is unavoidable when the student games are very small and contain only one or two levels where all content is required at once to play. There are, however, design choices that can guide the player through a game and gate the release of levels and content, or where the engine itself is architected to stream content in as the player follows somewhat similar journeys, will perform the best with the prediction stage of the service.

The application of specialist, per file-type compression for the first stage of reducing the overall size of a game is an effective approach, especially when a game does not benefit much from the prediction stage of the system. Games include multiplayer games such as first person shooters, which present one of the biggest challenges. This is particularly the case in games where any level can be selected, and once selected, the engine expects to load all content for the level at once. For these games, they will need to download most of the content and therefore rely the most on the first stage of specialist compression to reduce, as much as possible, the minimum download. These are the games most likely to take several minutes to be ready to play compared to only a minute for a comparably sized game but of a different genre.

The need to define a clear *green light* like process to prepare and review student titles for the *Utomik* service, along with targeted training that's covering the choice of compilation options and engine settings, asset type and specifications, along with the right decisions on the structure of the game itself are all just as important for areas of mobile development as they are for achieving good results on a cloud platform. Special attention needs to be paid to the size of the game where eliminating asset duplication and controlling the balance between the number of assets and game quality is important. Students need to be trained to understand the considerations of the overall game size which is something they take for granted today given the access to abundant storage space and fast internet connections.

Overall the project of moving our student games onto a cloud platform has been successful and the operation of this service is feasible in terms of costs, bandwidth, and usability. The results achieved games that are now permanently available with just one-click, requiring no installation, and playable in under a minute for smaller games and up to five minutes for large titles that our students produce. This has proved great so far in presenting games at open days, on student portfolios and whenever we need to demonstrate our students' capabilities. While we do not discourage students aiming for the release of their titles on other platforms, we have achieved a place where the barrier to entry is more receptive to our student games, team members are correctly credited and the high profile nature of the University games channel sitting amongst other game developers and publishers has all helped meet the original marketing aims while, at the same time, gaining valuable insights into the challenges associated with cloud gaming.

FUTURE RESEARCH

Further work could be done when it comes to low bandwidth clients where gameplay assets are delivered in lower resolution forms similar to the work done with the *SuperStreamer* research project. The challenge is that if lower resolution textures, models and audio are provided that the engine concerned does not error or fail to hot-load later refreshed higher resolution versions of assets when provided. This means that this type of feature may create porting complications for some game engines to the cloud platform, whereas the *SuperStreamer* example was done within the engine itself to support this approach effectively.

A basic level of cloud analytics is available regarding player profile, play sessions and error information. The platform itself could provide more advanced tools where developers can alter the game to provide statistical information about game progression and produce visualisations such as player heatmaps. For some titles the asset loading information itself may provide more insight into the pathway of progression in the game itself, but this depends on timely nature of how the engine loads the game assets.

At a higher level the cloud platform itself could leverage player gameplay history. Using this information the system could precache the minimum download for games the player is most likely to play next as a result. This feature would be most likely something that the user can control regarding whether the feature is enabled and the size of the cache to be used.

ACKNOWLEDGEMENTS

The authors would like to thank the people of *Utomik* for their enthusiasm and collaboration in making this possible. Specifically Mark Schroders, Chris van der Linden, Doki Tops, Rob van Gulik and Maarten Tops for helping to make it happen. We would also like to thank Harrie van Elderen and the rest of Breda University Marketing for the vision and support along with Will Davis and Fernando Cabello Domingo. And finally would also like to congratulate all the Breda University students that released their games.

REFERENCES

Beigbeder, Tom, Rory Coughlan, Corey Lusher, John Plunkett, Emmanuel Agu, and Mark Claypool (2004). "The Effects of Loss and Latency on User Performance in Unreal Tournament 2003". In: *ACM SIGCOMM* Workshops.October.

Cai, Wei, Ryan Shea, Chun-Ying Huang, Kuan-Ta Chen, Jiangchuan Liu, Victor C. M. Leung, and Cheng-Hsin Hsu (2016). "A Survey on Cloud Gaming: Future of Computer Games". In: *IEEE Access* 4, pp. 7605–7620.

Cheng, Liang, Anusheel Bhushan, and Renato Pajarola (2004). "Real-time 3d graphics streaming using mpeg-4". In: *Proceedings of IEEE/ACM* 1.August 2004, pp. 1–16.

Eisert, P. and P. Fechteler (2008). "Low delay streaming of computer graphics". In: *Proceedings - International Conference on Image Processing, ICIP*, pp. 2704–2707.

Eu, Yong Xue, Jermyn Tanu, Justin Jieting Law, Muhammad Hanif B Ghazali, Shuan Siang Tay, Wei Tsang Ooi, and Anand Bhojan (2016). "SuperStreamer: Enabling Progressive Content Streaming in a Game Engine". In: *Proceedings of the 2016 ACM on Multimedia Conference*, pp. 737–738.

Giesen, Fabian, Ruwen Schnabel, and Reinhard Klein (2008). "Augmented Compression for Server-Side Rendering". In: *Proc. of Interna- tional Fall Workshop on Vision, Modeling, and Visualiza- tion (VMV)*.

Grigg, Robert J and René Hexel (2017). "Communication versus Computation: A Survey of Cloud Gaming Approaches". In: *GAME'ON 2017 18th Conference on Intelligent Games and Simulation*. Ed. by Joseph Kehoe. Carlow, Ireland: EUROSIS-ETI, pp. 10–25.

Holthe, Ole-Ivar, Ola Mogstad, and Leif Arne Rønningen (2009). "Geelix LiveGames : Remote Playing of Video Games". In: *IEEE Consumer Communications and Networking Conference (CCNC)*, pp. 3–4.

Huang, Chun-Ying, Cheng-Hsin Hsu, Yu-Chun Chang, and Kuan-Ta Chen (2013b). "GamingAnywhere : An Open Cloud Gaming System". In: *Proceedings of ACM Multimedia Systems 2013*.

Jurgelionis, Audrius, Philipp Fechteler, Peter Eisert, Francesco Bellotti, David Haggai, Jukka-Pekka Laulajainen, Richard Carmichael, Vassilis Poulopoulos, Arto Laikari, Pekka Perälä, Alessandro De Gloria, and Christos Bouras (2009). "Platform for Distributed 3D Gaming". In: *International Journal of Computer Games Technology* 2009, pp. 1–15.

Liao, Xiaofei, Li Lin, Guang Tan, Hai Jin, Xiaobin Yang, Wei Zhang, and Bo Li (2016). "LiveRender: A Cloud Gaming System Based on Compressed Graphics Streaming". In: *IEEE/ACM Transactions on Networking* 24.4, pp. 2128–2139.

Marpe, Detlev, Heiko Schwarz, and Thomas Wiegand (2003). "Context-based adaptive binary arithmetic coding in the H.264/AVC video compression standard". In: *IEEE Transactions on Circuits and Systems for Video Technology* 13.7, pp. 620–636.

Nave, Itay, David Haggai, Alex Shani, Arto Laikari, Peter Eisert, and Philipp Fechteler (2008). "Games@Large Graphics Streaming Architecture". In: *Consumer Electronics* ISCE, IEEE, pp. 1–4.

Shi, Shu and Cheng-Hsin Hsu (2015). "A Survey of Interactive Remote Rendering Systems". In: *ACM Comput. Surv.* 47.4, 57:1–57:29.

Stegmaier, Simon, Marcelo Magallón, and Thomas Ertl (2002). "A Generic Solution for Hardware-Accelerated Remote Visualization". In: *Eurographics - IEEE TCVG Symposium on Visualisation*.

Xue, Zheng, Di Wu, Jian He, Xiaojun Hei, and Yong Liu (2015). "Playing High-End Video Games in the Cloud: A Measurement Study". In: *IEEE Transactions on Circuits and Systems for Video Technology* 25.12, pp. 2013–2025.

WEB REFERENCES

Ahiska, Dr Bartu (2012). *Approxy*. URL: `http://www.approxy.com/` (visited on 10/20/2014).

– (2015). *Numecent*. URL: `http://www.numecent.com/` (visited on 10/23/2015).

Amazon (2016). *Amazon Web Services (AWS)*. URL: `aws.amazon.com` (visited on 12/04/2016).

BigWorld (2015). *BigWorld Tech*. URL: `www.bigworldtech.com` (visited on 11/01/2015).

CodeGlue (2016). *CodeGlue*. URL: `www.codeglue.com` (visited on 11/13/2016).

DiStream (2004). *Triton (Game xStream)*. URL: `http://www.distream.com/` (visited on 01/01/2014).

Epic (2016). *Unreal Engine*. URL: `www.unrealengine.com` (visited on 12/05/2016).

Eximion (2006). *Kalydo*. URL: `http://www.kalydo.com` (visited on 01/01/2013).

Huang, Chun-Ying, Cheng-Hsin Hsu, Yu-Chun Chang, and Kuan-Ta Chen (2013a). *GamingAnywhere*. URL: `http://gaminganywhere.org/` (visited on 02/03/2015).

Lab, Linden (2011). *Second Life, www.secondlife.com*. URL: `http://secondlife.com/`.

Microsoft (2016a). *Azure Web Services*. URL: `azure.microsoft.com` (visited on 11/28/2016).

– (2016b). *Xbox Live*. URL: `www.xbox.com/live` (visited on 08/18/2016).

N-Dream (2015). *AirConsole*. URL: `https://www.airconsole.com/` (visited on 07/28/2017).

Newell, Gabe (2003). *Steam Explained*. URL: `http://www.valvetime.net/threads/steam-explained.6373/%7B%5C#%7Dpost-132937` (visited on 11/01/2013).

Sony (2016). *PlayStation First*. URL: `www.worldwidestudios.net/about-us/playstation-first` (visited on 09/01/2016).

Stitch Heads (2016). *Kabounce*. URL: `http://www.kabouncegame.com` (visited on 12/11/2016).

Studios, Cryptic (2015). *Star Trek Online (STO)*. URL: `www.arcgames.com/games/star-trek-online` (visited on 11/01/2015).

The Steam Review (2007). *The Secret of AWOMO's Island*. URL: `http://steamreview.org/posts/awomo-rto/` (visited on 12/05/2013).

Through Games (2016). *FRU*. URL: `http://frugame.com` (visited on 12/02/2016).

Tops, Doki (2015). *Utomik*. URL: `http://www.utomik.com` (visited on 09/20/2015).

Ubisoft (2016). *PlayStation First*. URL: `www.ubisoft.com` (visited on 09/03/2016).

Unity (2016). *Unity 3D*. URL: `www.unity3d.com` (visited on 11/25/2016).

Valve (2011). *Steamworks*. URL: `http://www.steampowered.com/steamworks/SteamworksBrochure2011.pdf` (visited on 01/13/2014).

Walkden, Roger (2009). *'What Are You Waiting For?' GDI Reveals How Unique High Speed Video Game Download Technology AWOMO Works*. URL: `http://www.prweb.com/releases/awomo/games/prweb1906194.htm`.

AN AUDIENCE PARTICIPATION ANGRY BIRDS PLATFORM FOR SOCIAL WELL-BEING

Yuxuan Jiang[†] Pujana Paliyawan[†] Ruck Thawonmas[†] and Tomohiro Harada[†]
[†]Graduate School of Information Science and Engineering, Ritsumeikan University, Shiga, Japan
[‡]College of Information Science and Engineering, Ritsumeikan University, Shiga, Japan
E-mail: ruck@is.ritsumei.ac.jp

KEYWORDS

Audience Participation Game, Angry Birds, Procedural Content Generation, Chat, Social Well-being

ABSTRACT

In this paper, we design an audience participation Angry Birds platform and evaluate its effectiveness in promoting social well-being. Existing work reported that playing online games together with other players has positive effects on the promotion of communication and improvement of social relations. With the advent of audience participation games, research on methods for cooperative operations is gaining attention. However, in games targeted in existing research, game rules are complicated, so audiences need time to learn how to play them. Therefore, we focus on the Angry Birds game with simple rules and propose a mechanism to let audiences control bird shots via chat. In addition, the maps, levels, or stages of those games in the existing audience-participation-game studies are fixed, so such games may become boring and make the audience give up playing. Therefore, we also propose another mechanism to let audiences generate original game levels via chat, which is expected to increase fun. Our results confirm the effectiveness of both proposed mechanisms.

INTRODUCTION

Nowadays, more and more people are paying attention to social well-being, such as interpersonal relationship, mental health, and promotion of communication. There are a multitude of applications that aim to solve societal health problems via gamification. In an existing study (Shen and Williams 2011), it was found that online games can help players promote communication with their existing social relations and their family members if they play games together. Our research question is whether audience participation games (APGs), stemming from the current popularity of game live streaming, have similar effects when proper mechanisms are introduced.

In this paper, we focus on Angry Birds and propose two chat mechanisms that transform the game to an APG: the first and the second mechanisms allow audiences to control bird shots and generate levels, respectively. Both mechanisms are evaluated in terms of seven satisfaction factors in Game User Experience Satisfaction Scale (GUESS) (Phan et al. 2016): Usability/Playability, Play Engrossment, Enjoyment, Creative Freedom, Personal Gratification, Social Connectivity, and Visual Aesthetics. According to the results of the conducted user study, the effectiveness of both mechanisms is confirmed.

RELATED WORK

Audience Participation Games

An audience participation game (APG) is a recent game genre where the player or the system receives comments from audiences on a live streaming video platform (Twitch, YouTube, etc.) and determines what to do next based on the received comments (Seering et al. 2017). Twitch Plays Pokemon (TPP), a Pokemon game that incorporates audiences' comments (Lessel et al. 2017a), is one of the most famous APGs. Since live streaming video platforms emerged, it has become easier to distribute gameplay directly to audiences. At the same time, research on methods for consolidating cooperative operations in the APG has been drawing attention. Two independent studies (Chen 2014, Margel 2018) discussed the two comment aggregation methods, anarchy and democracy, for cooperative play in TPP. Anarchy is an aggregation method that executes comments transmitted from the audiences in real time in order and reflects them in the game. Democracy selects and executes the most posted command within a certain period of time. More recent work (Lessel et al. 2017b) discussed how to cooperatively play CrowdChess against AI.

However, in the aforementioned existing studies on the APG, the game rules are complex, making it difficult for audiences to participate because they need to learn complex rules and the related skills. In addition, the game contents, such as maps, stages, or levels, are fixed. As a result, audiences might get bored with such fixed contents, leading to a decline in fun or making them quit playing. To solve these issues, in our work, Angry Birds, a game with simple rules, is adopted, by which many game players or audiences should be able to readily start

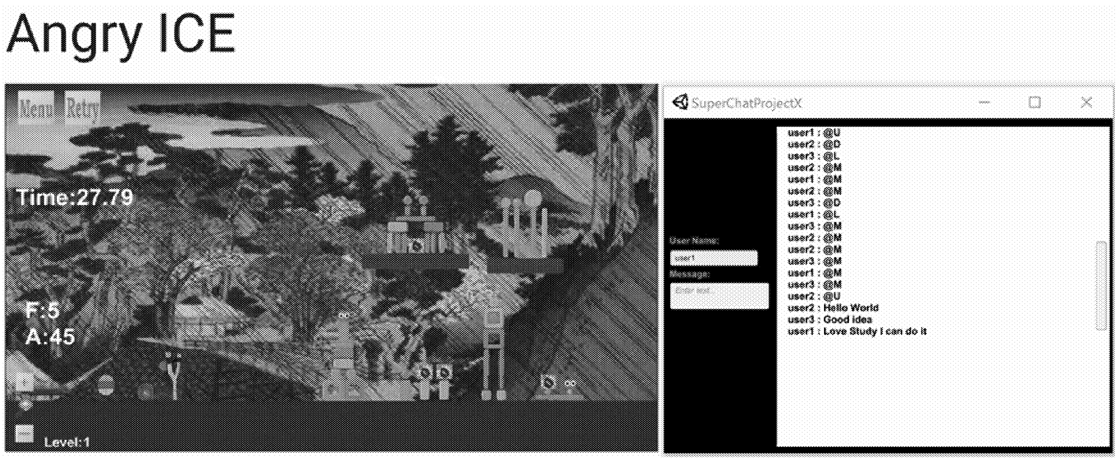


Figure 1: Angry ICE game screen

playing the game with no prior experience or knowledge. Audience-participation mechanisms for controlling bird shots and generating levels, both via chat, are proposed.

Angry Birds

Angry Birds (https://www.angrybirds.com/games/) is a famous action-puzzle game developed by a Finnish company Rovio Entertainment. The first Angry Birds game in the series was initially released in December 2009. The purpose of the game is that of using a slingshot to launch a bird to destroy all pigs in the stage. If all the pigs are destroyed after the last bird is launched, the level is cleared and the next level can be selected. If all the birds run out but pigs remain, the level is failed and can be tried again. To prevent players from boredom, recently there have been active research activities for automatically generating fun levels, described in the next sub-section, and a competition (https://aibirds.org/other-events/level-generation-competition.html).
We base our work on the Angry Birds clone (Science Birds) (https://github.com/lucasnfe/Science-Birds) used in the aforementioned competition and developed by Lucas Ferreira using Unity. Unlike the original Angry Birds game having several game objects, Science Birds' objects are limited to the Bird, Pig, 12 different blocks with three materials (Wood, Ice, Stone), TNT (detonated after a collision, damaging other objects around it) and finally Terrain, the only block type which is neither affected by gravity nor destroyed.

