POLITECNICO DI MILANO
Corso di Laurea Magistrale in Ingegneria Informatica
Dipartimento di Elettronica e Informazione

CASTLE: a video game for lateral thinking training

Relatore: Prof. Pier Luca Lanzi
Correlatore: Prof.ssa. Manuela Eliane Anna Cantoia

Tesi di Laurea di:
Davide Jones, matricola 771517
Roberto Casanovi, matricola 771465

Anno Accademico 2011-2012

Sempre, a Martina

Ringraziamenti

Vorremmo ringraziare il professor Pier Luca Lanzi per l'aiuto e la disponibilità manifestati durante la stesura di questa tesi. Vorremmo anche ringraziare la professoressa Manuela Cantoia e Alessandro per l'aiuto e l'assistenza forniti in questi mesi. Rivolgiamo inoltre un ringraziamento speciale a Fava e Lucio per tutto il meraviglioso tempo passato assieme in questi anni. Ringraziamo Efrem per tutto l'aiuto e i consigli forniti durante lo sviluppo del gioco.

Contents

List of Figures

List of Tables

Abstract

Video games have emerged as the most frequently used interactive media among children and as an important tool for brain training and education [1]. In collaboration with the Università Cattolica del Sacro Cuore, we developed a video game for lateral thinking training. We had 72 students playing the game on different platforms. Through data collection and analysis, we investigated how subjects with different backgrounds react to the problems proposed in the game and whether the platform affects player's cognitive performance.

Our game appears to be an useful tool to study various aspects and correlations of cognitive skills regarding lateral thinking. Findings indicated that the digital versions of the game stimulates lateral thinking skills of subjects more than non-digital version. Findings also suggested that gamers perform better in lateral thinking problems.

Our video game is currently used at the Università Cattolica del Sacro Cuore to investigate the use of lateral thinking in mid and high-school students.

Sommario

I videogiochi sono diventati uno tra i media più frequentemente utilizzati dai bambini e un importante strumento per l' apprendimento e per lo sviluppo delle capacità cognitive [1]. In collaborazione con l'Università Cattolica del Sacro Cuore, abbiamo sviluppato un videogioco per il miglioramento del pensiero laterale. Il videogioco è stato giocato da 72 studenti su diverse piattaforme. Attraverso la raccolta e l'analisi dei dati, abbiamo indagato come soggetti provenienti da contesti differenti reagiscono ai problemi proposti nel videogioco e se la piattaforma usata influenza le prestazioni cognitive dei soggetti.
I risultati suggeriscono che il nostro videogioco è uno strumento utile per studiare vari aspetti e correlazioni tra le capacità cognitive legate al pensiero laterale. I risultati suggeriscono che le versioni digitali dei giochi stimolano maggiormente le capacità di pensiero laterale nei soggetti rispetto a versioni non digitali. Inoltre vi è evidenza che i giocatori abituali ottengono risultati migliori nei problemi di pensiero laterale.
Il nostro videogioco è attualmente utilizzato presso l'Università Cattolica del Sacro Cuore per ricerca nell'ambito del pensiero laterale nelle scuole medie e superiori.

Chapter 1

Introduction

Educators increasingly recognize the impact of entertainment software and utilize games as a teaching device in a growing number of classrooms and business settings. In doing so, they are embracing the cultural and technological shifts of the 21st century and expanding the use of a favorite leisure activity, computer and video games, into a critical and still-emerging educational resource. More than just play, entertainment software helps imparting knowledge, developing life skills and reinforcing positive habits in students of all ages.

In addition to being a great way to keep students engaged, researchers have found that video games have real potential as next-generation learning tools. Games use new technologies to incorporate principles that are crucial to human cognitive learning.

University of Wisconsin education professor Dr. James Paul Gee concluded that video games intermix instruction and demonstration, a more effective learning technique than the style currently found in most classrooms [2]. According to a recent study conducted by researchers at the University of Michigan [3], puzzle games that exercise children's working memories can enhance their abstract reasoning and problem-solving skills, which can have a direct impact on future educational and occupational success. In addition, a study conducted by scientists at the University of Rochester found that video games can improve players' vision, attention and certain cognitive skills [4]. Study participants also performed better than non-gamers on certain tests of speed, accuracy and multitasking.

1.1 Lateral thinking

Among cognitive skills, lateral thinking is the ability to solve problems through an indirect and creative approach, thus lateral thinking distances itself from "vertical" logic (the classic method for problem solving: working out the solution step-by-step from the given data) or "horizontal" imagination (having many ideas but being unconcerned with the detailed implementation of them). Traditional thinking is to do with analysis, judgment and argument. In a stable world this is sufficient because it is enough to identify standard situations and to apply standard solutions. This may not apply in a changing world where the standard solutions may not work [5].

The term 'lateral thinking' was coined in 1967 by Edward de Bono who stated that lateral thinking can be taught and learnt through exercise. The term 'lateral thinking' is now part of our society and its tools are used by individuals and organisations around the world [5]: lateral thinking has been adopted in many fields, from business [6] to information technology [7].

1.2 CASTLE

CASTLE is a 2D multi-platform, dungeon crawling video game we developed in collaboration with the Università Cattolica del Sacro Cuore di Milano. CASTLE is a video game created to support research on lateral thinking in schools. To the best of our knowledge, there is no video game, publicly available, specifically developed and designed to study lateral thinking.
CASTLE was developed on the base of the expertise of the research unit of the Università Cattolica del Sacro Cuore. The application has been validated having 72 students playing the game on different platforms (personal computer, tablet and a paper version of the game). We used data mining techniques to analyse players' behaviour with respect to cognitive skills and lateral thinking. In our research, we investigated whether a subject who frequently plays video games performs better at lateral thinking problems and how the game platform (e.g. personal computer, tablet, etc) affects the cognitive performance of the subjects.

1.3 Data mining

Data mining is the process of analyzing data from different perspectives and summarizing it into useful information. It is widely used in computer science to analyse the behaviour of the information systems' users. Data mining is also often used to analyse players' behaviour in video games. For example data mining was used to analyse huge amounts of data to predict players' strategies in Starcraft [8]. In Tomb Raider Underworld, data mining was used to predict when a player would stop playing the game [9]. Thurau e Bauckhage used data mining to retrieve relevant information about social interactions in Massive Multiplayer Online Role-Playing Games [10].

1.4 Summary

This paper is structured as follows:
In the next chapter we give an overview of lateral thinking and video games.
In Chapter 3 we describe CASTLE.
In Chapter 4 we describe the tools and implementations used in the develop of CASTLE.
In Chapter 5 and 6 we describe how we managed to collect and analyse the data about player behaviors and comment the results.