AUDIENCE PARTICIPATION ANGRY BIRDS PLATFORM – ANGRY ICE

System Overview

We create an audience participation Angry Birds platform named Angry ICE, based on Science Birds. Angry ICE is an APG platform that allows participation to the game via comments or messages sent by audiences. By sending a special command as a comment message, an audience can cooperate with others and take part in game control. Furthermore, the most interesting message is selected from the normal – non-command – messages, and the next level displaying the selected message is generated. Details on shooting control and level generation are given later. For live streaming in user evaluation, the background music and sounds are removed. The background, characters, UI and other game objects are replaced by those with our original design. In the game screen, we append the UI for displaying the current force (F) and angle (A) of the current bird, as well as a countdown timer. An example of the Angry ICE game screen is shown in Fig. 1. The system architecture of Angry ICE is shown in Fig. 2.

Cooperative Play Module

The cooperative play module implements the mechanism for allowing audiences to cooperatively play the game via their chat messages (comments). Angry ICE presents the current bird's shooting power and angle to audiences in real time. The bird's force and angle are initialized to 5 and 45, respectively. Their ranges are [0, 10] and [-180, 180]. Audiences are able to send any of the special comments shown in Table 1 to control the bird together with other audiences. The sent comments from audiences will be reflected in the game in real time, but subject to network delays. For each bird, audiences only have 30 seconds to adjust the force and angle. After 30 seconds the bird will be shot automatically according to the latest received values of power and angle.

Level Generation Module

In order to avoid players' boredom stemming from limited or immutable game content, we propose another

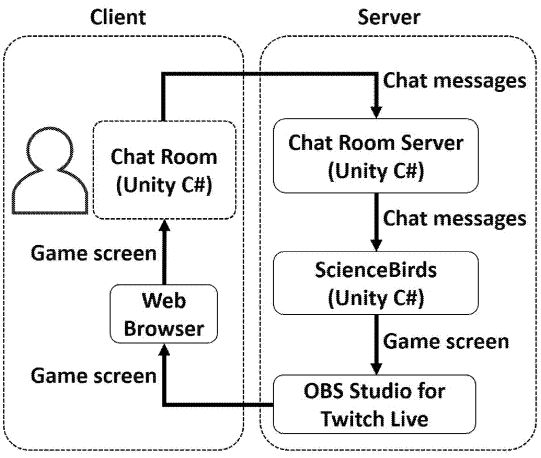


Figure 2: Angry ICE System Architecture

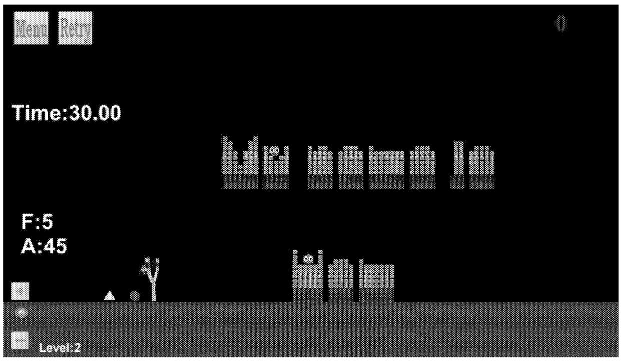


Figure 3: An example of a generated level displaying a humorous message, “My name is Ham,” sent from an audience

module that implements the mechanism for generating humorous-message levels automatically. The purpose of this module is that of promoting communication among audiences through humorous messages and at the same time increasing fun through competition for which message to be selected for the next-level. When the audiences send normal messages other than special comments, the system collects them. All the collected messages during a given interval are then filtered, and those messages containing bad or impolite words are removed. Characters other than alphabetical letters and numbers in the chat messages are deleted. Remaining messages longer than 20 letters (including spaces) will also be truncated to the first 20 letters, because of the limited screen width. After the current level is finished, the Duluth system (Yan and Pedersen 2017) is used to evaluate the humor degree of each of the remaining messages, whose lengths are up to 20 letters, and find the most humorous one.

Duluth is a system that can judge the degree of humor of words and won Subtask B of “SemEval-2017 Task 6 # Hashtag Wars: Learning a Sense of Humor”. Our system uses the Duluth news model which won the aforementioned task to evaluate messages sent by audiences. The most humorous message, the one most deviated from typical news messages, is used to generate the next level via Funny Quotes Generator (Jiang et al. 2017).

Table 1: List of special comments

Comment	Action
@M or @m	Increase the force by +1
@L or @l	Decrease the force by -1
@U or @u	Increase the angle by +1
@D or @d	Decrease the angle by -1

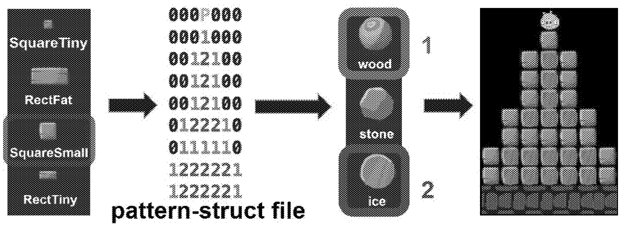


Figure 4: Generation of a character by Funny Quotes Generator

Funny Quotes Generator is an Angry Birds level generator developed previously by us. It won the first Angry Birds Level Generation Competition Fun Track in 2016. It was designed to generate interesting levels by expressing funny words or quotes. In order to make the word level easy to identify, the black background is used in this work. An example of a generated level displaying an audience message is shown in Fig. 3.

In order to make word levels more easily identifiable to enhance the fun of the game, Funny Quotes Generator was also improved to make the word level more recognizable. In order to generate word levels, Funny Quotes Generator originally uses two approaches called pattern-struct and preset-model. The pattern-struct approach generates each time different appearance for one of the alphabetical letters, numeral numbers, or some specific symbols. The preset-model approach is used for increasing the diversity, where models are built in advance for certain capital letters and numbers. Because the preset-model approach cannot be well recognized, compared to the other approach according to our experience, the improved version of Funny Quotes Generator does not use the preset-model approach. For the purpose of enhancing legibility, the pattern-struct approach (Fig. 4) in this work only uses four types of blocks (RectFat, SquareTiny, SquareSmall, RectTiny) and only the material wood and ice to express the letters or numbers.

In addition, due to the limitation of the game screen space, no more than 10 characters per layer in the word level are allowed. If a chat message of interest does not

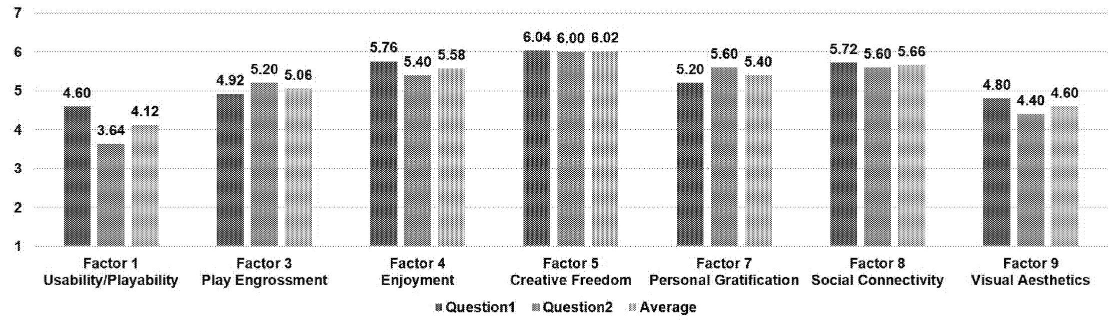


Figure 5: Results of each satisfaction factor of Angry ICE

exceed 10 characters, the message will be configured as a single layer. However, if it is longer than 10 characters, the message will be split into double layers (upper layer and lower layer, each layer up to 10 characters). In order to form a double-layer message, first all possible solutions that can split the message in double layers separated by a space (blank) are found, and if such solutions exist, one will be selected randomly; otherwise, the message will be equally divided to two character strings, one for the upper layer and the other for the lower layer. When no normal message is sent by audiences or all sent normal messages are considered not appropriate, we use a level generated in advance by a baseline level generator provided by the competition organizers as the next level. When using a pre-generated level, from an aesthetic viewpoint, Ukiyo-e (a Japanese traditional style of art) is used as the background. An example of such a level is shown in Fig. 1. In addition, in order to improve the playability of the game level, only pre-generated levels that can be cleared by an AI called Eagle's Wing AI are used. Eagle's Wing AI won the 2017 Angry Birds AI Competition.

EXPERIMENT AND RESULTS

Experiment

We carried out an experiment in order to verify Angry ICE's effectiveness regarding social well-being. In addition, we investigated if there were any issues that required further improvement. This experiment involved 25 multi-national college-student participants – the audiences in the experiment – having the age range of 20-26 with the average age of 22.5.

Each participant first learns how to control bird shooting and level generation via chat messages. Then, we show them a demonstration game. The participants are then asked to answer a questionnaire on their game experience so far and participate in the game for five levels with other participants. During this period, they can freely choose to simply watch or to chat for shooting control/ level generation. In addition, The first level is always a pre-generated level. Each level, either a pre-generated level or a word level, has three birds and two pigs and allows one retry of play. Finally, they are asked to answer a questionnaire about their experience on Angry ICE.

All of the questions in the questionnaire were modified from GUESS accordingly. The aim of their study was to develop and psychometrically validate a new instrument that comprehensively measures video game satisfaction based on key factors, resulting in nine factors. However, because there is no story and sound/music in Angry ICE, due to not in the scope of this research, our questionnaire excluded Factor 2 (Narratives) and Factor 6 (Audio Aesthetics). Each of the remaining seven factors is evaluated in a 7 point Likert scale from 1 (Strongly disagreed) to 7 (Strongly agreed). For each factor, the first question assesses the level-generation mechanism while the second one the cooperative-play mechanism. A total of 14 questions were presented in random order to each participant after they had completed their participation in all the five levels.

Results and Discussions

The results of each factor are shown in Fig. 5. We can see that, except for Factor 3 (Play Engrossment) and Factor 7 (Personal Gratification), the level generation mechanism receives higher evaluation than the cooperative play mechanism. Moreover, there is a statistically significant difference for Factor 1, p-value $= 0.02$ for a Wilcoxon Signed-Rank test.

For Factor 3 (Play Engrossment), the time given for adjusting the direction of the bird to be shot was 30 seconds. In order to clear the level by shooting the bird at a proper location, audiences had to adjust the bird's force and angle accordingly. This made most of the audiences focus on collaborative play and thus be more immersed. For Factor 7 (Personal Gratification), after a level is finished, only the most humorous message sent is used to generate the next level, while the other messages are not accepted. As a result, the level generation mechanism is of slightly lower evaluation for this factor. For Factor 5 (Creative Freedom), both mechanisms received very high evaluation score. This indicates that both effectively foster the audiences' creativity and curiosity. For Factor 9 (Visual Aesthetics), the word level generated by Funny Quotes Generator is more visually appealing than those generated by the baseline level-generator.

From our other work using Science Birds without the two chat mechanisms deploying 20 participants (Yang et al. 2018), Factors 1 and 3 and 4 scored 5.82, 5.58, and 5.78, respectively. There is a statistically significant difference for Factor 1 between Science Birds and Angry ICE (p-value < 0.01 for a Mann-Whitney U test). For Factors 3 and 4, Angry ICE's evaluation is slightly lower than Science Birds. However, Science Birds and original Angry Birds do not have functions that allow collaborative play game and that generate levels based on audiences' messages. Therefore, Factor 8 (Social Connectivity) could not be evaluated. We note that the score of Angry ICE for Factor 8 is the second highest and that according to previous work (Shen and Williams 2011, Hall et al. 2013, Spillers and Asimakopoulos 2014, Hamari and Koivisto 2015), this result indicates the potential of Angry ICE to help promote communication and improve relationships as well as mental health, subject to further verification through clinical studies.

For Factor 1 (Usability/Playability), the cooperative play mechanism in Angry ICE is more difficult than the level generation one, while having much lower evaluation than Science Birds. Our reason for this is that in Science Birds the players only use the mouse to play while in Angry ICE they need to collaborate the control with the other players. Furthermore, due to network delays, those command-messages cannot be reflected in the game in a real-time manner. This is an issue we need to improve in the future.

CONCLUSIONS AND FUTURE WORK

This paper proposed an audience participation Angry Birds platform. The conducted user studies confirm that this platform is useful for social well-being and has the potential for improvement of human relations and promotion of communication via participating cooperative gameplay. Our findings indicate the potential of APGs for social well-being promotion provided that proper mechanisms are introduced. In addition, both mechanisms that we proposed also help promote creativity. Our future work includes investigation of methods for comment aggregation and development of mechanisms to lessen the influence of network delays. We will also improve the graphic and aesthetic aspects in level generation.

REFERENCES

Chen S.D., 2014. *A crude analysis of twitch plays pokemon.* In *arXiv preprint arXiv:1408.4925.*

Hall M.; Caton S.; and Weinhardt C., 2013. *Well-beings predictive value.* In *Proc. of International Conference on Online Communities and Social Computing.* 13–22.

Hamari J. and Koivisto J., 2015. *Working out for likes": An empirical study on social influence in exercise gamification. Computers in Human Behavior*, 50, 333–347.

Jiang Y.X.; Harada T.; and Thawonmas R., 2017. *Procedural generation of angry birds fun levels using pattern-struct and preset-model.* In *Proc. of 2017 IEEE Conference on Computational Intelligence and Games.* 154–161.

Lessel P.; Mauderer M.; Wolff C.; and Kruger A., 2017a. *Let's Play My Way: Investigating Audience Influence in User-Generated Gaming Live-Streams.* In *Proc. of 2017 ACM International Conference on Interactive Experiences for TV and Online Video.* 51–63.

Lessel P.; Vielhauer A.; and Kruger A., 2017b. *CrowdChess: A System to Investigate Shared Game Control in Live-Streams.* In *Proc. of Annual Symposium on Computer-Human Interaction in Play.* 389–400.

Margel M., 2018. *Twitch Plays Pokemon: An Analysis of Social Dynamics in Crowdsourced Games.* In *http://www.cs.utoronto.ca/ mmargel/2720/paper.pdf (accessed on August 10, 2018).*

Phan M.H.; Keebler J.R.; and Chaparro B.S., 2016. *The development and validation of the game user experience satisfaction scale (GUESS). Human Factors*, 58, no. 8, 1217–1247.

Seering J.; Savage S.; Eagle M.; Churchin J.; Moeller R.; Bigham J.P.; and Hammer J., 2017. *Audience Participation Games: Blurring the Line Between Player and Spectator.* In *Proc. of 2017 Conference on Designing Interactive Systems.* 429–440.

Shen C. and Williams D., 2011. *Unpacking time online: Connecting internet and massively multiplayer online game use with psychosocial well-being. Communication Research*, 38, no. 1, 123–149.

Spillers F. and Asimakopoulos S., 2014. *A diary study comparing mobile health applications.* In *Proc. of Design, User Experience, and Usability. User Experience Design Practice.* 358–369.

Yan X. and Pedersen T., 2017. *Duluth at SemEval-2017 Task 6: Language Models in Humor Detection.* In *Proc. of 11th International Workshop on Semantic Evaluations (SemEval-2017).* 49–57.

Yang C.; Jiang Y.; Paliyawan P.; Harada T.; and Thawonmas R., 2018. *Smile with Angry Birds: Two Smile-Interface Implementations.* In *Proc. of 2018 NICOGRAPH International.* p. 80.

CITIZEN'S INTERACTIONS IN "SMART GAME-PLAYING ENVIRONMENTS"

Liliana Vale Costa
Digimedia, University of Aveiro
Campus Universitário de Santiago
3810-193 Aveiro, Portugal
E-mail: lilianavale@ua.pt

Ana Isabel Veloso
Digimedia, University of Aveiro
Campus Universitário de Santiago
3810-193 Aveiro, Portugal
E-mail: aiv@ua.pt

Óscar Mealha
Digimedia, University of Aveiro
Campus Universitário de Santiago
3810-193 Aveiro, Portugal
E-mail: oem@ua.pt

KEYWORDS
'Smart game-playing', Learning, Game-mediated interactions

ABSTRACT
In an increasingly networked society that opens up a set of services and opportunities to meet the citizens' needs, one of the greatest challenges is its complexity towards the global connectivity between individuals and their surroundings. Indeed, the citizen-environment interaction is crucial for designing a 'smart ecosystem' in which the citizens' context and mediated interactions are represented. So far, however, there has been little discussion about the players' spatial mental models and kinesics cues that can foster their interactions. This position paper addresses the use of kinesics behaviours, and space semiotics in game-mediated interactions, aiming at fostering reciprocal affordances between player-citizens and their surrounding environment, for an augmented smarter game-play experience.

INTRODUCTION

The proliferation of smart devices (*e.g.* smartphones, smartwatches, smart bands) that integrate the users' context and their surrounding environment has led to an increasing interest in designing and assessing the impact of technology on daily life (Wright, & Keith, 2014). The game industry has also embraced this proliferation by bringing people together in public spaces through gamification (*e.g. Nike+, Health Month*), augmented reality location-based games (*e.g. Pókemon Go*, *Ingress*) and "hybrid" consoles (*e.g. Nintendo Switch*) (Costa, Veloso, & Mealha, 2018).

However, much less is known about the players' spatial behaviours, shared spaces, relational distance and artefacts, applied in game-mediated interactions - e.g. Greenberg, (2011), Mueller, Stellmach, Greenberg et al. (2014). Taking into account such spaces beyond the use of game conventions (perceived affordances) (Norman, 2013) is crucial in order to meet the intrinsic motivations of the citizen-player to connect, increase social activities and foster internal challenges by interlinking the game-play activity to daily life (the last levels of the Maslow's Hierarchy of Needs – Social and Self-fulfilment) (Maslow, 1943). Furthermore, the Human-Computer Interaction field has been evolving over the years from the mere functional to human-centric approaches (Mealha, 2016) that begin to embody more and more cognitive, emotional and social dynamics (Calvo, & Peters, 2014; Picard, 2000) that are usually dependent on both interactions that happen on-screen and off-screen.

This paper is divided into two sections, including the Introduction and Conclusion. It begins by setting out the research problem and the rationale for studying the player-citizens' interactions whereas the second section discusses the concept of 'Smart game-playing' and its interlink with active learning and gamification. Section 2 is concerned with the body movement (kinesics) and the use of signs to indicate position, distance, orientation and movement in on-screen and off-screen game spaces (space semiotics). Finally, the paper concludes by discussing the implications for designing 'smart game-playing' environments that reinforce the player-citizens' interactions within the ecosystem.

1. LEARNING IN A "SMART GAME-PLAYING" ECOSYSTEM

In broad terms, a 'Smart ecosystem' can be defined as an environment in which "individuals that take part in the local processes have a high level of skills and, at the same time, they are also strongly motivated and engaged by continuous and adequate challenges, provided that their primary needs are reasonably satisfied." (ASLERD, 2016, p.3). In this definition, the following game elements can be highlighted: the player-citizens' skills ("high level of skills"), motivation ("strongly motivated"), challenges ("adequate challenges") and a game-playing experience that improves the players-citizens' wellbeing and quality of life ("needs are reasonably satisfied"). Based on the aforementioned definition, we propose the concept of 'Smart game-playing' to refer to the activity of playing that enables scaffolding/different levels of progression and collection of skills, rewards motivated actions and involves the citizens in challenges/missions that can have an impact on society and improve citizens' wellbeing and quality of life.

In the learning context, understanding "What makes things fun to learn?" (Malone, 1980) and matching the gamification elements to the learners' context and needs are fundamental. Figure 1 shows the main game-based elements that captivate learning according to Malone (1980) intertwined with *gamification* mechanics presented by Werbach and Hunter (2012), Kapp (2012), and Zichermann and Cunningham (2011) in order to build the levels, challenges and rewards of 'Smart game-playing'. As can be seen in Figure 1, the main components suggested by Malone (1980) that make games and learning enjoyable are: fantasy, challenge, and curiosity. Thus, it is necessary to clarify what is meant by each concept:

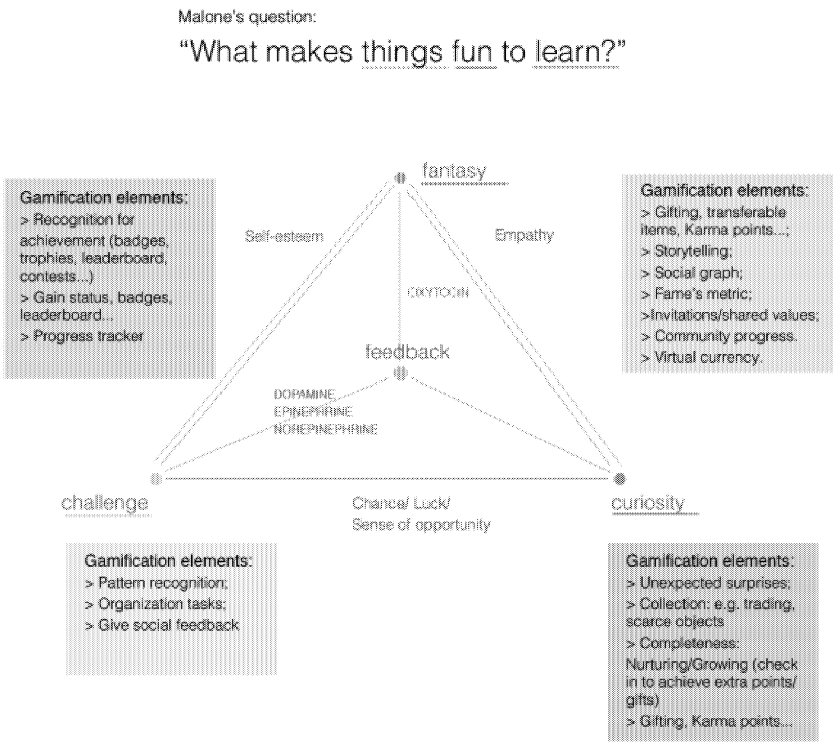


Figure 1. Proposed framework for building the levels, challenges and rewards of 'Smart game-playing' based on the Malone's domains for motivated learning (Malone, 1980) and gamification elements (Werbach, & Hunter, 2012; Kapp, 2012; Zicherman, & Cunningham, 2011)

- *Challenges* are related with a sense of mastery and progression towards uncertain goals and some randomness;
- *Fantasy* is based on schemas and representations from previous experiences;
- *Curiosity* is related to an optimal level of complexity of the learning environment. Indeed, the learning environment should be novel and surprising but not too complex.

The author Malone (1980) also points out that the concepts of 'challenge' and 'curiosity' should be treated separately. However, when players perceive luck, chance or cultivate a sense of opportunity, challenges may be linked with curiosity. In the same vein, self-esteem results from both a challenge with fantasy and fantasy with curiosity.

In our perspective, the feedback element should also be added. In fact, the elements presented by Malone (1980) do not work alone and they are fuelled by feedback. Furthermore, the combination of these elements can activate the neurotransmitters that play an important role in motivation, internal reward (ephinephirne, norepinephirne and dopamine) and emphatizing capacity (oxytocin) (McGonigal, 2011). Regarding the gamification elements, they can be subdivided into the following categories (Werbach, & Hunter, 2012; Kapp, 2012; Zicherman & Cunningham, 2001, Zichermann & Linder, 2013):

- *Challenge*: Pattern recognition, organisation tasks and social feedback tend to be stimulating for the players' brain. For example, the *Match-three* or *Dinner Dish* games are based on pattern recognition and organisation tasks, which triggers brain's activity. Social feedback and helping others can also be internally motivating (McGonigal, 2011);
- *Curiosity*: Curiosity is triggered by unexpected surprises and events. The collection of objects (exploration, scarce objects...), gifting, karma and extra points are some examples (Zicherman, & Cunningham, 2011);
- *Fantasy:* Fantasy is mainly triggered by the storyline. Other elements that can contribute to fantasy are avatars, social graphs, fame's metrics, invitations and shared values in communities, virtual currency, community progress, gifting, transferable items and *Karma* points;
- *Feedback:* Feedback is essential for providing information regarding the players' performance. Indeed, imediate feedback is important for encouraging a habit that will lead to a positive outcome. Badges, leaderboards, progress trackers and points are some of the elements that can provide feedback on the players' actions.

In a nutshell, the concept of 'Smart game-playing' encompasses the different domains for motivated learning proposed by Malone (1980) and the use of *gamification* elements in order to build the levels of difficulty, challenges and rewards. However, the proposed framework seems to be insufficient to involve player-citizens in challenges/missions that can have an impact on society and the inner state of wellbeing and quality of life. Having that said, the next Section discusses kinesics behaviours and space semiotics in game-mediated interactions.

2. KINESICS BEHAVIOURS AND SPACE SEMIOTICS IN GAMES

The game space and the player-citizens' movements towards an activity are also fundamental to design a 'smart game-playing' environment. In *Kinesis and Context: Essays on Body Motion Communication*, the anthropologist Ray Birdwhistell (1970) draws our attention to the fact that the body motion and language are intertwined and these make part of the Human basic needs to feel connected with others and with the surrounding environment. As Birdwhistell (1970, p.48) puts it: "[...] human beings are constantly engaged in adjustements to the presence and activities of other human beings." For Birdwhistell (1952, p.5), Kinesics is defined as "the movements of the human body in its higher level activities as a member unit in the cultural context." A more detailed and most widely-used definition, though, is presented by Poyatos (2002, p.204), in which kinesics is seen as: "Conscious and unconscious psychomuscularly-based body movements and intervening or resulting still positions, either learned or somatogenic, of visual, visual-acoustic and tactile and kinesthetic perception, which, whether isolated or combined with the linguistic and paralinguistic structures and with other somatic and objectual behavioral systems, possess intended or unintended communicative value."

In this definition, there is an emphasis on non-verbal communication and context (that stem from the interrelationship between Humans and the environment). The body movements can also be divided into: innate (e.g. eye blinking, facial flushing); socio-cultural (e.g. expressions, gestures, body language); and a combination of both (e.g. laughing, shrugging the shoulders) (Danesi, 2004). In the case of socio-cultural movements, these are normative and associated to a set of representations that can communicate position, orientation, distance, sequence or movement. In other words, the kinesics behaviours and the use of space constitute a mode of communication and negotiation with the environment and similarly to linguistics that uses signs (signifier) to convey meaning (signified) that is covered in Semiotics (Danesi, 2004, Nöth, 1995), kinesics behaviours (posture/ the body movement/ gestures/ expressions) (signifier) are also used to convey some sort of meaning (non-verbal) (sound-image, signified) and, thus, the relevancy of understanding Spatial Semiotics (Thiering, 2015) to represent spatial cognition.

In the context of games, Table 1 provides an overview in which space semiotics can be applied to games. As can be seen in this Table, different representations of game-play activities already exist in current games and can be extrapolated based on the player-citizen's spatial mental model and the environmental stimuli. That way, such signs as 'facial expression & gaze movement', 'gestures & hand movements', 'posture', 'locative indicators', 'manual contact indicators', 'force indicators', 'partially controlled actions', 'free movement actions', and 'distal and proximal movements' can recognize and antecipate the player-citizens' behaviours (Birdwhistell, 1970; Thiering, 2015). These can also be affected by synchrony, sequence, speed, scale and object placement and quantity in order to augment the sense of 'being there' (McNeil, 1992). Kinesics behaviours and space semiotics can also bring embodied experiences to player-citizens. Indeed, the term 'embodiment' has been used to refer to the physical and social relations that happen in the space, in a sense of being participative by fostering similarity with daily life, familiarity with the sign provided (*e.g.* linguistics, imagetic representations) and the imediacy of acting with different senses (Thiering, 2015, p.31). When the citizen-players engage in these experiences, they can 'know more than we [they] can tell.' (Polanyi, 1966, p.4). They can know through acting/doing, in which the mind and the body are intertwined. This type of knowledge is tacit, proximal, intuitive, experience-based and, thus, difficult to be described (Grant, 2007).

In terms of games that use natural user interfaces or that are based on the players' location, these representations of game-play activities illustrated in Table 1 are also used. For example:

- *Facial expression and Gaze movements*: Facial experessions are explored through the use of camera in order to recall the player what has been his/her player journey. In *Kinect Adventures*, the camera takes photos and shows them during the post-gameplay experience and send the invitation to play again;
- *Gestures and Hand movements*: The gestures and hand movements are particularly relevant when interacting with natural user interfaces or direct and reactable user interfaces, considering that the game activity are dependent on their representation (e.g. grab and throw the ball in Xbox 360 Bowling).
- *Manual contact indicators with Partially controlled actions*: The use of cards in augmented reality games (e.g. *Skylanders Battlecast*) also determines the game inventory, interaction zones and blocking areas with such actions as sorting or dropping;
- *Posture*: Gestural user interfaces in games also indicate the posture in order to activate the game activity. For example, Golf Kinect like other eSport mimetize physical actions that are necessary to perform the activities;
- *Free movement actions:*The gamification app Zombies, Run! aims to encourage the player to run. The player runs in order to avoid zombies chases. The players' movement actions are free and driven by a set of missions and supplies' location;
- *Distal and proximal movements:* In the location-based game Ingress, the end-user has to find portals in his/her city and the location of these spots are related with some architecture and/or art objects (e.g. statues, monuments, historic buildings);

Table 1. Representations used of game-play activities based on the players' spatial mental models

Space semiotics/ Kinesic cues	Description/Examples	Game context
Facial expression & Gaze movements	Facial expressions and eye contact are mostly used to communicate the game's characters emotional states and induce empathy. These expressions are also used in emoticons and mood boards in text messages.	Game characters and Avatars' emotions – *e.g. The Last of Us, Final Fantasy, Life is Strange* Emoticons used, for example, in text messages (MMORPGs)
Gestures & Hand Movements	There are different types of gestures (McNeill, 1992): semiotic, ergotic and epistemic. *Semiotic gestures:* communicate information (e.g. *thumbs up*); *Ergotic gestures:* manipulate physical objects; *Epistemic gestures:* relative to motion that convey information; *Mimesi*s – Imitation of gestures to incite the players to act with a real or virtual object (sequence, accuracy, duration)	Representation and use of gestures in games: *Example of semiotic gestures:* Sonic's foot tap to convey impatience *Example of ergotic gestures: Xbox 360 Bowling, Guitar Hero* *Example of epistemic gestures: Just Dance, WiiFit* Mimetic interfaces
Manual contact indicators	Manual contact indicators are relative to object interaction (interaction zones, configuration, continuity/ blocking areas, proximity). The objects have different properties (*e.g.* texture, shape, weight) that determine the interaction. In games, these properties can be changed – morphing ("Customized") In game missions and rules, the game language used can suggest manual contact – *e.g. 'crafting', 'grab ledge', 'give', 'put', 'carry', 'strum'* Tactile feedback is also used to alert, remind and communicate danger as an analogy to nature – *e.g.* earthquaques	*Reactable games* in which objects are configured, items can be bought and used. There is an exploration of the area accordingly with the position of the object. *Sifteo cubes* – blocks are used that respond to motion and touch *Example of the use of language in game commands/instructions/ missions:* crafting command – *Minecraft*, crafting mode – *Guild Wars 2*; Grab ledge – *Prince of Persia...* *Example of tactile feedback* – game controller shacking
Posture	Players' position (intentionally, non-intentionally)	*Examples:* *Xbox Sports* (player's body oriented to the front); *WiiFit Plus* (balance exercises); *Mario Kart* (players' inclination accordingly with the position of the car)
Locative indicators	The Game Interface often uses references to the place where the game character is relative to the other game elements and environment *Example:* Game map Language used in the instructions: 'in', 'on', 'at' and 'near'	*Example: Assassin's Creed Black Flag* – Map (Representation of the environment, secondary missions – find treasure) Find *Fade Rifts* in *Dragon Age: Inquisition*
Force indicators	The Game Interface often uses references to the force that can be employed *Example:* Animations Language used: 'press', 'jump', 'collide', 'attack' and 'parry'	*Example: Combos in Street Fighter, Jump and other movements in SuperMario, Press up a Joystick*
Partially controlled action	The Game Interface often uses references to partially controlled actions – *e.g.* 'drop', 'toss', 'throw'	*Example: 'Throw' in Angry Birds, Crush the Castle*
Free movement actions	Navigation: Up-Down, Front-Back, Right-Left; Footsteps; Walking	*Example: Following monster's footsteps in Monster Hunter*
Distal and proximal movements	Distal and proximal movements are also used in games in order to approach or distance the game character from the enemies (Game Proxemics)	*Example: Detection of monsters in Pókemon Go (vibration), Red circle in Final Fantasy to indicate that it is an attack zone*

- *Locative and force indicators:* These indicators are a reference to the interaction between the players and their surrounding environment and objects.

Overall, game designers should take these elements into account, when designing different types of mediated interactions, the basic structure and relation of "place" and "digital artefacts and algorithms."

CONCLUSION

This position paper draws our attention to the different domains that are necessary to build a 'Smart game-playing' environment. Indeed, 'Smart game-playing' refers to the activity of playing that enables scaffolding/different levels of progression and collection of skills, rewards motivated actions and involves player-citizens' in challenges/missions that can have an impact on wellbeing and quality of life.

In regards to the development of game levels of difficulty, challenges and rewards, a framework was proposed based on the Malone's (1980) elements for motivated learning and *gamification* elements (Kapp, 2012; Werbach, & Hunter, 2012; Zichermann, & Cunningham, 2011). In terms of the player-citizens' involvement in challenges/missions that can have an impact on society and the inner state of wellbeing and quality of life, *kinesics* and spatial semiotics are discussed.