Chapter 2

Lateral thinking

Lateral thinking is defined as the solving of problems through an indirect and creative approach, using reasoning that is not immediately obvious and involving ideas that may be not be obtained by traditional logic. The term was coined in 1967 by Edward De Bono [11], who also argued that lateral thinking can be taught and improved. Most of the games we are accustomed to are based on traditional logic, for example, Sudoku and Nim require nothing but mathematical logic [12][13].
In this chapter, we present examples of both classical and video games involving lateral thinking.

2.1 Practicing lateral thinking

Lateral thinking puzzles [14] are multiplayer games where a host describes an odd situation and the other players are asked to come up with an explaination for what happened. Players are stimulated to go overcome preconceptions to solve the puzzle. A famous example of lateral thinking puzzle is "A man rode into town on Monday. He stayed for three nights and then left on Monday. How come?". The described situation is impossible only if we think that Monday can only be the name of a day of the week. However the solution is that Monday is the name of the man's horse.

In the nine-dots puzzle a three by three matrix of dots is given and the player has to connect all the dots using four straight lines without lifting the pen. The solution requires the player to draw the lines outside the boundaries of the square area defined by the nine dots (Figure 2.1).

(a) (b)

Figure 2.1: The nine dots problem (a) and its solution (b)

The nine-dots puzzle popularized the phrase "thinking outside the box". The nine-dots puzzle first appears in Sam Loyd's 1914 Cyclopedia of Puzzles [15]. Sam Loyd's original formulation of the puzzle entitled it as "Christopher Columbus's egg puzzle".

Video games often involve aspects related to lateral thinking and have become among the most popular leisure time activity for children and young adults. The computer game industry is a billion-dollar business and its products have become a major part of today's media [16]. Several researches show that video games can improve various skills, such as motor, spatial and cognitive. Games like Electronic Arts's *Medal of Honor* or Activision's *Call of Duty* can improve visual acuity [17]; *Tetris* stimulate at a cognitive level, engaging our visual system in low-level pattern recognition [18]. The advantage video games have with respect to traditional games is that they constantly give to the player the prospect of being gratified for problem solving. Video games follow the principle of "low cost of failure and high reward success" [2] so that failure became as much motivating as rewards because every failure intrinsically teaches something useful to achieve success.

There are several examples of video games involving aspects related to lateral thinking. Some games like horror puzzle *Anna*[1] or the *Monkey Island* series[2] require the player to interact with the environment finding correlations between items to solve puzzles and advance; sometimes solutions to the puzzles involve non-traditional logic.

Other games give the player the possibility to solve puzzles in different ways using a creative approach. Examples of these games are *Crayon Physics Deluxe* and *Portal*. In *Crayon Physics Deluxe*[3] the player has to guide a ball from a start point collecting every

[1] www.dreampainterssoft.com/wp
[2] www.lucasarts.com/games/monkeyisland
[3] www.crayonphysics.com

star placed on the level. The player cannot control the ball directly, but rather must influence the ball's movement by drawing physical objects on the screen. *Portal*, developed by Valve Corporation in 2007[4], is the combination of a first-person shooter with a puzzle game. The game comprises a series of puzzles that must be solved by teleporting the player's character and simple objects using the "Aperture Science Handheld Portal Device", a device that can create inter-spatial portals between two flat planes. In 2011 Valve Corporation published a sequel[5] .

Similar to *Crayon Physics Deluxe*, *Fantastic Contraptions*[6] requires the player to build a contraption and to use it to move an object to a specific location. The game allows almost infinite ways to solve each level.

There are no video games aimed at studying lateral thinking. A research on brain fitness Nintendo's video game *Brain Age* showed that the effects of the brain training game were transferred to executive functions and to processing speed but there was no transfer effect on any global cognitive status nor attention [19].

On the other hand, a research supported by the National Science Foundation and Human and Social Dynamics program conducted on a sample of 490 parents and their 490 middle school children, suggests that children who play video games more are more creative than children who play video game less, while parent behavior on the dimensions of warmth and control is unrelated to their

[4] www.valvesoftware.com/games/portal
[5] www.valvesoftware.com/games/porta2
[6] www.fantasticcontraption.com

child's creativity [20].
In a study conducted by the Department of Psychology of the California State University, riddles and video games were used to develop lateral thinking. Preliminary results from their analysis provide promising evidence that these methods are effective in enhancing creative and critical thought in college students [21].
Another study shows that video game generated emotion significantly affects creativity through the interaction of arousal and valence [22].
CASTLE is a video game created to support research on lateral thinking in schools. CASTLE was developed on the base of the expertise of the research unit of the Università Cattolica del Sacro Cuore. The application has been validated having 72 students playing the game on different platforms (personal computer, tablet and a paper version of the game). We used data mining techniques to analyse players' behaviour with respect to cognitive skills and lateral thinking. In our research, we investigated whether a subject who frequently plays video games performs better at lateral thinking problems and how the game platform (e.g. personal computer, tablet, etc) affects the cognitive performance of the subjects.

2.2 Summary

In this chapter, we introduced the term 'lateral thinking' and and how games and video games are related to it.

In the next chapter we describe our video game CASTLE.

Chapter 3

CASTLE

Creative Activities Strengthening Thoughtful Learning Experiences (CASTLE) is a 2D video game we developed for the training of lateral thinking capabilities. The player, in the role of a student, is called to free a realm from an evil wizard who cast a curse that blocked inhabitants' creativity. The player has to explore the realm's castles solving puzzles to re-obtain the creativity needed to defeat the evil wizard. In the journey in CASTLE world the player encounters puzzles developed to train lateral thinking in different ways.

3.1 Game structure

When the game is launched, it shows a menu from which it is possible to start a new game, load a saved game or open the option screen (Figure 3.1).

Figure 3.1: Menu

At the beginning of a new game the player has to choose a name and the sex for the character. Then, the world-map opens up and the player can go to one of the four castles available (Figure 3.2).

Figure 3.2: World map

Three castles have twenty-five rooms, the fourth has five rooms.

Each room is viewed with a top-down perspective. When the player clicks a position on the screen, the character will move to it. Rooms are connected through doors. Some doors are closed, preventing the access to a room unless the player finds the key to open it by solving a puzzle. A room can contain various items such as tables, paintings, torches, and non-playing characters such as knights and ladies (Figure 3.3).

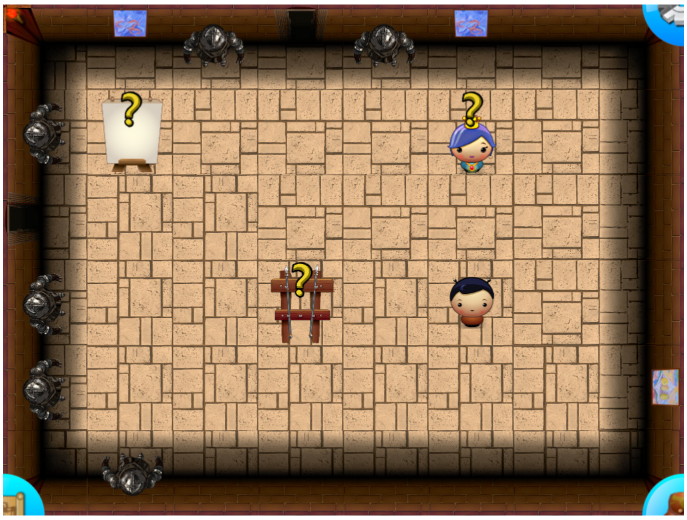

Figure 3.3: A room

Some rooms have no source of light and the player can only see in a small area surrounding the character (Figure 3.4).

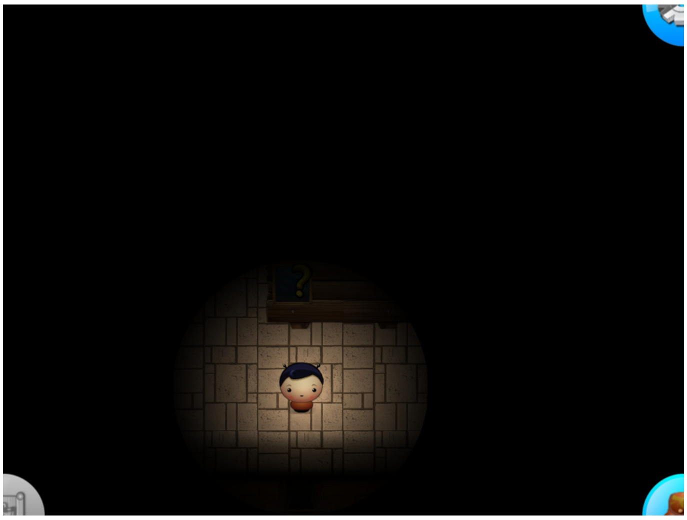

Figure 3.4: A room with no lights

To access puzzles the player has to select some items or non-playing characters on screen. Solving a puzzle, the player receives a certain amount of creativity points and may receive an item which will be stored into the inventory.
To open the inventory the player has to access the game menu by pressing buttons located in the corners of the screen (Figure 3.5).

Figure 3.5: Game menu

Figure 3.6: Inventory

The menu also shows the map of the visited rooms and reports information about positions, connections, light sources and closed doors (Figure 3.7).

Figure 3.7: Castle map

3.2 Puzzles

In CASTLE there are three puzzle types, divided into *verbal*, *spatial* and *visual*. The player receives a certain amount of creativity points by solving a puzzle. In some puzzles, the player can ask for hints. If hints are used the player gains less creativity points completing the puzzle. It is possible to check the current creativity level by opening the game menu. In addition to the creativity points, puzzles may reward the player with an item. Items work as keys to open castle's doors.

Verbal puzzles

In CASTLE there are three types of verbal puzzles: *riddles*, *associations* and *sequences*.

Riddles are statements or questions having a double or veiled

meaning, requiring thought to answer or understand. The player can ask for 1-3 hints, the longer the solution to the riddle, the more hints the player can ask for. Each hint reveals a letter of the solution. If the player fails three times the puzzle won't be accessible anymore.
An example of *riddle* is: "Al suo passaggio tutti si tolgono il cappello. Ha i denti ma non morde." The correct answer is: "Pettine" (Figure 3.8).

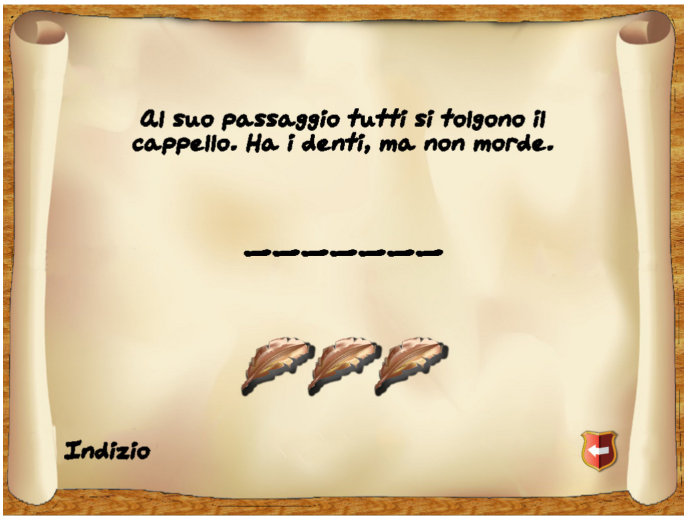

Figure 3.8: Riddle example

Associations are puzzles in which three words are given and the player is challenged to find a fourth one that connects them all. The player can ask for 1-3 hints and has three tries, as in riddles. An example of *association* is: "Cosa lega queste tre parole? Aria - Scrivere - Continuo?". The correct answer is "Getto" (Figure 3.9).

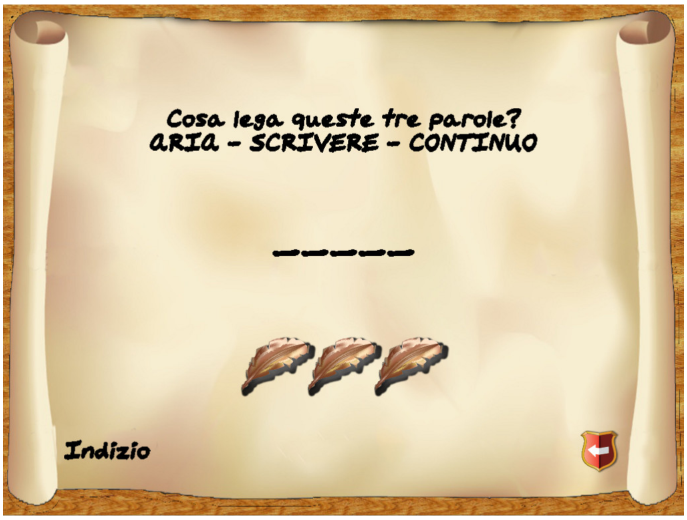

Figure 3.9: Association example

Sequences are puzzles in which the player has to find the correct answer from a short definition. If the answer given by the player is wrong, another part of the definition is shown and the creativity gained in case of success decreases. The player has five tries.
An example of *sequence* definition is: "E' giallo". If the players gives a wrong answer, the next definition will appear: "Vive in Africa". The correct answer is: "Ghepardo" (Figure 3.10).