Kinesics behaviours and the use of space constitute a mode of communication and negotiation with the environment and, therefore, an analysis of representations of game-play activities that already exist in current games and can be extrapolated based on the player-citizens' mental model and environmental stimuli are presented.

Overall, designing a 'Smart game-playing' environment and its inherited domains (Motivated learning and gamification, Kinesics behaviours and space semiotics in games) can meet the motivations of citizen-players to connect, increase social activities and create tension between idealized mental representations and the environment.

In this reciprocal affordance (Gibson, 2015) between player-citizen and the environment that extends the use of game conventions (perceived affordances) (Norman, 2013), it is crucial to invite citizen-player to (re) act to an adaptive and constantly-changing ecosystem with the progress of technologies. The present study provides additional knowledge to the 'Smart game-playing' concept and, thus, leads to the 'perfect smartness' in game-playing – i.e. a complete cognitive immersion by taking into account human interrelationship within the surrounding environment.

Acknowledgments. Research reported in this publication was supported by FCT (Fundação para a Ciência e a Tecnologia) and ESF (European Social Fund) under Community Support Framework III SFRH/BD/101042/2014, the project SEDUCE 2.0 nr. 31696 and the grant SFRH/BSAB/128152/2016 (Fundo Social Europeu and Portuguese financial resources from the Ministry of Science, Technology and Higher Education - MCTES).

REFERENCES

ASLERD(2016). Timisoara Declaration Better Learning for a Better World through People Centred Smart Learning Ecosystems, pp. 1–9 Retrieved from http://www.aslerd.org/inevent/events/Aslerd/?s=222 (Access date: May 29, 2018)

Birdwhistell, R. L. (1970). Kinesics and context: Essays on body motion communication, Philadelphia: University of Pennsylvania press.

Birdwhistell, R. L. (1952). Introduction to Kinesics. Louisville, USA: University of Louisville.

Calvo, R. A., & Peters, D. (2014). Positive computing: technology for wellbeing and human potential. MIT Press, Cambridge, Massachusetts

Costa L.V., Veloso A.I, Mealha Ó. (2018). A Review of Proxemics in 'Smart Game-Playing.' In Ó.Mealha, M. Divitini, & M. Rehm (Eds.), Citizen, Territory and Technologies: Smart Learning Contexts and Practices. Proceedings of the 2nd International Conference on Smart Learning Ecosystems and Regional Development, University of Aveiro, Portugal, 22-23 June 2017, Cham: Springer International Publishing, pp. 201-213. Doi: https://doi.org/10.1007/978-3-319-61322-2_22

Danesi, M. (2004). Messages, signs, and meanings: A Basic Textbook in Semiotics and Communication. Vol. 1, Toronto, Canada: Canadian Scholars' Press.

Gibson J.J. (2015). The Ecological Approach to Visual Perception. New York, USA: Psychology Press.

Grant, K. A. (2007). Tacit knowledge revisited – we can still learn from Polanyi. The Electronic Journal of Knowledge Management, 5 (2), pp. 173-180

Greenberg, S. (2011, September). Opportunities for proxemic interactions in ubicomp (keynote). In IFIP Conference on Human-Computer Interaction – INTERACT 2011. Lecture Notes in Computer Science, vol. 6946, Springer, Berlin, Heidelberg (pp. 3-10). Springer, Berlin, Heidelberg. https://doi.org/10.1007/978-3-642-23774-4_3

Kapp, K. M. (2012). The gamification of learning and instruction game-based methods and strategies for training and education. New Jersey, USA: John Wiley & Sons.

Malone, T. W. (1980). What makes things fun to learn? Heuristics for designing instructional computer games. In Proceedings of the 3rd ACM SIGSMALL symposium and the first SIGPC symposium on Small systems – SIGSMALL' 80, pp. 162-169. Doi: https://doi.org/10.1145/800088.802839

Maslow, A. (1943). A Theory of Human Motivation, Psychological Review 50 (4), 370–396 http://dx.doi.org/10.1037/h0054346

McGonigal, J. (2011) Reality is broken: Why games makes us better and how they can change the world, London, UK: Penguin Books.

McNeill, D. (1992).Hand and mind: What gestures reveal about thought. Illinois, USA: University of Chicago Press.

Mealha, Ó. (2016). Citizen-driven dashboards in smart ecosystems: a framework. Interaction Design & Architecture(s) – IxD&A, vol. 31, pp. 32-4

Mueller, F., Stellmach, S., Greenberg, S., Dippon, A., Boll, S., Garner, J., ... & Altimira, D. (2014, June). Proxemics play: understanding proxemics for designing digital play experiences. In Proceedings of the 2014 conference on Designing interactive systems, DISC 2014 (pp. 533-542). ACM. https://doi.org/10.1145/2598510.2598532

Norman D.A. (2013). The design of everyday things: revised and expanded edition. Basic Books, New York, USA.

Nöth, W. (1995). Handbook of semiotics. Indiana, USA: Indiana University Press.

Picard, R.(2000) Affective Computing. Massachusetts, USA: MIT Press.

Polanyi, M. (1966).The Tacit Dimension. Chicago, USA: The University of Chicago Press.

Poyatos, F. (2002). Nonverbal Communication Across Disciplines: Paralanguage, kinesics, silence, personal and environmental interaction. Vol. 2. Amestardam, Netherlands: John Benjamins Publishing Company.

Thiering, M. (2015). Spatial semiotics and spatial mental models: figure-ground asymmetries in language. Vol. 27. Berlin, Germany: Walter de Gruyter GmbH & Co KG.

Werbach, K., Hunter, D (2012). For the win: How game thinking can revolutionize your business. Philadelphia, USA: Wharton Digital Press.

Wright, R., & Keith, L. (2014). Wearable technology: If the tech fits, wear it. *Journal of Electronic Resources in Medical Libraries*, *11*(4), 204-216. doi: 10.1080/15424065.2014.969051

Zicherman, G.: Cunningham, C. (2011). Gamification by design: Implementing game mechanics in web and mobile apps. Sebastopol, CA: O' Reilly Media, Inc.

Zichermann, G., & Linder, J. (2013). The gamification revolution: How leaders leverage game mechanics to crush the competition. New York, USA: McGraw Hill Education.

ACKNOWLEDGEMENTS

The research reported in this publication was supported by FCT and ESF under the CSF III – SFRH/BD/101042/2014, the project SEDUCE 2.0 nr. 31696, and the grant SFRH/BSAB/128152/2016 (Fundo Social Europeu and Portuguese financial resources from the Ministry of Science, Technology and Higher Education - MCTES).

BIOGRAPHIES

LILIANA VALE COSTA is a researcher at Digimedia and visiting scholar at the University of Aveiro. She holds an European PhD in Information and Communication in Digital Platforms; a M.A in Multimedia Communication and a BSc in New Technologies of Communication. Her research interests are digital games, universal design, virtual communities, ageing studies, learning, human-computer interaction, computer-mediated communication, natural interfaces, eHealth, mobile apps, gerontotechnology and digital inclusion.

ANA ISABEL VELOSO is an Assistant Professor at the Department of Communication and Art, University of Aveiro. Throughout years, Ana Veloso has been supervising many students of master's and doctoral degree levels. She has also been the coordinator of several research projects, such as SEDUCE, EYES ON GAMES, IMP.cubed. Her Ph.D. is in Communication Sciences and Technologies from the University of Aveiro. Before her doctoral studies in 2006, she obtained her MSc in Biomedical Engineering and bachelor in Informatics Engineering from the University of Coimbra. Her research interests include video games, information and communication in technology- mediated contexts, gerontechnology.

ÓSCAR MEALHA is an Associate Professor at the Department of Communication and Art, University of Aveiro. He develops his research in the area of "Information and Communication in Digital Platforms" in the context of "Knowledge Media and Communities" with several projects, masters and doctoral supervisions and publications on methods and usability evaluation techniques and visualization of interaction/infocommunication activity. He is now involved in infocommunication mediation projects such as "Visualization of Open Data Dashboards for Citizen Engagement and Learning" in intelligent territories, e.g. Águeda Smart City, Murtosa cyclabke, ASLERD – aslerd.com.

SERIOUS GAMING

MOTIVATIONAL GAME DESIGN AND PRO-ENVIRONMENTAL ELEMENTS IN SUSTAINABILITY APPLICATIONS

Kaisa Könnölä, Tuomas Mäkilä, Kaapo Seppälä, Teijo Lehtonen
Department of Future Technologies
20014, University of Turku
Finland
E-mail: kaisa.konnola@utu.fi, tuomas.makila@utu.fi, kaapo.seppala@utu.fi, teijo.lehtonen@utu.fi

KEYWORDS
sustainability, gamification, game design, design elements, mobile application

ABSTRACT

Sustainability applications aim at changing behavior to a more sustainable direction ot informing about sustainability. According to our research, information sharing seems to be the key pro-environmental element used in sustainability applications, even though also other elements, such as commitment, comparison, feedback and rewards are often utilized. Badges, points and levels are commonly found game design elements for feedback and reward purposes. Sharing one's action with friends and teams can create commitment and comparison, whereas usage of a story creates commitment through emotions.

INTRODUCTION

Sustainability and its three categories – environmental, economic and social – are ways to ensure that also the next generations have possibilities to enjoy our planet. Even small actions can have an effect on changing the attitudes of people. Many websites and applications share information and advice on sustainability. In this paper, we will take a closer look at the existing sustainability applications and explore how elements from gamification and pro-environmental psychology are used in them.

BACKGROUND

Pro-environmental behavior can be seen as conscious actions taken by individuals to lessen the negative human impact on the environment (Kollmuss and Agyeman 2002). Several theories are commonly used to explain the environmentally sustainable behavior (Froelich 2010: Sawitri 2015). According to the theory of planned behavior, behavior can be predicted from the behavioral intentions. In norm activation theory, the intention is not enough, but also knowledge of consequences, ascription of responsibility and personal norms predict the behavior. The Values-Beliefs-Norms -theory emphasizes the importance of person's values and beliefs.

To motivate pro-environmental behavior, several popular means are used to affect the elements that precede the behavior according to the theories (Froelich 2010). *Information* can be provided in several ways, e.g. campaigns and websites, which are assumed to make people act in more environmentally beneficial ways. Another way to motivate people is *goal-setting*, which can be made by individuals, groups or e.g. external coaches. *Incentives and disincentives* are motivation techniques, which come before the action, whereas *rewards and penalties* are coming after the action. When people express their *commitment,* it increases the probability to pursue that behavior. Similarly *comparison* between individuals and groups can be useful in motivating an action. In order to be effective, these motivation techniques are often used together with *feedback*.

Gamification can be defined to be the application of game design principles for the purpose of engaging users with other products or services than a game (Deterding et al. 2011). Besides engaging users, gamification is used to motivate people to participate and even to create loyalty.

One way to apply and research gamification is through the usage of different motivational game design (Weiser et al. 2015). These elements can be seen to address different motivational perspectives: trait perspective, behaviourist learning perspective, cognitive perspective, perspective of self-determination, perspective of interest and perspective of emotion (Sailer et al. 2013). Even though several definitions for game elements used in gamification exist(Weiser et al. 2015; Sailer et al. 2013; Hamari et al. 2014), they all have some common elements, such as assignments, quests and goals; points, credits and levels; achivements and/or badges; and leaderboards and collections.

RESEARCH METHOD

The aim of this research was to understand *what kind of gamified mechanisms are used in sustainability applications* and *how they relate to ways to motivate pro-environmental behavior*.

The research articles provide one view on the gamified ways used for creating pro-environmental behavior. The two strongest themes presented in the research articles were electricity and transport, and in total of 6 articles were selected to this research. The research-oriented applications are usually designed for the purpose of research, and thus they are utilized only by a small group of research participants and only for the period of the research. To gain a wider view about the design of sustainability applications also Google Play applications were considered. The 30 first applications with a search word "sustainability" was reduced to ten applications by taking into account only thost that had more than 1000 downloads. The search was conducted in March 2018 from Finland, from the researcher's computer and the google search algorithms have an effect on the applications the search found. Thus the applications can be considered as a quite random presentation of the existing applications.

RESULTS

The vast majority of applications, including all six research applications, aimed at encouraging changes in behavior through recording sustainable actions of the users or at least giving information about the possible actions. The rest of the applications focused on sharing information about sustainability in general or sustainability of a company. Two of the applications, ShareBuddy and the application by Brewer et al. (2015), included minigames and their motivational elements are listed in the table in parenthesis, but not taken into discussion and conclusions.

Sustainability Applications in Research Articles

The applications in research articles (in Table 1) inform users how to be more sustainable, which seems to translate into environmental-friendly actions taken by the participants (Jylhä et al. 2013; Froehlich et al. 2009; Dillahunt et al. 2008; Brewer et al. 2015) or can be seen in opinions of the participants (Nguyen 2014; Kjeldskov et al. 2012). Social activity is seen to be one of the key components to commit people to sustainable choices, and it is either empirically investigated (Nguyen 2014; Brewer et al. 2015) or theoretically discussed (Froehlich et al. 2009). Typically, these applications are related to changing personal behavior, such as household energy usage (Dillahunt 2008, Kjeldskov et al. 2012; Nguyen 2014; Brewer et al. 2013), or transport methods (Froehlich at al. 2009; Jylhä et al. 2013).

Citation	Motivational elements
Froehlich et al. 2009	Points, badges, feedback, progress. Commitment, rewards&penalties, feedback.
Jylhä et al. 2013	Point, badge, goals, challenge, feedback. Information, goal-setting, incentive, comparison, feedback.
Dillahunt 2008	Achievement, progress, story. Rewards&penalties, commitment, feedback.
Kjeldskov et al. 2012	Achievement, badge, friends, feedback. Information, commitment, comparison.
Nguyen 2014	Goals, achievements, leaderboards, friends, feedback. Information, goal-setting, commitment, comparison.
Brewer et al. 2015	Avatar, points, leaderboards, progress. Information, commitment, comparison, feedback.

Table 1. Motivational Elements from Gamification or Pro-environmental Psychology in Research Articles.

Dillahunt et al. (2018) designed an application with a virtual polar bear in "tamagotchi" style to find out whether virtual pets could have positive impact on environmentally responsible behavior. A story was used to develop attachment to the pet (for one of the groups in the research) and the amount of environmental actions was visualized through the increasing and decreasing size of the ice below the polar bear.

Power Advisor by Kjeldskov et al. (2012) provides information about power consumption in a household through an additive automatic meter reader. The application includes relating user's consumption to similar households, comparing to user's historical data, advice messages from experts as well as community messages. Used motivational elements included e.g. smiley faces as badges and visualization through comparative speedometer and charts.

Similarly to Power Advisor, the mobile application utilized by Nguyen (2014) is following energy consumption through a smart meter in a household. The application gives the user feedback about the hourly energy usage in form of charts and comparison to previous usage. To create commitment and enable comparison, a newsfeed provides information about the user's friends and their consumptions, goals and personal records. It is also possible to compare oneself with others through leaderboards. (Nguyen 2014)

ShareBuddy does not focus only on reducing electricity and water usage, but also encourages players to shift the electricity usage throughout the day. The players get feedback from their behaviour in form of resource points, which are used to advance in the game itself and to play minigames. Players have an avatar which can walk and take actions (e.g. showering, cooking). Players can also compare themselves to others in a leaderboard. The game design elements in minigames are similar to the main game, e.g. points. (Brewer et al. 2015)

UbiGreen consists of an application and a motion sensor to detect different kind of transportation automatically. UbiGreen provides feedback visually either in form of a tree growing leaves, or in the form of new polar bears and seals representing the achievements of the user as a background image in a mobile phone. The graphical illustrations are accompanied with badges, which inform the users their recent green activities. (Froehlich 2009)

Similarly to UbiGreen, MatkaHupi sums up the CO_2 emissions produced by the automatically detected transportation method (walking, cycling, bus/tram/metro, driving a car). The application includes information about more sustainable choices for the trip, a journey planner and a set of challenges. Visual feedback on the CO_2 emissions of the current and last three weeks are presented. As gamification elements, the users get points and badges from their trips and completion of challenges. (Jylhä et al. 2013)

Popular Sustainability Applications

The sustainability applications (in Table 2) found in Google Play can be divided into two categories. "*Habit changing*" applications encourage users to take small actions, and two of the five applications include saving or even informing other players about the actions taken. "*General information*" applications are only providing information about sustainability in different areas, and do not give advice on changing personal habits.

Application	Motivational elements
JouleBug	Goals, points, badges, leaderboards, friends, feedback, rewards. Information, goal-setting, rewards, commitment, comparison, feedback.
sustainability@BU	Goals, points, badges, leaderboards, friends, feedback, rewards. Information, goal-setting, rewards, commitment, comparison, feedback.
SDGs in Action	Assignments, friends, story. Information, goal-setting, commitment.
Carbon3R	Information.
Sustainable Seafood Guide	Information.
Sustainable Canned Tuna Guide	Information.
Sustainable Development Goals	Information.
Sustainable Development Goals (regional)	Information.
SDG Game & Quiz	Points, leaderboards, (avatar, virtual goods). Information, comparison, feedback.
ITC Sustainability	Story. Information.