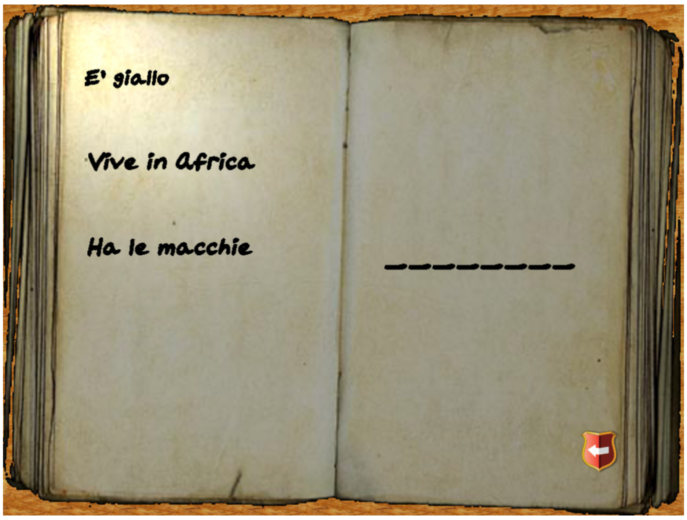

Figure 3.10: Sequence example

Spatial puzzles

In CASTLE there is one type of spatial puzzle: *matchsticks*.

In *matchsticks* puzzles the player receives a number of matchsticks arranged as squares, rectangles or triangles. The player has to rearrange matchsticks to make a target figure. There are no hints and the player has an unlimited amount of tries. The player can also reset the position of all the matchsticks by pressing the "Reset" button in the lower right corner.
An example of *matchsticks* is "Sposta tre bastoni per ottenere tre quadrati congruenti", shown in Figure 3.11. The solution is shown in Figure 3.12.

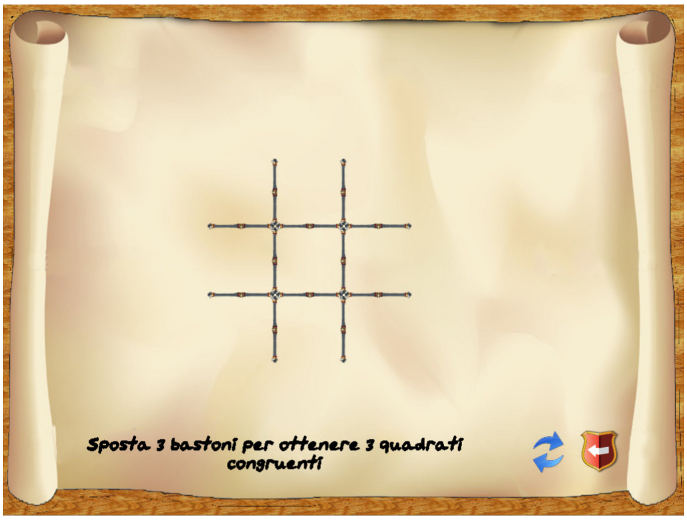

Figure 3.11: Matchsticks example

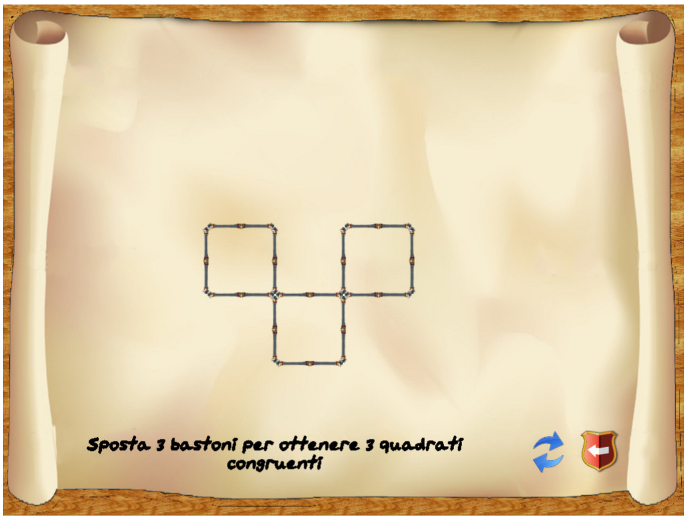

Figure 3.12: Solution of the matchsticks example

Visual puzzles

In CASTLE there are three types of visual puzzles: *details*, *ambiguous images* and *find-it*.

Details are puzzles in which a detail of a picture is shown and the

player has to understand which object it represents. If he fails another detail appears and the creativity gained in case of success decreases. If the player fails three times the puzzle won't be accessible anymore. It is also possible to ask for the next detail without trying to give a solution.
An example of *details* is shown in Figure 3.13. The correct answer is "Anello".

Figure 3.13: Details example

Ambiguous images are images which exploit graphical similarities and other properties of visual system interpretation between two or more distinct image forms to provide multiple, although stable, perceptions. Classic examples of this are the rabbit/duck and the Rubin vase [23]. Given an ambiguous image, the player has to identify the two forms. There are no hints and the player has an unlimited amount of tries.
An example of *ambiguous images* is shown in Figure 3.14. The

correct answers are "Anatra" and "Coniglio".

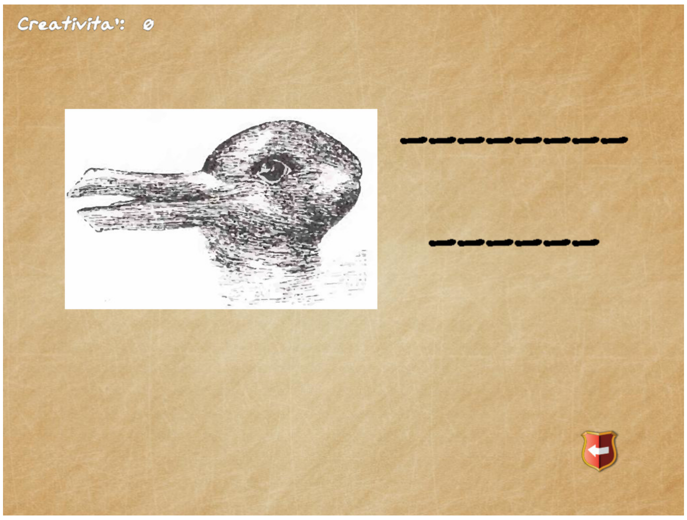

Figure 3.14: Ambiguous image example

In *find-It* puzzles the player has to find every occurrence of a specific object within an image. The images are full of different objects in order to raise confusion. There are no hints and the player has an unlimited amount of tries but after a certain number of wrong tries the player is forced to exit the puzzle and start over.

An example of *find-it* is shown in Figure 3.15. In this example the player has to find every heart in the figure.

Figure 3.15: Find-it example

3.3 Lateral thinking in CASTLE

CASTLE's puzzles are developed to offer the player the possibility to train and challenge lateral thinking.

In *riddles* and *sequences* the player is asked to answer a question given a definition. However, the definition is often vague and confusing: words used in definitions often have double meaning and the player has to analyze the sentence in multiple ways. An example is: "What falls, but does not break, and what breaks but does not fall?". This puzzle, whose answer is "night and day" (night falls and day breaks), is based on the double meaning of the words "fall" and "break". Similarly in *associations* the solution word has completely different meanings when referred to one of the three given words.
Matchsticks puzzles are designed to challenge spatial visualization,

the ability to mentally manipulate two and three-dimensional figures. They are the digital version of a classical spatial-visualization game. Moreover, researches by the University of Toronto have discovered that differences between men and women on some tasks that require spatial skills are largely eliminated after they both play a video game for only a few hours [24]. According to a 2009 study in Current Directions in Psychological Science, playing video games enhances performance on mental rotation skills, visual and spatial memory, and tasks requiring divided attention [25].

Ambiguous images are well-know tools used in psychology experiments in which the player has to recognize two items drawn in an ambiguous figure. This process trains middle-level vision. Middle-level vision is the stage in visual processing that combines all the basic features in the scene into distinct, recognizable object groups. This stage of vision comes before high-level vision (understanding the scene) and after early-level vision (determining the basic features of an image). When perceiving and recognizing images, middle-level vision comes into use when we need to classify the object we are seeing. High-level vision is used when the object classified must now be recognized as a specific member of its group. For example, through middle-level vision we perceive a face, then through high-level vision we recognize a face of a familiar person.

In *details* the small visible portion of the image is not a key-part of the image itself and to give the correct answer the player has to use creativity and imagination. *Find-it* works the same way: the items to find are often placed so that only a detail is visible. To help the use of lateral thinking, both in *find-it* and *details* puzzles some items do not have their natural color (e.g. hearts are white, not red).

3.4 Tracing

CASTLE implements a mechanism for data collection which creates a log file each time the game is played. A log file contains the following information:

- The seed used to generate the game contents
- Name and sex of the character
- The time spent in each room. If the player exits a room and comes back later, two different values are registered
- The time spent trying to solve each puzzle. If the player exits the puzzle screen and comes back later, two different values are registered
- The total time passed in each castle
- The exploration time (e.g. total time minus the time spent in puzzle) for each castle
- The exploration paths for each castle

- The number of hints used
- The number of puzzles solved
- The number of puzzles failed
- The number of puzzles left behind
- The number of touches of the screen for each room. If the player exits the puzzle-view and comes back later, two different values are registered
- The number of times the player opens the inventory
- The number of times the player opens the map

3.5 Summary

In this chapter, we have described the structure of the game CASTLE and why it can be used to train lateral thinking.
In the next chapter we describe the tools used for the development and some implementation algorithms.

Chapter 4

Tools and implementation

CASTLE was developed using the C++ programming language and Cocos2d-x game engine. Game content is stored in XML format and parsed using the PugiXML library. In this chapter, we describe these tools. We also describe the algorithm used to generate maps using Procedural Content Generation and our implementation of the A* algorithm used to move the player's character.

4.1 Cocos2d-x

Cocos2d-x [26] is a free cross-platform open-source engine for 2D game development that we used to create CASTLE. It includes a set of C++ libraries for handling game graphics, physics, audio, scripting and user input.

4.1.1 Game structure in Cocos2d-x

In Cocos2d-x, a game is structured as a set of *scenes*. A scene is a tree structure that stores elements to draw on the screen.
A scene represents a single section (e.g. a level or a menu) of the game (Figure 4.1). A scene can contains all the game elements, including sprites, buttons, particle effectes, etc. When rendered, the scene first draws the last items added providing automatic culling. In Cocos2d-x a Scene is rapresented by the `CCScene`[1] class. Elements can be added to the scene using the `addChild` method of `CCScene`.
This Cocos2d-x feature allows the creation of a very hierarchical structure of the game: at any moment, the engine has one and one only active `CCScene` class reference working as the root of the current scene. Every item added to the active scene reference will thus be rendered on the screen.

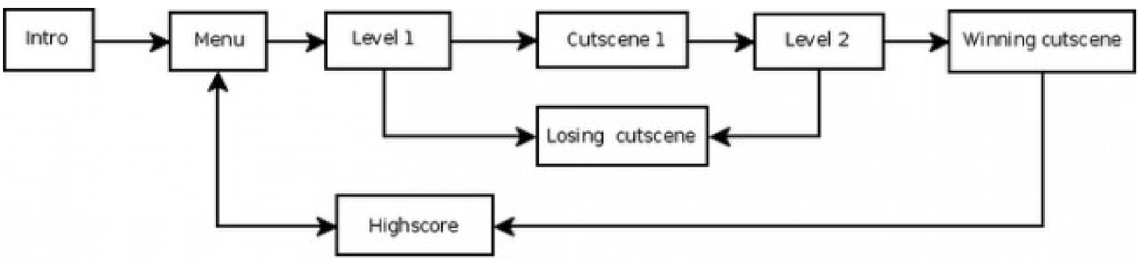

Figure 4.1: Scenes flow example

In the game there may be several stored `CCScene` instances and we can switch from one scene to another. Additionally, we can specify the transition effect (e.g. fade out, fade in) and duration. When

[1]Every Cocos2d-x class uses the 'CC-' prefix

switching scene, the engine automatically releases the source one, freeing up memory. CASTLE is organized in four main scenes: a scene for the game menu (`MenuScene`), a scene for the world map (`WorldMapScene`), a scene representing a room (`RoomScene`) and a scene representing a puzzle (`QuizScene`); the engine constantly changes the rendered scene accordingly to the user input.

4.1.2 Memory management

Cocos2d-x features automatic memory management. It implements an autorelease pool in a similar way to the one used in ObjectiveC [27]. Whenever a Cocos2d-x object is created, it is automatically added to the autorelease pool. At the end of the current code section, every item in the autorelease pool will be automatically released, unless they've been added to a scene or retained using the `retain` method. When an item has been retained this way, the programmer has to release it later on using the `release` method, to avoid memory leaks.

This may be useful to avoid the automatic release of an object that we may need later: we may choose to retain it to save the creation computational time. For example, when a transition from the `RoomScene` to the `QuizScene` occurs, we entirely retain the `RoomScene` in order to avoid to create it from scratch when we go back to the `RoomScene`.

4.1.3 Actions and Callbacks

Cocos2d-x uses `CCAction` to move, rotate or manipulate sprites. When a `CCAction` is created, it is associated to a target (e.g. the player's sprite) and launched with the target's `runAction` method. There are several kinds of `CCAction`, such as `CCMoveTo` which allows to move the target from its actual position to a new point on the screen. It is also possible to organize multiple `CCAction` objects in a `CCSequence` and then run the entire sequence. During the execution of a sequence, actions are sequentially executed one at a time. `CCSequence` is particularly interesting because of the possibility to execute not only CCAction objects but also functions. These functions, called callbacks, are treated as a `CCAction` and invoked in the same way. Similarly, `CCMenuItems` which represents menu buttons, can be associated with a callback function so that when the user presses a button, the associated method is called.