Table 2. Motivational Elements from Gamification or Pro-environmental Psychology in Google Play Applications

Habit Changing Applications
JouleBug has developed two similar sustainability applications, sustainability@BU and JouleBug. They include several actions categorized into habits, energy, water, waste, transportation, food&drink, shopping, office, outdoors and advanced. The users can complete actions, track their progress and share their progress with their friends (including Facebook and Twitter). Thus, feedback is given not only by the application, but also by other users. Information about the possible ways to be more sustainable is presented through the description of a quest. Upon completion of a quest, points and badges are received as rewards. The users can compare themselves with friends through leaderboards.

Where JouleBug has readymade actions, SDGs in Action encourages people to create their own actions and invite others to join them. The creator of an action must link it against one of the United Nation's Sustainable Development Goals (SDG). Social sharing is an integral part: in addition to inviting other users to participate in an action, the applications includes also likes and shares. Despite all the gamified approaches, information sharing is the largest part of the application, including detailed information about the 17 SDG's and their targets as well as explanatory videos and latest sustainability development news. The videos can be considered as using stories for information sharing.

Carbon3R-Sustainable Lifestyle is an application which gives information about eco-friendly lifestyle through three categories: reduce, reuse and recycle. It has only textual information about the actions to be taken, excluding features for saving or sharing the actions. In addition, the application includes integrated news feed from Tree Hugger, Yahoo Green and New York Times Green Living. There are some logos used for different actions such as recycle or compost, and widely thought these could be considered as badges.

Sustainable Seafood Guide is an information sharing application for Australia and Sustainable Canned Tuna Guide is for Canada. The idea is similar: information about different tuna marks is provided in form of traffic lights. More information about the data behind the color, e.g. how and where the tuna was caught, is also available. The traffic lights can be considered as a gamified design element, closest to badges. At the time of testing the applications, Sustainable Canned Tuna Guide did not work at all, so the information is based on the figures in Google Play.

Informative Applications
Three out of four informative applications focus on United Nation's Sustainable Development Goals (SDG). Each one has a different aim and approach. Sustainable Development Goals is aimed for people who need to check the information about the SDG's regularly. It includes the goals, their targets and indicators in textual form. The aim seems to be a quick check-up place for the information. The second Sustainable Development Goals application is aimed for local and regional governments. There, the relevant targets for local and regional governments in each of SDGs are selected and their practical meanings explained textually.

SDG Game & Quiz has two distinct parts: a game and a quiz. Before or after the game, user gets textual information about SDGs and sustainability issues around the world. In the game itself, "good" and "bad" things drop down, and user catches the "good" things by touching them. These are actually logos representing different issues in SDGs. The other part of the application is a quiz about facts related to SDGs and by answering the quiz the user gains extra lives for the game. There are leaderboards for both the quiz and the game.

ITC Sustainability is an informative application of an Indian company called ITC. In addition to the company's sustainability goals, there are multiple topics on where the company is active in the community: e.g.renewable resources, forestry, water, e-commerce for farms, education, women empowerment, recycling, training, health and hotels. The form of sharing this information is short stories accompanied with photos, and possibly links to videos or other related material such as pdf files.

DISCUSSION AND CONCLUSIONS

Motivational game design elements and **ways to motivate pro-environmental behavior** have several overlapping themes. ***Feedback*** is found in similar role in both of these models. **Goal-setting** is presented in form of *quests, goals and challenges*. **Incentives and rewards** can be expressed e.g. through *points, badges and rewards*. **Commitment** can be expressed through *avatars* and *progress*; to *friends*, *teams* and *groups*; or even in form of **competition** through *leaderboards*. **Information** is an important way to motivate people in pro-environmental psychology, but has no direct counterpart in gamification. Perhaps a *story* or a *theme* can be utilized for this. In general, we can see, that game design elements are more practical and can be considered as ways to implement the motivational ways described in pro-environmental psychology.

Points, badges and leaderboards are typical elements in gamified applications (Hamari et al. 2014) as was found in our research also. In addition to these (basic) elements, social interaction with other people was also highlighted. One interesting factor is also the utilization of a story, which was used successfully to create commitment and as a way to share information about possible or actions taken. In addition to text based information, also charts, figures or even videos were used for information sharing. Information was the only element always present and for some applications the only element used, especially for popular applications. We can summarize our findings as follows:

- **Information** sharing can be found in all the applications and thus be considered as a key element.
- ***Feedback*** from user's current sustainability is an important way when the aim is to encourage behavior change.
- *Points, levels and badges* are typically used as **rewards** from the recorded sustainable actions.
- To create **commitment** and **comparison**, social networks are used for sharing the results with e.g. *friends* or through *leaderboards.* Also a *story* can be used to create commitment through emotions.

Our research higlights elements that are currently used in both scientific and popular applications giving guidance to application developers. The absence of only information sharing applications in scientific research brings up a question that is information sharing enough, or does it need to accompagnied by feedback and social elements.

ACKNOWLEDGEMENT

The research reported in this article has been conducted as a part of SUSTIS (Sustainability and Transparency in Shipbuilding Networks) project. The project is carried out in collaboration with VTT Technical Research Centre of Finland, DNV GL Business Assurance, Evac, Lautex, Meriteollisuus, Meyer Turku, NIT Naval Interior Team, Paattimaakarit, Piikkio Works, Sininen Polku and SSAB Europe. The project is mainly funded by Tekes – the Finnish Funding Agency for Technology and Innovation.

REFERENCES

Brewer R. S.; N. Verdezoto, T. Holst, and M. K. Rasmussen. 2015. "Tough Shift: Exploring the Complexities of Shifting Residential Electricity Use Through a Casual Mobile Game". In *Proceedings of the 2015 Annual Symposium on Computer-Human Interaction in Play (CHI PLAY '15)* (London, UK, Oct. 5-7). ACM, New York, NY, USA, 307-317.

Deterding S.; D. Dixon; R. Khaled and L. Nacke. 2011. "From Game Design Elements to Gamefulness: Defining "Gamification"". In *Proceedings of the 15th International Academic MindTrek Conference: Envisioning Future Media Environments* (Tampere, Finland, Sep. 28-30). ACM, New York, NY, USA, 9-15.

Dillahunt T.; G. Becker; J. Mankoff; R. Kraut. 2008. "Motivating environmentally sustainable behavior changes with a virtual polar bear". In *Workshop on Pervasive Persuasive Technology and Environmental Sustainability.*

Froehlich J.; T. Dillahunt; P. Klasnja; J. Mankoff; S. Consolvo; B. Harrison; and J. A. Landay. 2009. "UbiGreen: investigating a mobile tool for tracking and supporting green transportation habits". In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '09)* (Boston, MA, USA, Apr. 4-9). ACM, New York, NY, USA, 1043-1052.

Froehlich J.; L. Findlater; J. Landay. 2010. "The design of eco-feedback technology". In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '10)* (Atlanta, Georgia, USA, Apr. 10-15). ACM, New York, NY, USA, 1999-2008.

Hamari J.; J. Koivisto and H. Sarsa. 2014. "Does Gamification Work? A Literature Review of Empirical Studies on Gamification". In *Proceedings of 2014 47th Hawaii International Conference on System Sciences* (Waikoloa, HI, USA, Jan 6-9). IEEE, 3025-3034.

Jylhä A.;P. Nurmi; M. Sirén; S. Hemminki: and G. Jacucci. 2013. "MatkaHupi: a persuasive mobile application for sustainable mobility". *In Proceedings of the 2013 ACM conference on Pervasive and ubiquitous computing adjunct publication (UbiComp '13 Adjunct)*. ACM, New York, NY, USA, 227-230.

Kjeldskov J.;M. B. Skov, J. Paay, and R. Pathmanathan. 2012. "Using mobile phones to support sustainability: a field study of residential electricity consumption". *In Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '12)*. ACM, New York, NY, USA, 2347-2356.

Kollmuss, A. and J. Agyeman. 2002. "Mind the Gap: why do people act environmentally and what are the barriers to pro-environmental behaviour?" *Environmental Education Research* Vol. 8, No. 3, 239–260.

McKenzie, S. 2004. "Social Sustainability: Towards some Definitions". *Hawke Research Institute, University of South Australia*.

Nguyen S. P. 2014. "Mobile application for household energy consumption feedback using smart meters: Increasing energy awareness, encouraging energy savings and avoiding energy peaks," *2014 International Conference on Collaboration Technologies and Systems (CTS)*. Minneapolis, MN, 2014, pp. 291-296.

Sailer, M.; J. Hanse; H. Mandl; and M. Klevers. 2013. "Psychological perspectives on motivation through gamification." Interact. Des. Archit., 19, 18–37.

Sawitri D. R.;Hadiyanto H.; and Hadi S.P. 2015. "Pro-environmental Behavior from a SocialCognitive Theory Perspective", *Procedia Environmental Sciences*. 23, 27-33.

Weiser P.; D. Bucher; F. Cellina and V. De Luca. 2015. "A Taxonomy of Motivational Affordances for Meaningful Gamified and Persuasive Technologies". *In Atlantis Press*. Atlantis Press, 271-280.

GAMIFYING LEARNING OF MARITIME STANDARD OPERATIONAL PROCEDURES

Olli I. Heimo[1,2], Tapani N. Joelsson[2], Sari Nyroos[3] & Tuomas Mäkilä[2]

[1]Department of Management and Entrepreneurship, Turku School of Economics
[2]Department of Future Technologies
[3]Centre for Maritime Studies, Brahea Centre
[1,2,3]20014 University of Turku, Finland

KEYWORDS

Gamification, Octalysis, Maritime, Simulation

ABSTRACT

In this paper we analyze the design decisions of a maritime chemical incident OP and SOP training game through the Octalysis gamification framework. The goal is to understand how well the learning game manages to implement the gamified features and thus increase its engagement. We argue that even though the game is still under development and the framework is not a silver bullet in analyzing the gamified content, the design decisions made for the game seem valid and thus it motivates people to train saving lives in these dangerous incidents.

INTRODUCTION

In this paper we introduce the on-going development of a maritime chemical incident search and rescue gamified learning platform GameSAR and analyze the gamification process through Octalysis framework. The learning platform is part of EU-funded ChemSAR -project which aims to *"create uniform operational plans and standard operational procedures (SOP) to save human lives in maritime HNS accident areas in a manner that the lives of the rescue crews will be protected and the impact on the environment will be minimized."*(ChemSAR, 2018.)

Maritime sector transports great amounts of hazardous and noxious substances. If vessel carrying these substances has an accident, the resulting search and rescue operation requires its own set of procedures to counter the adverse effects caused by the cargo. Traditional training methods in crisis management include classroom learning for theoretical background, live training in physical simulations and variety of software-based methods ranging from serious games to virtual reality environments. (E.g. Heldal & Wijkmark, 2017)

All of these means are used to relay experience based information and teach skills and procedures that will minimize harm to the rescue personnel and other stakeholders on the harm's way. As noted earlier, there are highly immersive virtual reality environments for these purposes. One of these is the RS Maritime System by VSTEP B.V. For other examples see e.g. Göbel (2016).

Environments like this are highly immersive and can be used in all levels of the training, including the training of the on the ground personnel and their hands-on procedures. Downside of these environments is their relatively high price and requirements for equipment and facilities.

On the other end of digitized teaching methods we have the serious games and gamified learning materials. Serious games and gamified materials are less immersive than virtual reality or 3D simulators, but they can be used as platform to promote case-based reasoning on situations in which subjects are exposed to a new situation with limited information and they have to start to solve the situation based on their knowledge.

In this paper we aim to test the game through the Octalysis framework. We aim to evaluate our design decisions through and validate our decisions via the framework as well as bring a new case for the framework. We describe the background and reasoning for the case, a new learning game for a search and rescue protocol, and present the methods and analysis for gamifying the learning simulation.

CHEMSAR PROJECT AND ITS GOALS

As the ChemSAR background report (Yliskylä-Peuralahti, 2017) attests, there is a lack of operational plans (OPs) and standard operational procedures (SOPs) for search and rescue (SAR) operations at sea, applicable to cases of hazardous and noxious substances (HNS) incidents in the Baltic Sea Region (BSR). The Interreg BSR project ChemSAR - Operational Plans and Procedures for Maritime Search and Rescue in HNS Incidents - combats this problem by creating operational plans and standard operational procedures for use in SAR operations where HNS are involved. To support the implementation of the SOPs and OPs, the project creates an e-learning platform with learning materials, together with a chemical databank and a gamified training tool. The learning environment will raise awareness and emphasize the complexity and limitations of an HNS-related SAR operation at sea.

ChemSAR learning materials are supplementary to already existing SAR material, e.g. national guidelines, SAR education and training, as well as the International Aeronautical and Maritime Search and Rescue Manual of the International Maritime Organization. The target groups include rescue, environmental and maritime authorities, SAR training and maritime academies, and shipping companies and seafarers.

The complexity of the issue provides an interesting and challenging framework for learning material development and gamified training tool. ChemSAR concept highlights the need for joint procedures, coordination and cooperation between different actors within and beyond national boundaries. Due to the different kinds (differing physical and chemical properties), quantities and combinations of chemicals transported, the variety of hazards they may pose is extremely large. Within this versatile and challenging operating environment of highly skilled professionals, ChemSAR e-learning platform strives to offer a pleasant and effective learning experience.

GAMIFICATION AND SIMULATION EXERCISES

Simulation exercises for ChemSAR e-learning platform are aimed for the operational leads, thus they are conveying best practices for handling various situations during incidents involving cargo with NHS on board. In these cases the important factors are usage of time and suitability of the decisions for the ongoing situation. For this we decided that instead of full blown virtual environment much simpler decision making simulation based on the real-time clock and situational awareness was deemed to be most suitable solution.

As we didn’t want the 'gameness' to be something that is glued on top of simulation, we looked for existing design guides and frameworks. As we are already familiar with gameplay design patterns we started from them. These patterns contain “commonly recurring parts of the design of a game that concern gameplay”. (Björk & Holopainen, 2004) As our simulation tool is not an game for entertainment, but more like a game-like or gamified solution, the gameplay design patterns were not really helpful for us expect as a general design tool. There exists a small set of gamification patterns by Ašeriškis & Damaševičius (2014) which we found to be more helpful on the design decisions. These patterns do not contain a way to validate if the produced system really is gamified, so for this we had to find another way. As we have not yet done extensive user testing, we had to be able to do it on by expert review. To find a tool for this review, we looked for the relevant literature and found review by Mora et al. (2015) for existing gamification design frameworks. We familiarized ourselves with these and looked for other works that had utilized the frameworks presented on the review. With this process we selected the Octalysis framework (Chou, 2015a; 2015b) and accompanying Octalysis evaluation tool (Chou, 2015c) to help us in our evaluation. Previous uses for this tool contained the work by Yfantis and Tseles (2017) in which they used Octalysis to evaluate civic platform Challenge.gov and Economou et al. (2016) who evaluated the serious games platform for learning.

Octalysis is a gamification framework which has set of eight "Core Drives", scored using a scale from 0 to 10. These are related to different aspects on eliciting motivating and engaging experience, as summarized in the following Table 1 (based on Chou, 2015b).

The analysis is based on the first level of Octalysis framework. On the deeper levels we could be analyzing things like how we are onboarding the player or how we handle the end game. But, our game is about learning in predetermined time frame so we don't have to worry about these kinds of aspect.

GAMESAR – DESIGN GOALS AND GAME DESCRIPTION

The main goal of the GameSAR is to support the other learning material produced by the project and thus support the goals of the project in educating the operational plans and standard operational procedures in HNS incidents. Thus the game is also an educational tool amongst a larger variety of tools and must be understood as such.

The objective of the GameSAR is to learn SOPs and OPs by repetitive but yet engaging manner so that the player a) can learn how to act on a maritime chemical rescue situation where a lot of information is primarily unavailable and thus improve their skills as a rescue center officer and b) is aware and understands how the rescue center works - even though they may not work there.

There are two main design goals for the game. First and foremost the game must be easy to adopt and use thus negating the possibility of digital divide. The user-base must be understood to be varied by age, gender, and gaming history and therefore only the most common game elements as well as those most easily understood can be used in the UX design.

Table 1: Core Drivers of Octalysis (Chou, 2015b)

Epic meaning and calling	Person believes they are doing something greater than oneself or was "chosen" to take the action
Development and accomplishment	Internal drive for making progress, developing skills, achieving mastery, and eventually overcoming challenges
Empowerment of creativity and feedback	Users are engaged in a creative process where they repeatedly figure new things out and try different combinations
Ownership and possession	Users are motivated because they feel like they own or control something
Social influence and relatedness	Incorporates all the social elements that motivate people, including: mentorship, social acceptance, companionship, and even competition and envy
Scarcity and impatience	Wanting something simply because it is extremely rare, exclusive, or immediately unattainable
Unpredictability and curiosity	Constantly being engaged because you don't know what is going to happen next
Loss and avoidance	Motivates us to avoid something negative from happening

Secondly the purpose of the game gives limitations. The game must – in its' part – develop and verify the skills and know-how for the officials in case of chemical maritime incidents. The scenario scripting as well as the virtual tools represented in the game must be accurate (or at least accurate enough) to ensure the use proper procedures during the actual incidents. Therefore the game must – to a certain point – represent the actual world and its' procedures and the use of fiction to enable more elaborate experience is limited.