4.2 PugiXML

Game puzzle content is stored in XML files. We used a C++ XML processing library named PugiXML [28] to parse the content of the XML files and load puzzle descriptions.

4.2.1 XML parsing

PugiXML loads an XML document as a tree data structure that has one or more child nodes. Nodes may have different types; depending on the type, a node can have a collection of child nodes, a collection of attributes, and some additional data (e.g. name) (Figure 4.2).

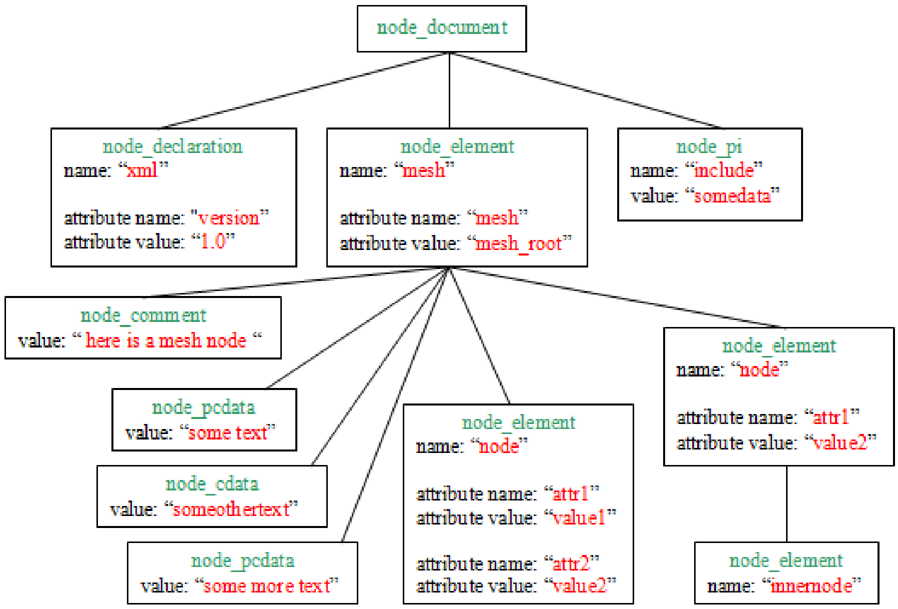

Figure 4.2: Example of a xml tree structure in PugiXML

Once loaded, we can use the library methods to explore the tree from the root and retrieve attributes and data from nodes.

4.3 Procedural map generator

In CASTLE, we used Procedural Content Generation to create castle's maps and to place items and puzzles inside rooms. Procedural Content Generation (PCG) [29] is a term used to identify an algorithm for the automatic creation of random contents. This

approach is often used in video games to increase the amount of levels or challenges the player may encounter.
Given a NxN boolean matrix to describe a map where *true* means the presence of a room and *false* means otherwise, we designed maps with three constraints:

- every map had to contain a specific number of rooms (that is, specific number of *true* values in the matrix)
- two adjacent rooms had to be linked (that is, you are able to move from one to another)
- no rooms may be placed outside the matrix boundaries

At the beginning of a new game, we randomly generate a string. This string is used as a seed for the random calls of the algorithm. From here, we start with a single instance of Room class, representing a room. This class contains four empty references to other rooms that may be placed on the north, east, west and south sides. The algorithm places this room in the middle of the matrix, then it tries to place a new room on every side of it, sequentially. The placement is successful if:

- there is no already existing room on that side
- the new room is inside the boundaries of the matrix
- there is no adjacent room to the new one
- given the seed, a pseudo random boolean function returns *true* value

If the placement is successful the algorithm is applied recursively to the new room and a *true* value is set in the matrix, elsewise it tries to place a new room on another side. The algorithm stops when a certain amount of rooms are placed.

Once we have this data structure for the map, we proceed to populate every room except the first one with three puzzles. For each puzzle type puzzles are randomly selected using the same seed used for the map generation. The number of selected puzzles of every type is fixed.

4.4 Player movement

In the room scenes, in order to explore, change room and reach puzzles, the player is able to move the character touching any point on the screen. We divided the screen into a 6 per 8 tile matrix. The character has 48 locations accessible: he can move in a discrete way from the center of a tile to the center of another tile.
To compute the optimal path from the position of the character to the location clicked by the player we used the A* algorithm [30] with Manhattan distance heuristic. Whenever the Manhattan distance between A and B is greater than half the screen width dimension, we first compute a path from A to the midpoint of the line segment connecting A and B and then a second path from the midpoint to B. This way the A* algorithm computes smaller

paths, which require less computational time. As shown in figure 4.3 and 4.4, this approach also helps to avoid long paths along the borders of the screen.