In practice, the game was designed to be played from the perspective of emergency response center as the actions and decisions made in it seemed the most various and most important to the process as well as because it is a task that everyone involved to the process benefits to understand. Also the educational factors were best included in this task. As the game is single-player the player must do the tasks of several persons in the response center team and thus the possibility for simultaneous events was ruled out.

The virtual tools in the game design are similar to the tools used by emergency response center: a telephone, radio, interactive chart, whiteboard, and chemical database. Whereas the real world operations take hours or even days to solve, the educational computer game should be completed in less time. Also, the game should be more intensive and therefore the time flow was simulated and the operations e.g. making a phone call or using radio advances the game time.

The game starts by introducing the current situation (i.e. mayday-call) and puts the player to the position of the emergency officer. The player can use the aforementioned tools to both gather more information about the situation and to make decisions with the aim to save the vessel and the passengers in the emergency situation. The actions cost time, which is limited, (e.g. using telephone advances the time by n minutes) as well as some actions give new options to advance. More options can also open as the game time advances, as certain time-points trigger events which update or change the information. All the decisions of the player are stored and at the end of the scenario the game gives feedback according to the decisions and the outcome from them at the end.

The game can include numerous different scenarios with different storylines, chemicals, and events written by the project group to both give variance and to further enlighten the players about the multitude of different situations encountered by the emergency response center officials.

Technically GameSAR is a browser-based application and it is created using a open source HTML5 game framework called Phaser (Photon Storm, 2018). Browser was selected as a platform as our audience has a diverse set of devices in their use. With these tool selection decisions we can support multiple platforms and devices with different screen sizes and input methods without creating separate packages for every platform.

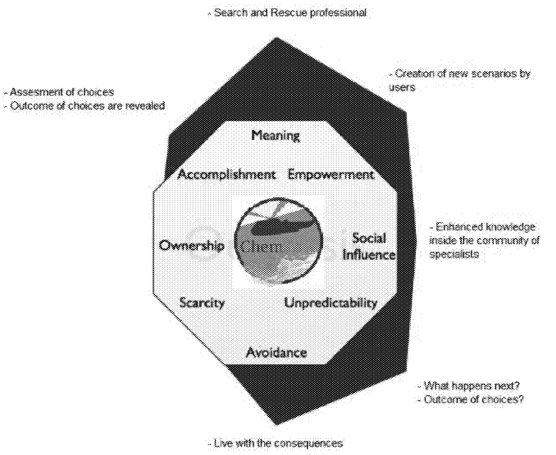
OCTALYSIS ANALYSIS

Our game is aimed to support the learning of maritime chemical rescue operations. Thus learning is the most important aspect of it and everything else should support it. To evaluate the game and its aspects intended to be motivational and engaging, we use the Octalysis model to to find out the weaknesses and the strengths of the game.

The following Picture 1 shows the summary visualization of the analysis with the Octalysis tool.

In GameSAR you are training how to correctly act on a disaster event which can save lives in real-life situations. The core drive of *Epic meaning* is therefore fundamental to our tool. This aspect is valued to 8 as the potential consequences are presented to the player constantly.

On the *Development and accomplishment* the score is 3. Earlier version of the game included scoring mechanism, but after testing with the target audience it was removed as it was seen as distracting feature by them. To them the important thing was to see outcome of the choices made, and what were the correct ones.



Picture 1: ChemSAR analysis with Octalysis tool (tool provided by Chou, 2018).

For the *Empowerment of creativity and feedback* our score is two-fold. Creativity is not encouraged by the game during the gameplay, so to this aspect the score is 1. But the main outlet for creativity is the possibility to create own scenarios for others to play with simple scripting language. Hence the final score is 6.

Ownership and possession is about owning or controlling things. In our game you are trying to take the situation under your control, but you are not really owning or gaining things with your actions so we evaluate this aspect to 0 points. This is something we could improve, maybe by making the player/learner to be more possessive on the resources they have.

Social influence and relatedness are social elements, which the game does not support directly. The game is also meant to be played in classrooms and that way social interaction between the players is encouraged. But as there are no in-game mechanics supporting this core drive, it is valued at 2.

On the drives for *Scarcity and impatience* the score is 0 points. As we are aiming for teaching real-life procedures for search and rescue operations, we do not put limitations to how long and how much the game can be played. Also, all scenarios are playable immediately.

For the core drive closest to engagement with the content, *Unpredictability and curiosity*, valuation is 6 points. Game presents players with unexpected situations to which they have to react based on their own skills. To make most out of these situations, they have to be paying attention to details and correct procedures to follow. This also relates directly to the *Loss and avoidance*. If players are not paying attention the in-game situation escalates quickly and leads to an unrecoverable situation, just like in the real world. Therefore this core drive is also evaluated to 6 points.

CONCLUSIONS

Octalysis is developed as a tool for analyzing the design of gamified services and how they appeal to different motivational factors. The main purpose of our learning game is to be an educational tool, and to support this purpose it uses design practices from games for its interface and gamification as a design guide for other aspects. But, as the game is only a part of the predetermined curriculum for learning SOPs, it is not intended to be used for prolonged periods of time. Thus the game differs from most of the services that are usually using gamification techniques to extrinsically to motivate people to use them.

According to the Octalysis framework the game can be classified as a gamified solution and therefore its design is successful. The analysis also shows which parts we might need to think about improving our design in case the user experience testing shows that the application is not engaging enough. Similarly on the lines with Economou et al. (2016) we note that Octalysis (or other gamification frameworks) should be improved or modified to fit cases where main task for the gamified service is learning, since some of the core drives are a bit difficult to apply in this domain.

Maritime HNS SAR questions are complex and, since the research being just on-going, largely unanswered thus the game setting is limited to the knowledge currently at hand. As the ChemSAR project around the game advances, more content can be implemented to the game. Due to the modular structure of the game mechanics the extra content however can easily be implemented and thus extending the life-span of the game. It is obvious that the research is still in its early phases. The team's future research will focus on developing the game according to the principles found in this research as well as continue the development by finding more information through conducting empirical research.

The game aims to be pedagogic yet entertaining and thus it is only partially realistic simulator on what the work in the response center, moreover a never-tiring teacher on different situations may one encounter and how one should react to the situations at hand. Whereas the game is still in early alpha stage there are indications that this game will indicate the possibilities of gamifying serious and even critical aspects of our society - our safety and thus it can be a recommended option in teaching those that will eventually be responsible for our lives.

REFERENCES

Ašeriškis, D., & Damaševičius, R. (2014). Gamification Patterns for Gamification Applications. Procedia Computer Science, 39, 83-90.

Björk, S., & Holopainen, J. (2004). Patterns in game design (game development series). 1. ed., Charles River Media, December 2004.

ChemSAR (2018), ChemSAR – Operational Plans and Procedures for Maritime Search and Rescue in Hazardous and Noxious Substances (HNS) Incidents, ChemSAR project web-pages. Available at https://www.utu.fi/fi/yksikot/mkk/hankkeet/chemsar/Sivut/home.aspx, accessed 21.2.2018.

Chou, Y.K. (2015a). Actionable Gamification: Beyond Points, Badges, and Leaderboards. Octalysis Media.

Chou, Y.K. (2015b). Octalysis: Complete gamification framework. Available at the: https://goo.gl/zgdmHv Accessed 12.2.2018

Chou, Y.K. (2015c). Octalysis Tool. Available at the: http://www.yukaichou.com/octalysis-tool/ Accessed 12.2.2018

Economou, D., Doumanis, I., Pedersen, F.S., Kathrani, P., Mentzelopoulos, M., Bouki, V. and Georgalas, N. (2016). Westminster Serious Games Platform (wmin-SGP) a tool for real-time authoring of roleplay simulations for learning. EAI Endorsed Transactions on Future Intelligent Educational Environments, Vol. 16, Issue: 6. e5. ISSN 2409-0034.

Göbel S. (2016) Serious Games Application Examples. In: Dörner R., Göbel S., Effelsberg W., Wiemeyer J. (eds) Serious Games. Springer, Cham.

Heldal, I. & Wijkmark, C.H. (2017). Simulations and Serious Games for Firefighter Training: Users' Perspective. In eds Aurélie Montarnal Matthieu Lauras Chihab Hanachi F. B. Tina Comes (Ed.), Proceedings of the 14th International Conference on Information Systems for Crisis Response And Management (pp. 868–878). Albi, France.

Mora, A., Riera, D., Gonzalez, C., & Arnedo-Moreno, J. (2015, September). A literature review of gamification design frameworks. In *Games and virtual worlds for serious applications (VS-Games), 2015 7th international conference on* (pp. 1-8). IEEE.

Photon Storm Ltd. (2018). Phaser Desktop and Mobile HTML5 game framework. Corporate website: http://phaser.io/ Accessed: 22.2.2018

VSTEP B.V. (2018). Corporate website https://www.vstepsimulation.com/ Accessed 12.2.2018.

Yfantis, V., & Tseles, D. (2017). Exploring Gamification In The Public Sector Through The Octalysis Conceptual Model. Era-12 International Scientific Conference, October 2017.

Yliskylä-Peuralahti, J. (2017). Preparedness to Maritime Chemical Accidents in the Baltic Sea Region. Publications of the ChemSAR project 3:2017.

Designing Gamification for Constructive Competition

Mark Featherstone
Sheffield Hallam University
Sheffield
Email: m.featherstone@shu.ac.uk

Keywords
Gamification, mobile, education, constructive competition, video games

Abstract
This paper examines the need for constructive positive extrinsic motivational competition within gamification projects. Gamification takes common game design principles such as points, leaderboards and competition, then applies them to non-gaming activities. Participants often require extrinsic motivators to engage with gamification, such as financial reward, compulsory participation or prizes. This approach can reduce intrinsic motivation, creativity and sense of agency. One powerful extrinsic motivator is competition, which can be effective even without any real world prize. Competition can be divided into constructive and destructive types. Destructive competition can cause anxiety and lower self-esteem in participants. Constructive competition is motivating without these negative side-effects. It isn't possible to guarantee that a competition will be constructive, but there are broad principles that can be applied to design for constructive competition. These principles were investigated using a purpose built mobile application called UniCraft. This app was used in a cross-over study with university students in an attempt to increase their satisfaction with one of their subjects and it featured a 3D video game-like competitive battle mode. Online analytics recorded a statistically significant increase in app usage when this competitive game mode was enabled.

Introduction
To gamify an activity is to take something that is not game-like and then wrap game design principles around that activity (Deterding et al., 2011a), for example increasing your heart rate beyond a previous exercise session earns points, displayed on a leaderboard (Whitson, 2013). When game design principles are applied to an activity, people have a tendency to find that activity more compulsive, which some perceive as being more fun (Hopson, 2001). Points, leaderboards and achievements tend to make it easier for 'players' to judge their progress and aptitude for a task both alone and in comparison to others. Gamification can have the following positive impacts on any task (Deterding, 2015):

- The task becomes more enjoyable due to the new sense of playfulness.
- The task is performed correctly.
- The 'player' increases their productivity.
- The task is performed to a higher quality.

However, gamification often reduces intrinsic motivation, a desirable state where a participant is engaged fully with a task, often in a state of flow (Chen, 2007), it's a condition of optimal learning potential and creativity. The participant is engaged with the task for its own sake with no outside coercion.

Reduced intrinsic motivation can manifest negatively within the individual in a number of ways (Fuchs et al., 2014; Raczkowski, 2013):

- A loss of agency.
- Reduction in creativity.
- A loss of self-worth.
- A loss of interest or engagement with the activity.
- Feelings of oppression or that the system is overly prescriptive.

To mitigate against this requires an understanding of root causes.

Gamification requires progress within any activity to be measured so points can be awarded or removed. Measuring an activity means defining it in detail, which can reduce the creative freedom of participants. Such measurements are often made public via a leaderboard to encourage participants to compete and compare their progress.
Competition is an extrinsic motivator, people are generally competitive and it can provide an extra impetus to progress. This can cause anxiety, demotivation and stress (Hanus and Fox, 2015; Lepper and Malone, 1987; Shafer, 2012).
Rewards provide further extrinsic motivation when they are of value to participants. Rewards can be linked into compulsory participation, for example, a prize for students achieving a certain grade. Participants can focus on the reward instead of the activity, becoming disillusioned if they don't get the reward or unhappy with the value of the reward (Deci and Ryan, 2000).
It's not possible to predict with accuracy how human participants will respond to gamification schemes, just as it isn't possible to guarantee the success of a video game design (Koster, 2013). However, like video game design methodology, there should be a 'best practice' approach to the design of gamification (Deterding, 2015). This paper analyses the design of competition in gamification. The term, 'constructive competition' refers to competitions designed to avoid negative side-effects which might reduce intrinsic motivation.

Design

What follows is a set of 'best practice' guidelines that can be applied when designing for constructive competition.

Non-prescriptive measurement

Any complex activity can be distilled into measurable sub-tasks, with points awarded for completion. Sometimes a sub-task has a very specific methodology, especially if there are health and safety implications. Often, sub-tasks can be more general or fluid in their definition of methodology and outcome. This supports the participant's desire for independence and agency (Deci and Ryan, 2000).

Team based play

When participants compare their progress, scores can represent the individual or the group. When participants feel they are acting together as part of a group, the impact of success or failure is shared. Persevering together and even failing together can foster a feeling of comradeship and mutual support that nurtures friendships.

Cohort based play

To compare progress and compete doesn't mean pitting one group against another. In video game design this is known as PvP and can be very stressful. Another approach, known as PvE, allows an entire cohort of participants to work together against a virtual opponent, such as a fantasy monstrous enemy (Adams, 2013). There is the potential within competition for participants to become antagonistic towards each other. If the participants see the 'opposition' as a virtual enemy then feelings of antagonism towards that opponent can be expressed safely and healthily.

Multiple measures of progress

When sub-tasks within an activity have to be completed in sequence, there is the potential for a participant that is struggling with the task to feel there is no path forwards or no obvious way to increase their scores as they fall to the bottom of the leaderboard. In video games this issue is addressed by including multiple measures of success with multiple paths to achieve them. This approach enhances participant agency allowing them to delay or bypass or navigate around challenging tasks, while remaining competitive.

Fun - the power of video games

Gamification is based on techniques within game design and it can be presented using video game imagery, phrases and concepts, even when used with a non-gaming related activity. This can help people recognise the competition as fun and playful as well as encouraging participation. Tools like Unity3D and

Unreal allow developers to deliver gamification projects that more closely resemble popular modern 3D video game aesthetics on small budgets (Axon, 2016).

Asynchronous play

Maintaining a sense of agency in participants can include allowing them to decide when, how and where they take part. One way to enable flexible participation is using personal mobile devices to interact with the gamification system. In terms of games design, asynchronous multiplayer competition allows players to participate in a shared world together, but without having to be present concurrently (Zagal et al., 2000).

Virtual rewards

Gamification's extrinsic motivators (points, leaderboards, competition, etc.) require an extra driver which is often some kind of reward (Whitson, 2013). As previously discussed, valuable rewards can create negative associations, for example, becoming overly reliant on financial reward. Video games often use virtual rewards, without real-world importance. Usually these are associated with a player avatar, for example, clothing, pets, housing, vehicles, etc. Virtual rewards can form part of an economy, for example, a stallion or sports car that is expensive and rare within the virtual economy of the game. Players transfer value onto virtual items, yet they don't have any real-world importance.

Avatars

People care about how they are perceived by their peers. Within a competition, where progress is displayed on a public leaderboard, this can be motivating, however there are risks, as previously discussed. Avatars are anonymous virtual representations of participants and work optimally when the user can customise the avatar to better represent their idealised image (usually using virtual reward items). People care about their virtual avatars (Behm-Morawitz, 2013), but it provides a degree of separation between them and the potential tension and embarrassment of being identified via competition.

Elective participation

When any activity becomes compulsory, participants lose agency and independence. However, if a competition is not compulsory then participants may drop out at any point. Within video games participation in multi-player competition is a well know problem. This can be addressed by allowing people to take part asynchronously at a convenient time. The competition event can be split into multiple shorter competition events creating multiple smaller prizes. This allows participants to take part in a more ad-hoc fashion, maintaining their independence.

Player matching

People respond positively to a well-played game, even if they lose (DeKoven, 2002). Video games often use algorithms to match players of similar ability or rank for competition, increasing the likelihood of a well-played game (Jennings, 2014).

Holistic approach

The effect of each of these design axioms is amplified when they are combined. For example, without compulsory participation, why engage with a competition? By using video game themes, avatars, virtual rewards, etc. the competition begins to regain the motivational levers necessary to maintain engagement with a lower probability of reducing intrinsic motivation.