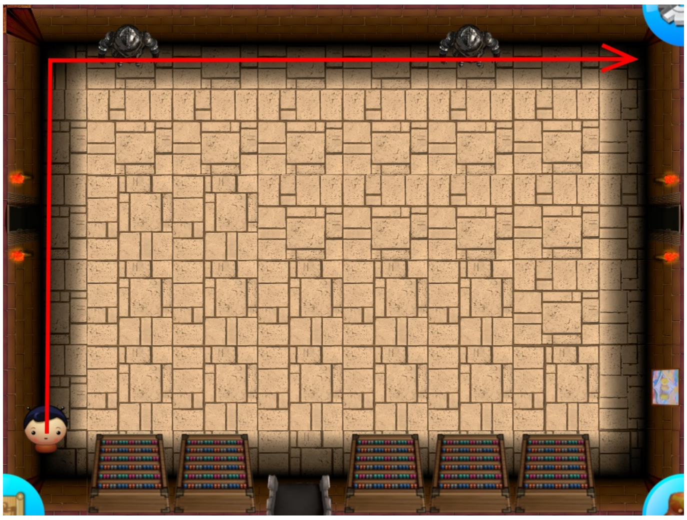

Figure 4.3: Player movement with standard A*

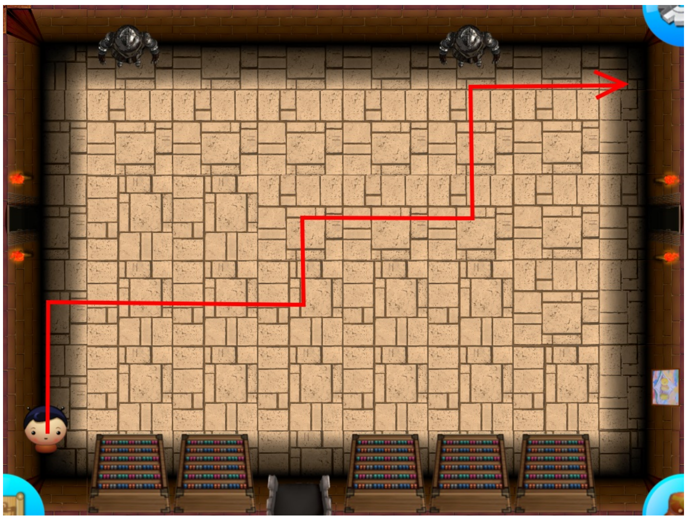

Figure 4.4: Player movement with our implementation of A*

4.5 Summary

In this chapter we described the tools used during the development of CASTLE. In particular we described the main features of Cocos2d-x and PugiXML. We then detailed the algorithm used to randomly generate maps using PCG and the algorithm used to make the player move in the game using the shortest path.
In the next chapter we describe methods and results of the research.

Chapter 5

Data analysis

In this chapter we present our empirical study to test whether a subject who frequently plays video games performs better at lateral thinking problems and how the game platform affects the cognitive performance of the subjects. We describe the methods we used for collecting the data from the subjects, the statistical analysis and our findings. Through data collection and analysis, we have studied how people with different backgrounds (e.g. students from different universities) reacts to the lateral thinking problems proposed in CASTLE. We also tested whether the device used to play the game influences the use of lateral thinking of the players.

5.1 Methods

In collaboration with with the Università Cattolica del Sacro Cuore, we validated CASTLE with 72 students playing the game on different platforms (personal computer, tablet and paper). We applied data mining techniques to analyse players' behaviour with respect to cognitive skills and lateral thinking.

5.1.1 Partecipants

We collected data from 72 subjects, divided into two groups of 36 players each: the former group was composed of students of engineering, the latter of students of humanities. All the subjects were volunteers and gained no credits for performing the test. The subjects were between 19 and 32 years old; the average age was 23.35 years. One of the subjects was discarded from the analysis because his age was outside the range of interest. The genre was not balanced, beacuse it was not considered an important dimension of analysis.

5.1.2 Materials

Each student had to complete a questionnaire and to play the game. We provided three versions of the game: a tablet version, a personal computer version, and a paper version. The paper version consisted in a short game-book that allows the player to explore various narration paths through the use of numbered pages, and proposed the same puzzles of the tablet/personal computer version.

5.1.3 The questionnaire

The questionnaire had four sections. The first section contained questions about video game habits such as how much time the subjects dedicate to electronic devices, video games and puzzle games in general and what they think about various aspects of video games. The second section contained the Object-Spatial Imagery Questionnaire (OSIQ) [31], a self-report questionnaire designed to distinguish between two different types of imagers: object imagers who prefer to construct vivid, concrete and detailed images of individual objects (e.g. visual artists) and spatial imagers who prefer to use imagery to schematically represent spatial relations among objects and to perform complex spatial transformations (e.g. scientists). The third section contained the Barratt Impulsiveness Scale (BIS) [32], a questionnaire designed to assess the personality/behavioral construct of impulsiveness. The BIS is scored to yield three factors: *attentional* impulsiveness, *motorial* impulsiveness and *nonplanning* impulsiveness. The last section contained the Torrance Tests of Creative Thinking (TTCT) [33]. The subjects had to draw six images given a number of starting lines. The sequence of images had to tell a story. The subjects also had to give a title to each image. The TTCT scores the results on four scales: *fluency*, the total number of interpretable, meaningful, and relevant ideas generated in response to the stimulus; *flexibility* the number of different categories of relevant responses; *originality*, the statistical rarity of the responses; *elaboration*, the amount of detail in the responses.

5.1.4 Procedures

Subjects played CASTLE for 12 minutes after completing the questionnaire. We divided each 36 people group into three subgroups, thus each subgroup played on a different platform. The game also provides the possibility to choose one of three predefined maps instead of a random generated one. These maps are identified as map A, map B and map C. For each platform we equally tested every predefined map. Table 5.1 sums up the distribution of test subjects.

		Map A	Map B	Map C	
Engineering	Ipad	4	4	4	12
	Pc	4	4	4	12
	Paper	4	4	4	12
Humanities	Ipad	4	4	4	12
	Pc	4	4	4	12
	Paper	4	4	4	12
		24	24	24	72

Table 5.1: Test subjects distribution

5.1.5 Data analysis

The collected data, consisting in compiled questionnaires and game logs, were organised into a spreadsheet file and analysed using SPSS 20, a software for statistical and data mining analysis.
Initially we preprocessed some of the attributes as follows:

- We used players' paths through rooms in CASTLE to divide players into three categories: *stationary*, players who have tendencies to stay close to a specific room; *explorers*, players who have tendencies to explore every room in the map; *straightforward*, players who have tendencies to follow one direction exploring the rooms

- Answers to the open question "Why do you think people should play video games?" were categorised into "to improve cognitive skills", "to gain control of emotions" and "fun"

For the analysis we used the chi-squared (χ^2) test, a test of independence that assesses whether paired observations on two variables, expressed in a contingency table, are independent of each other distribution. It tests a null hypothesis stating that the frequency distribution of certain events observed in a sample is consistent with a particular theoretical distribution.

5.1.6 Performance

We scored players' performance using the following indexes:

- For puzzle performance we considered to total number examined puzzles and the relations between the number of solved and examined puzzles, failed and examined puzzles and left and examined puzzles
- For exploration performance we considered the total number of explored rooms and how much the subject was oriented to exploration

5.2 Results

Data analysis showed no evidence that university of origin or subjects' genre affect the subjects' performances.
Our analysis shows that there is no evidence that impulsiveness and creativity dimensions are related.
It also shows that the impulsiveness is significantly related to time spent playing video games (χ^2 (1, 54) = 6,629, p < 0,05): the 77,1% of the subjects that spend less than 1,5 hours a day playing video games, obtained low motorial impulsiveness scores; the 57,9% of the subjects that spend more than 1,5 hours a day playing video games, obtained high motorial impulsiveness scores.
Our analysis shows that scores of the Object-Spatial Imagery Questionnaire section of the questionnaire significantly differ with respect to the field of study: humanities students obtained higher scores in object visualization (χ^2 (2, 71) = 8,507 p < 0,05) while engineering students obtained higher scores in spatial visualization (χ^2 (2, 71) = 8,573 p < 0,05). Scores of the TCTT section differ with respect to the university of origin: humanities students show more originality, obtaining significantly higher scores (χ^2 (2, 67) = 8,011 p < 0,05).
The analysis reports that the device significantly influenced the performances of the subjects: the subjects who played the tablet version or the personal computer version completed more puzzles (χ^2 (3, 71) = 16,314 p < 0,001), used more hints (χ^2 (3, 71) = 16,903 p < 0,001) and failed less puzzles (χ^2 (2, 71) = 12,756 p < 0,01). There are no significant differences between players who