UniCraft battles

The author has investigated these ideas within a gamification project with second year computing higher education students (Featherstone and Habgood, 2018). UniCraft is a mobile gamification platform with cloud hosted database and built in analytics to record the time and type of every interaction with the application, see Figure 1.

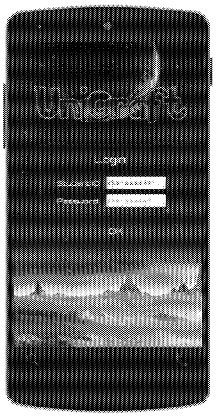


Figure 1. UniCraft mobile app

Students in the second year of their course were separated into two tutorial groups by surname. These two groups were offered the chance to participate in a cross-over study and became groups A and B totalling 26 students, see Table 2. The organisation of the study is shown in Table 1.

	Weeks in semester											
	1	2	3	4	5	6	7	8	9	10	11	12
interview						A						B
normal lessons	B	B	B	B	B	B	A	A	A	A	A	A
using the app	A	A	A	A	A	A	B	B	B	B	B	B
battle game is available	×	×	×	✓	✓	✓	✓	✓	✓	×	×	×

Table 1. Organisation and schedule of study

Students earn credits for attendance, asking questions, completing tutorials, handing in work, etc. Credits buy virtual items to customise their virtual avatars, see Figure 2. Participants compete within a fantasy battle competition. Outcomes of battles are randomised, but those with more expensive virtual items are more likely to survive longer, therefore encouraging students to earn as many achievements as possible.

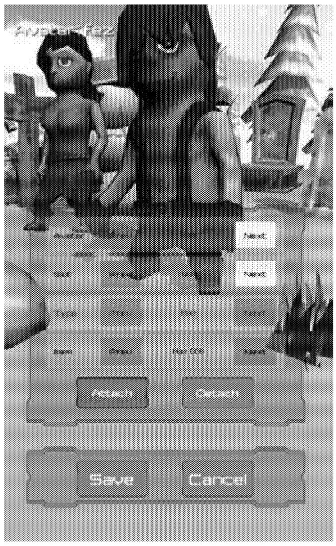


Figure 2. UniCraft virtual avatars

These competitive battles can themselves be used to earn more credits, proportional to how long the player survives. They can be played non-interactively, while the student is working or interactively with the player gaining a small advantage by 'catching' hearts from fallen enemies. It is based on the popular one-click game design mechanic seen in many mobile games (Unger and Novak, 2012).

Avatars can compete alone or in small teams (see Figure 3) against a computer controlled enemy (PvE).

Figure 3. Three student avatars team up for a battle

When in non-interactive mode, a competition event can be displayed on a projector, with the avatars of the entire cohort taking part in a 'battle royale'. This example of constructive competition showed an increase in engagement with the gamification app of 217% compared to using the system without the competitive battles, see Table 2.

Student group	App events – battle inactive	App events – battle active	Increase in Unicraft usage	ANOVA
A	300	1215	305%	F(1,18)=16.79 P=0.0007 η2=0.93
B	383	1176	207%	F(1,20)=3.3 P=0.08 η2=0.17

Table 2. Impact of constructive competition on app engagement

Within the student group it was noted that people interacted with the system on different days of the week, at different times of the day, using the app to differing degrees, interacting with some aspects more than others, both in and outside class. This technologically enabled and designed-for flexibility helped maximise engagement.

During interviews, the students reported that they had enjoyed the competitive battle game and did not find it stressful. They claimed this was because it was seen as a light-hearted fun activity allowing them to compete with their peers without pressure and it helped motivate them to engage with the gamification project.

After the study a comparison of student attainment was made to see if there had been any impact. Student assessment results were compared to the previous cohorts over three years. A 17% increase in attainment was measured, compared to the three previous years (single factor ANOVA $F(3,162)=3.45$, $P=0.018$, $\eta^2=0.06$), see Figure 4.

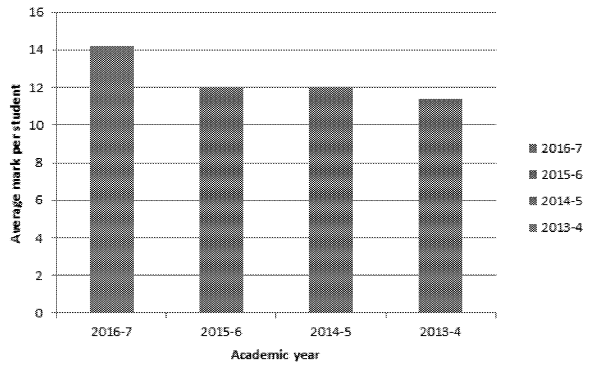


Figure 4. Student attainment, UniCraft was used in 2016-7

Conclusion

Gamification has repeatedly demonstrated its efficacy when applied to a range of activities (Deterding et al., 2011b; Laird, 2017; Rigsby, 2012) and competition plays a key part in engaging participants. However, it isn't possible to accurately predict how people will react to such systems. The likelihood of competition having a positive and constructive impact can be increased if a theory of best practice can be developed, promoting a holistic design approach. Constructive competition is one example of a powerful extrinsic motivator that is compatible with maintaining intrinsic motivation, which is vital in supporting an individual's sense of self-determination.

Gamification works, but participants must be motivated to stay engaged with the gamification process. Constructive competition can provide that motivation while limiting the chance of any negative impact that competition might have on the intrinsic enjoyment or satisfaction in the task being gamified.

References

Adams, E., 2013. Fundamentals of game design. New Riders.

Axon, S., 2016. Unity at 10: For better—or worse—game development has never been easier | Ars Technica UK [WWW Document]. Arstechnica. URL https://arstechnica.co.uk/gaming/2016/09/unity-at-10-easy-game-development/ (accessed 9.2.17).

Behm-Morawitz, E., 2013. Mirrored selves: The influence of self-presence in a virtual world on health, appearance, and well-being. Comput. Human Behav. 29, 119–128.

Chen, J., 2007. Flow in games (and everything else). Commun. ACM 50, 31. doi:10.1145/1232743.1232769
Deci, E.L., Ryan, R.M., 2000. The “What” and “Why” of Goal Pursuits: Human Needs and the Self-Determination of Behavior. Psychol. Inq. 11, 227–268.
DeKoven, B., 2002. The well-played game: a playful path to wholeness. iUniverse.
Deterding, S., 2015. The lens of intrinsic skill atoms: A method for gameful design. Human–Computer Interact.
Deterding, S., Dixon, D., Khaled, R., Nacke, L., 2011a. From game design elements to gamefulness: defining gamification, in: Proceedings of the 15th International Academic MindTrek Conference: Envisioning Future Media Environments. ACM, pp. 9–15.
Deterding, S., Sicart, M., Nacke, L., O’Hara, K., Dixon, D., 2011b. Gamification. using game-design elements in non-gaming contexts, in: CHI’11 Extended Abstracts on Human Factors in Computing Systems. ACM, pp. 2425–2428.
Featherstone, M., Habgood, J., 2018. Unicraft: exploring the impact of asynchronous multiplayer game elements in gamification. Int. J. Human-Computer Stud. (in Press.
Fuchs, M., Fizek, S., Ruffino, P., Schrape, N., 2014. Rethinking gamification. Meson Press.
Hanus, M.D., Fox, J., 2015. Assessing the effects of gamification in the classroom: A longitudinal study on intrinsic motivation, social comparison, satisfaction, effort, and academic performance. Comput. Educ. 80, 152–161. doi:10.1016/j.compedu.2014.08.019
Hopson, J., 2001. Behavioral Game Design [WWW Document]. Gamasutra. URL https://www.gamasutra.com/view/feature/131494/behavioral_game_design.php (accessed 9.8.17).
Jennings, I., 2014. Matchmaking Algorithm: Skill-based Matchmaking | PubNub [WWW Document]. PubNub. URL https://www.pubnub.com/blog/2014-07-31-skill-based-matchmaking-multiplayer-games-tutorial/ (accessed 8.16.17).
Koster, R., 2013. Theory of fun for game design. O’Reilly Media, Inc.
Laird, S., 2017. Top 10 Enterprise Gamification Cases That Will Make Employees More Productive [WWW Document]. yukaichou.com. URL http://yukaichou.com/gamification-examples/top-10-enterprise-gamification-cases-employees-productive/ (accessed 9.2.17).
Lepper, M., Malone, T., 1987. Intrinsic motivation and instructional effectiveness in computer-based education. Aptitude, Learn. Instr.
Raczkowski, F., 2013. It’s all fun and games... A history of ideas concerning gamification. DiGRA Conf.
Rigsby, J., 2012. Yammer, Badgeville Join to Bring Gamification to Enterprise Social [WWW Document]. CMSWire. URL http://www.cmswire.com/cms/social-business/yammer-badgeville-join-to-bring-gamification-to-enterprise-social-015629.php (accessed 9.2.17).
Shafer, D.M., 2012. Causes of State Hostility and Enjoyment in Player Versus Player and Player Versus Environment Video Games. J. Commun. 62, 719–737. doi:10.1111/j.1460-2466.2012.01654.x
Unger, K., Novak, J., 2012. Mobile game development. Delmar/Cengage Learning.
Whitson, J.R., 2013. Gaming the quantified self. Surveill. Soc. 11, 163–176.
Zagal, J.P., Nussbaum, M., Rosas, R., 2000. A Model to Support the Design of Multiplayer Games. Presence Teleoperators Virtual Environ. 9, 448–462. doi:10.1162/105474600566943

Author Biography

Mark Featherstone is a PhD student and lecturer at Sheffield Hallam University within its Applied Gaming Technology research group.

PhD Supervisor

Dr. Jacob Habgood is the principal investigator of the Horizon 2020 REVEAL project and manages Sheffield Hallam University's PlayStation teaching facility and Steel Minions Game Studio.

Developing a Gamified Platform to Involve Unemployed Youth in Job-Seeking Activities

Juho Mattila, Eeva Leinonen, Ilkka Hietaniemi, Aryan Firouzian, Petri Pulli
University of Oulu
Faculty of Information Technology and Electrical Engineering (ITEE)
Oulu Advanced Research on Service and Information Systems (OASIS)
email: juho.mattila@ikunelabs.com, ilkka.hietaniemi@student.oulu.fi
email: {eeva.leinonen, aryan.firouzian, petri.pulli}@oulu.fi

KEYWORDS
Gamification, Serious Game, Unemployment, Job-seeking

Abstract

Although it is commonly known that young people are active online users, the challenge is, how to attract them to use job-seeking services. Using game design elements in non-game context to create immersive experiences can have significant impacts on the usage of motivation. Our interest was to examine how to find solutions that make the elements of gamification and human skills complement each other in the harmonic and possible way. The project's objective was to construct a platform using game technologies that bridge isolated young people and work services together for the purpose of helping young people in their job-seeking activities. After extensive research and prototyping, a gamified design for the main application was selected, and serious gaming ideology was chosen as the learning method for a range of training games that the main application would unify together as a platform. We are arguing that gamified job-hunting services have the potential to support and activate the unemployed young in job search processes by promoting employment and corresponding to the needs of young adults.

INTRODUCTION

Unemployment is extremely expensive for the societies and public sectors, and it can cause adverse effects on mental health, such as depression and bipolar disorder (Murphy and Athanasou 1999). The circumstances and experiences of young people in society have changed considerably today compared to the previous generation. These changes affect experiences in the lifestyle, education, labor market, and ability to become an independent member of society. Young people have multiple choices of pathways to choose from and therefore can cause fear of risk of failure. According to a report on the well-being of young people, more than half of unemployed feel anxious about everyday life situations and try to avoid meeting new people. In their population-based study, Axelsson and Ejlertsson (2002) argue that social support among young unemployed can avoid becoming socially isolated and slipping away from the jobs market. (Axelsson and Ejlertsson 2002)

The growing demand for qualification is expected to cause lack of skilled workers for the market, and it brings up life-long, and self-learning approaches as a central paradigm in education and training, which gives ICT solutions have a vital role in this process. Young unemployed are a potential target group to motivate through a gamified application. There is a total of 241,600 job seekers in Finland, and 52,000 of them are at the age of 25-34 (Tilastokeskus 2018). According to Eurostat (2008) 50% of EU unemployed people aged 20-34 years are ready to change their home place for a job, 21% are prepared to move inside the same country for a job, while 12% would consider moving inside of EU area, and 17% are even ready to step outside the EU region. It raises the question if the gamified applications can be used to promote the job-seeking process for the young unemployed.

Serious gaming has been prominent in computer-assisted learning in recent years in fields such as education, healthcare, engineering and science, and its applications continue to spread steadily to new areas of learning.

To establish the definition for a game, Michael et al. (2006) define it as "voluntary activity separate from the real world, creating an imaginary world that may or may not have any relation to the real world and that absorbs the players full attention. Games are played out within a specific time and place, are played according to established rules, and create social groups out of their players."

For serious games, a commonly used definition is by Marsh (2011): "Serious games are digital games, simulations, virtual environments and mixed reality/media that provide opportunities to engage in activities through responsive narrative/story, gameplay or encounters to inform, influence, for well-being, and/or experience to convey meaning." Girard et al. (2013) added to the definition that Serious games are virtual games with a useful purpose because otherwise all virtual games could be considered to teach something even though it's not the actual purpose. Michael et al. (2006) defines serious games as follows: "A serious game is a game in which education (in its various forms) is the primary goal, rather than entertainment."

The development project depicted in this paper, later designated as HireMe, was designed to bridge isolated young people and the work centrum together to help young people in their job-seeking activities. After extensive research and prototyping, a gamified design for the main application was chosen. Serious gaming design principle was selected as the learning method for a range of training games.

To follow a systematic development of simulation-based serious games and applications, we studied the literature and attempted to find solutions, which suits our case model. Greenblat and Duke (1981) introduced a game development framework through four iterative stages of initiation, design, construction, and use. Initiation step indicates game specification, system description, and game components. Design stage focus on the game mechanism, and element design. The construction stage includes development, pilot and usability tests. The final stage is real operation through field testing. (Greenblat and Duke 1981)

Considering those stages as a general development framework, Robinson (2008) states that initiation and design stages should be integrated to iterative sets of conceptual modeling, modeling coding, experimentation, and implementation. He claimed that this approach leads to a proper understanding of real-world problem and solutions. (Robinson 2008)

Van der Zee et al. (2010) defined simulation model used by emphasizing the aspects of game elements on the conceptual modeling activities. They claimed the players and game leaders should be considered as stakeholders, and case examples should be implemented in the design to fulfill the requirements for training and educational purposes. Following various practices of educational training, a serious game solution should not only target soft-skills such as language, negotiation, networking, and leadership; but also more practical skills must be improved. (Van der Zee et al. 2010)

Garris et al. (2002) introduced an input-process-output game-based requirements acquisition process model. The input phase is a pre-gaming phase, in which the developers integrate instructional content into the game content. The process phase is the stage to collect and assess user reaction and behavior and system feedback. The output phase is the stage to evaluate the game achievements, which can be the learning outcome of specific skills in a serious game scenario. (Garris et al. 2002)

OHagan et al. (2014) conducted a systematic literature study of software process in game development. The study finding indicates that agile development cycles such as Scrum and Kanban are more relevant to knowledge intensive domain, where the iterative evaluation points are needed through user reactions and system feedback. (OHagan et al. 2014)

Asuncion et al. (2011) followed an iterative agile process with Scrum to develop a series of serious games related to a campus tour in the university. Their analysis unveiled that the selected development process would strongly support the involvement of users in the development stages, which supports the main principal of User-Centered Design. (Asuncion et al. 2011)

IMPLEMENTATION

The initial concept was formed as a gamified application which provides training and services in the job-seeking process for unemployed youth. It encourages the users to follow self-learning approach and also impose more extensive virtual presence in business and employment-oriented services, and online expert community like LinkedIn and Stack Exchange networks. We believe that the virtual presence in those networks leads to a comprehensive understanding of job market requirements.

Platform

The platform is designed to improve a collection of skill-sets of the users, in the job hunting process. Those skills are categorized into two groups. Soft skills are a set of general practices such as language literacy, negotiation, networking, team-work, and leadership. On the other hand, practical skills are defined by the nature of the job.

We chose to use virtual rewards and levels as game elements in our application based on Zuckerman and Gal-Oz (2014) research, studying the effectiveness of different game elements. Competition element was as effective as virtual rewards, so we decided to use virtual rewards. The motivation behind the virtual rewards and level-based progression is that companies hiring employees could see the users achievements. The better the achievements are, the better chance

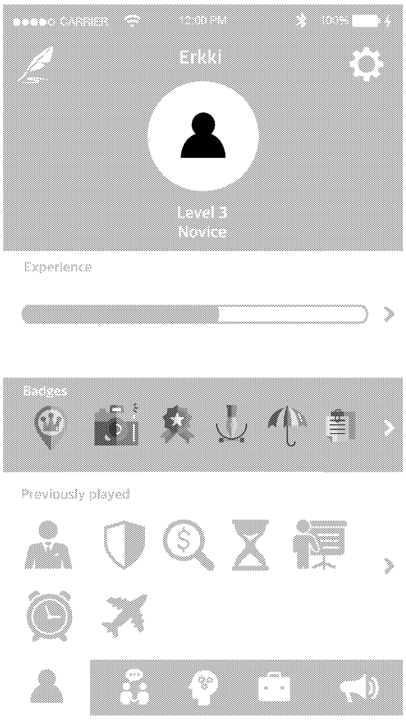


Figure 1: The first prototype was developed as an iOS application.