played the personal computer version and players who played the tablet version. This suggested to re-encode the device variable with only two dimensions: digital and paper.
The analysis shows that the total number of examined puzzles (variable considered only for the digital versions) is significantly related with the originality dimension measured by the TCTT: original subjects read more puzzles (χ^2 (4, 44) = 12,300 p < 0,05) and left more puzzles (χ^2 (4, 67) = 10,007 p < 0,05).
The number of errors is significantly related with the object visualization scale of the OSIQ: subjects with higher object visualization scores have the tendency to fail more puzzles (χ^2 4, 71) = 10,161 p < 0,05). It is also related with motorial impulsiveness: the higher the motorial impulsiveness score is, the more errors the player made (χ^2 (2, 71) = 8,346 p < 0,05). Errors are also related to the time spent playing video games: the more is the time spent, the less is the number of errors (χ^2 (2, 54) = 7,958 p < 0,05).
Map C (Figure 5.1) seems to encourage exploration: the 86,7% of the subjects who played that map explored at least four rooms.
Map A (Figure 5.2) seems to discourage exploration: the 75% of the subjects who played that map explored less than four rooms.

Figure 5.1: Map C

Figure 5.2: Map A

The object visualization scores are significantly related to exploration: higher scores mean higher exploration time (χ^2 (2, 47) = 7,289 p < 0,05) and a higher tendency to explore every room in the castle (χ^2 (4, 47) = 13,459 p < 0,01).

5.3 Summary

In this chapter, we presented methods and results of our analysis. The analysis suggested that the digital version of the game stimulates subjects more than the paper version and that there are no significant differences between players who played the personal computer version and players who played the tablet version. It also suggested that the number of errors made by the subjects is related to the time spent playing video games: gamers commits less errors.

Chapter 6

Conclusions and further work

CASTLE appears to be an useful tool to study various aspects and correlations of cognitive skills regarding lateral thinking. In particular the analysis suggested that a digital version of a game stimulates lateral thinking processes of the subjects more than a paper version. We did not reported significant differences between perfomances of the players who played the personal computer version and the palyers who played the tablet version. This may be due to the fact that by design CASTLE didn't take advantage of the full potential of touch devices. In addition, we noticed that the map influenced the behaviour of the players. This should be taken into account in further analysis.
The analysis also shows that the number of errors made by the subjects is related to the time spent playing video games: gamers commits less errors. This may suggets that gamers perform better in lateral thinking problems.
Even if analysis' results may suggests this conclusions, further analysis with an higher number of subjects are required. With

long term analysis it may be possible to study if the game actual improves lateral thinking skills.

Finally, we plan to submit a free modified version of the game with more castles and puzzles, and without data collection, to the App Store.

Bibliography

[1] John Kirriemuir; Angela McFarlane. Literature review in games and learning, 2004.

[2] Paul Gee. What video games have to teach us about learning and literacy, 2007.

[3] www.medicalnewstoday.com/articles/228449.php. Brain exercise video games can improve kids' school performance, 2011.

[4] D. Bavelier. Action video games improve vision; www.rochester.edu/news/show.php?id=3342, 2009.

[5] `www.edwdebono.com`.

[6] P. Sloane. The leader's guide to lateral thinking skills: Powerful problem-solving techniques to ignite your team's potential, 2003.

[7] Yann Landrin-Schweitzer; Pierre Collet; Evelyne Lutton. Introducing lateral thinking in search engines, 2006.

[8] B. G. Weber; M. Mateas. A data mining approach to strategy prediction, 2009.

[9] T. Mahlmann; A. Drachen; J. Togelius; A. Canossa; G.N. Yannakakis. Predicting player behavior in tomb raider: Underworld, 2010.

[10] C. Thurau; C. Bauckhage. Analyzing the evolution of social groups in world of warcraft, 2010.

[11] `www.edwdebono.com/lateral`.

[12] Catherine Greenhill. Sudoku, logic and proof, 2011.

[13] C. L. Bouton. Nim, a game with a complete mathematical theory, 1901.

[14] Paul Sloane; Des MacHale. Lateral thinking puzzlers, 1992.

[15] `www.mathpuzzle.com/loyd`.

[16] Peter Vorderer; Jennings Bryant. Playing video games: Motives, responses, and consequences, 2006.

[17] Achtman; R. L. Green; Bavelier D. Video games as a tool to train visual skills, 2008.

[18] Nick Tannahill; Patrick Tissington; Carl Senior. Video games and higher education: What can call of duty teach our students?, 2012.

[19] Rui Nouchi; Yasuyuki Taki; Hikaru Takeuchi; Hiroshi Hashizume; Yuko Akitsuki; Yayoi Shigemune; Atsushi Sekiguchi; Yuka Kotozaki; Takashi Tsukiura. Brain training game improves executive functions and processing speed in the elderly: a randomized controlled trial, 2012.

[20] Linda A. Jackson; Edward A. Witt; Hiram E. Fitzgerald; Alexander von Eye; Yong Zhao. Parent behavior, children's technology use and creativity: Videogames count but parents don't!, 2011.

[21] John H. Doolittle. Using riddles and interactive computer games to teach problem-solving skills, 1995.

[22] Elizabeth Huttona; S. Shyam Sundarb. Can video games enhance creativity? effects of emotion generated by dance dance revolution, 2010.

[23] Edgar Rubin. Synsoplevede figurer, 1915.

[24] Jing Feng; Ian Spence; Jay Pratt. Playing an action video game reduces gender differences in spatial cognition, 2007.

[25] Matthew Dye; Shawn Green; Daphne Bavelier. Racing, shooting, and zapping your way to better visual skills, 2009.

[26] `www.cocos2d-x.org`.

[27] `www.developer.apple.com/library/mac/#documentation/Cocoa/Conceptual/MemoryMgmt/Articles/MemoryMgmt`.

[28] `www.pugixml.org`.

[29] `www.pcg.wikidot.com`.

[30] S. Russell; P. Norvig. *Artificial Intelligence - A Modern Approach.* Prentice-Hall, third edition, 2010.

[31] Olessia Blajenkova; Maria Kozhevnikov; Michael A. Motes. Object-spatial imagery: a new self-report imagery questionnaire, 2006.

[32] Jim H. Patton; Matthew S. Stanford; Ernest S. Barratt. Factor structure of the barratt impulsiveness scale, 2006.

[33] E. P. Torrance. Torrance tests of creative thinking, 1974.