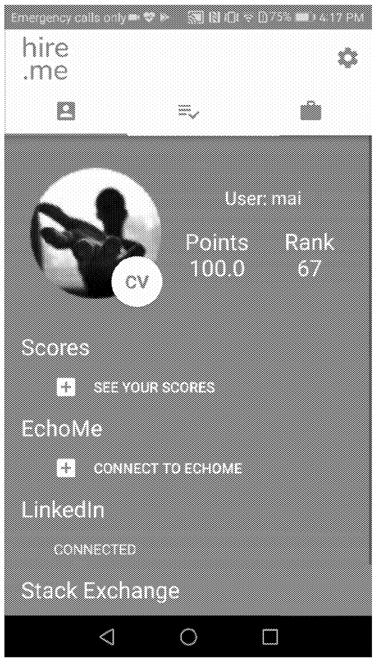


Figure 2: The second prototype was developed as an Android application.

the user has getting recruited. (Kim and Werbach 2016)

There are many online educational communities for training different practical skills. They usually emerge as a high-quality network of questions and answers on topics in varied fields. Those are places where users can prove their competence in specific fields, such as Stack Exchange, which includes dedicated communities in various fields such as programming, mathematics, different branches of engineering and so on. HireMe application is developed as an integrated platform to fetch users activity points from Stack Exchange networks, through available third-party API.

HireMe platform was developed in three periods. The initial concept was designed at the beginning of fall 2016, and then three development iterations were conducted. In the first iteration, the application was developed in fall 2016 by a student group in the research and development project course, in University of Oulu. Later in summer 2017, the second development iteration was conducted by another student group in the summer school of Kajaani University of Applied Sciences, Finland. Finally, the last development iteration was conducted as part of the research and development project course in fall 2017 in University of Oulu. Three different student groups were involved in those development iterations.

Scrum framework was used during development iterations. Project members met weekly through scrum sprints to hold intense interaction and discussion. They act as the product owner, scrum master and other stakeholders in different scenarios related to the HireMe platform, and mini-games.

The first HireMe prototype was developed for iOS platform using React Native framework. Since it was the first iteration, the purpose was to have an easy-to-use API to enable integration of third-party training games into the application. In the beginning, the back-end was a Node.js server with MongoDB non-relational database, and it was running on Heroku cloud application platform. It was designed to be highly flexible and lightweight enabling various types of trials when testing the HireMe prototype. (Figure 1)

For the second and third prototype, Android platform was chosen for the application, which could be tested by a more extensive variety of users. It was developed by Android Studio to design the front-end, and Docker engine with Vert.x framework and MongoDB to develop the back-end.

Training Games

There are two approaches to integrate educational and training information in HireMe environment. Third-party developer can design games for the platform, or available APIs from other third-party applications can be used to monitor the users activities in other training networks. Figure 3 shows the current schema

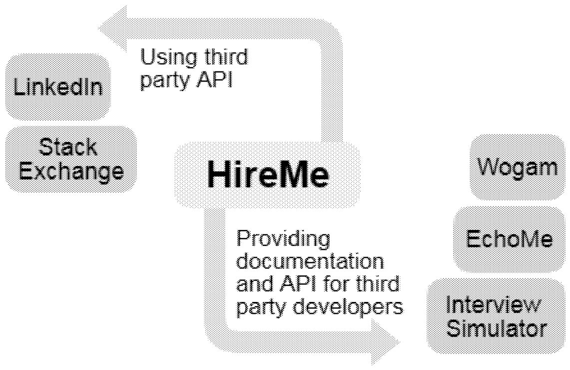


Figure 3: Training games can be developed as native mini-games of HireMe platform, or third-party APIs of available training applications can be used to monitor activities and achievements of users.

of available and implemented mini training games and use of third-party APIs in HireMe platform.

Parallel to HireMe development, three mini-games were integrated into the platform. An interview simulator was developed in Unity environment to simulate a job-seeker and recruiter interaction. A word puzzle game (Wogam) was designed to improve the language literacy in French and English languages. Furthermore, EchoMe was developed as a web application to improve the presentation skills of the users through peer support processes. (Van der Zee et al. 2010)

Interview simulator
As the first example to demonstrate the process that HireMe platform connects to training games through the main application, we developed a simulation type of interview game by Unity game engine. In the game, the player is sitting in a chair in a first-person view, and the interviewer is asking job interview related questions. Player has four different choices for each question and a limited time to answer. Every choice has a different point value that the player receives. The player also receives feedback after the answers. We reduce errors and improve the players independent learning by providing feedback to them. Feedback also creates the feeling of accomplishment and motivates the player (Kwon and Lee 2016). The game aims to prepare the player for a real-life job interview. In fact, a job interview cannot be trained easily in real life context. Ypsilanti et al. (2014) stated that the success of learning using serious games lies in the actual involvement of a participant playing the game, which in turn, creates increased cognitive links with real-life situations. (Figure 4)

EchoMe
EchoMe, Figure 5, is an application to improve the user's presentation skills. It is developed as an online

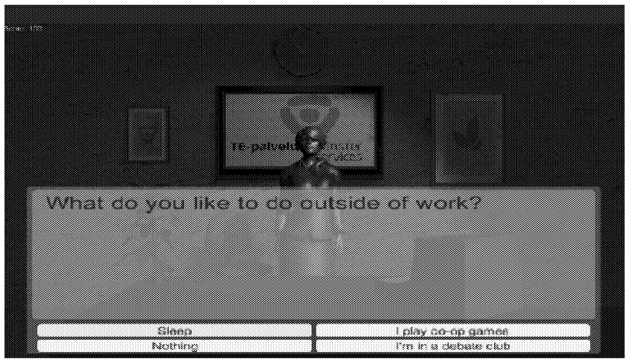


Figure 4: Interview Simulator

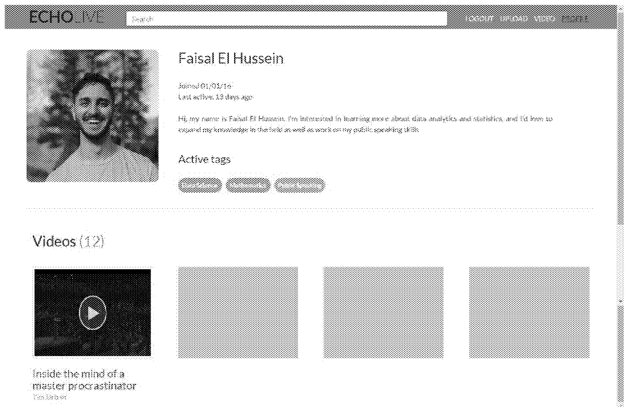


Figure 5: EchoMe is an online solution to improve the presentation skill through peer coaching.

solution using peer coaching to improve the users' skills, and engaging them through gamification elements while integrating into the HireMe platform. Its implementation provides features to create and categorize video presentations, and a possibility to start a discussion on a video track, which will appear on a particular timestamp when playing the video file. EchoMe's front-end user interface was developed by CSS, HTML, and JavaScript, while AngularJS was used to develop the real-time commenting system. For the database implementation, Laravel and PHP framework were used.

Wogam
Wogam is another mini-game, and it was developed by Unity game engine as a student project. It helps users to improve French and English vocabulary and language literacy. Figure 6 shows that based on the displayed card image, users have to create the correct word by placing letter cards in the right order and discover new vocabularies. Spaced repetition system and simple pointing system are implemented to encourage the user to be engaged in this gamified application.

LinkedIn & Stack Exchange integration
The HireMe platform is currently integrated with LinkedIn and Stack Exchange network through avail-

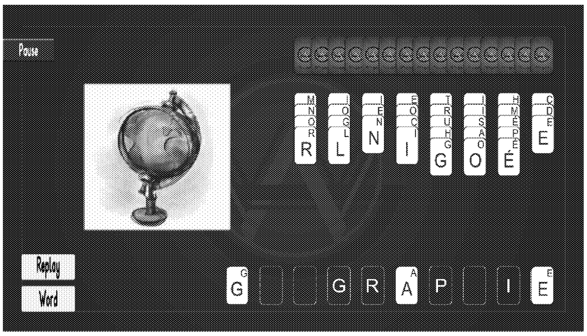


Figure 6: Wogam is a puzzle card game to assess users vocabulary knowledge in French and English language and help them to improve it.

able REST APIs. LinkedIn API fetches users' essential information to display it in HireMe profile. That information includes experiences, publications, languages, skills, certificates, education and recommendation fields. It provides the recruiter with comprehensive background information on the job applicants. Stack Exchange API fetches users information regarding their activities in different Stack Exchange networks. It includes achievements such as points, privileges, and badges, and most frequent tags associated with the user. In short, HireMe provides an integrated environment for the job seekers and recruiters, where job seekers can reflect their points and achievements from a different expert and educational networks, and the recruiters can monitor practical activities and skills of applicants in a unified format.

EVALUATION

For the acquisition of user requirements and the job-hunting process, we had a brainstorming session on 24th of October 2017 with experts of young people labor administration at TE-office, Finnish unemployment office. During the session, we presented HireMe and EchoMe concepts to the experts, and they provided us with the knowledge of characteristics, value and job search processes of young unemployed that were needed to be considered in further development tasks.

The initial prototypes were evaluated through a heuristic evaluation as well as scenario-based evaluation methods by the development team. Specific sprint sessions were dedicated to the evaluation task.

In the later development iteration, while mini-games and third-party applications were integrated into the platform, we conducted a two-stage evaluation. The participants were recruited in by the help of unemployment office in the city of Oulu. The first phase was in a recruiting event in Oulu, Finland (MegaMatchmaking 2017). It aims to improve the test process and provide proper benchmarks for the measurements in the next phase. The second phase of the evaluation was conducted in the facilities of the University of Oulu.

The first test was planned to be conducted as an observation test by audio and screen recording. Five test subjects participated in the test, and they were asked to fill in the questionnaire after the test. The age of the participants ranged from 24 to 37 with an average age of 31. Users were supposed to create an account in HireMe, add dummy resume and job interest, and then connect the account to mini-games and third-party applications. Participants were asked to use a dummy or real third-party accounts such as LinkedIn or StackOverflow. The test revealed some technical usability flaws in design such as fragment displacement, menu navigation problem, and unhandled errors if the connection was lost. The first test results provided us with a benchmark for the second usability test.

After minor modification in the platform, the second test was conducted with setting similar to the first test, and the same tasks were assigned to the subjects. The subjects were supposed to accomplish their task in estimated benchmark time from the first test. Seven test subjects participated in the second test with the age range of 25-37 and an average age of 29. Subjects were asked to answer a questionnaire after the test. Results of the second test revealed few technical usability flaws, such as dealing with unresponsive API of a third-party or lack of feedback from the UI. The questionnaires were designed to collect information about the attitudes of the users, and they indicated a positive satisfaction rate towards the HireMe concept, despite usability flaws in the technical design.

The main aim to conduct the evaluation phase is to assess the effectiveness, efficiency, and satisfaction of HireMe among target users by assigning predefined tasks. The tasks focused on the functionality of the application, and they start by creating a profile and then connecting to third-party services and mini-games, and finally, all achievement and points should be presented in HireMe profile in a unified format.

CONCLUSION

There is little evidence of research carried out into gamified services for the unemployed young. It is known that unemployment is expensive for societies and tackling it is important but challenging. Particularly focusing on attracting the young unemployed to use job-seeking services is essential from the point of view of society and the unemployed young. ICT based services with gamification elements seem to have the capacity to motivate searching information on employment, developing new skills and extending social networks in order to add user experience and value. The findings of this study also support such an approach.

The HireMe platform that uses game technologies for bridging unemployed youth and work service providers together were presented in this paper. After conducting the literature review, brainstorming sessions and prototyping a gamified design for the main application, serious gaming ideology was chosen as the learning method for a range of training games that the main application would unify together as a platform. Usability testing with experts and young people has been carried out as a way to asses the impact of HireMe platform on services for the unemployed youth.

Findings show that for the purpose of helping young people in their job-seeking activities HireMe has the potential to support and motivate unemployed youth by promoting employment and corresponding the needs of developing their skills. Despite not having the ideal sample size of ten subjects in the second test, the results still give confidence in the usability test that was conducted. Based on the second test, HireMe falls behind in satisfaction compared to other Android applications and has some critical flaws that prevent the usage of the application, hence it is not usable as it is. While the data still might be accurate, it is not as reliable as a completely recorded test session.

As part of the future plan, we aim to implement an API document to specify necessary detailed information to develop third-party mini-games, which can be integrated to the platform.

ACKNOWLEDGEMENT

The research presented in this paper was supported by Serious Games platform for Business and Education (SEGABU) project. We thank Mika Haapaniemi, Jarkko Tuovinen, Anni Ahonen, Joni Pennala, Aatos Lang, Toni Närhi and Vinoth Selvaraj for their commitment and contribution in research and development course.

References

Asuncion H.; Socha D.; Sung K.; Berfield S.; and Gregory W., 2011. *Serious game development as an iterative user-centered agile software project.* In *Proceedings of the 1st International workshop on games and software engineering.* ACM, 44–47.

Axelsson L. and Ejlertsson G., 2002. *Self-reported health, self-esteem and social support among young unemployed people: a population-based study. International Journal of Social Welfare*, 11, no. 2, 111–119.

Garris R.; Ahlers R.; and Driskell J.E., 2002. *Games, motivation, and learning: A research and practice model. Simulation & gaming*, 33, no. 4, 441–467.

Girard C.; Ecalle J.; and Magnan A., 2013. *Serious games as new educational tools: how effective are they? A meta-analysis of recent studies. Journal of Computer Assisted Learning*, 29, no. 3, 207–219.

Greenblat C.S. and Duke R.D., 1981. *Principles and practices of gaming-simulation.* SAGE Publications, Incorporated.

Kim T.W. and Werbach K., 2016. *More Than Just a Game: Ethical Issues in Gamification. Ethics and Inf Technol*, 18, no. 2, 157–173. ISSN 1388-1957. doi:10.1007/s10676-016-9401-5. URL http://dx.doi.org/10.1007/s10676-016-9401-5.

Kwon J. and Lee Y., 2016. *Serious Games for the Job Training of Persons with Developmental Disabilities. Comput Educ*, 95, no. C, 328–339. ISSN 0360-1315. doi:10.1016/j.compedu.2016.02.001. URL http://dx.doi.org/10.1016/j.compedu.2016.02.001.

Marsh T., 2011. *Serious games continuum: Between games for purpose and experiential environments for purpose. Entertainment Computing*, 2, no. 2, 61 – 68. ISSN 1875-9521. doi:https://doi.org/10.1016/j.entcom.2010.12.004. URL http://www.sciencedirect.com/science/article/pii/S1875952110000224. Serious Games Development and Applications.

Michael D.D.R.; Chen S.; and ebrary I., 2006. *Serious games : games that educate, train and inform.* Boston, Mass. : Thomson Course Technology. ISBN 1592006221. URL http://books.google.com/books?isbn=1592006221.

Murphy G.C. and Athanasou J.A., 1999. *The effect of unemployment on mental health. Journal of Occupational and organizational Psychology*, 72, no. 1, 83–99.

OHagan A.O.; Coleman G.; and OConnor R.V., 2014. *Software development processes for games: a systematic literature review.* In *European Conference on Software Process Improvement.* Springer, 182–193.

Robinson S., 2008. *Conceptual modelling for simulation Part I: definition and requirements. Journal of the operational research society*, 59, no. 3, 278–290.

Van der Zee D.J.; Holkenborg J.; Gunal M.; Tjahjono B.; Robinson S.; and Taylor S., 2010. *Exploring simulation-based serious gaming–Framing the issues.* In *Proceedings of the Operational Research Society Simulation Workshop.* 108–117.

Ypsilanti A.; Vivas A.B.; Räisänen T.; Viitala M.; Ijäs T.; and Ropes D., 2014. *Are Serious Video Games Something More Than a Game? A Review on the Effectiveness of Serious Games to Facilitate Intergenerational Learning. Education and Information Technologies*, 19, no. 3, 515–529. ISSN 1360-2357. doi:10.1007/s10639-014-9325-9. URL `http://dx.doi.org/10.1007/s10639-014-9325-9`.

Zuckerman O. and Gal-Oz A., 2014. *Deconstructing gamification: evaluating the effectiveness of continuous measurement, virtual rewards, and social comparison for promoting physical activity. Personal and Ubiquitous Computing*, 18, 1705–1719.

AUTHOR LISTING

AUTHOR LISTING