Single Button Interaction Technique Patterns Applied to Digital Game Design

Felipe Breyer Judith Kelner Eduardo S. Albuquerque* Luis Vasconcelos**

UFPE, Centro de Informática, Brazil *UFG, Instituto de Informática, Brazil **UFPE, dDesign, Brazil

ABSTRACT

This paper investigates how interaction design can be applied to the design of digital games, more specifically, to the player controls scheme definition and user experience. To achieve this objective, this work studies three methods of knowledge formalization, ontologies, tropes and patterns, and as a case study the digital button was chosen as the interaction device to be researched. Initially, a set of games was observed through the inductive research method in order to define analysis parameters that enabled modeling user interaction with the game. Thus, we extend the pattern library of game design, creating a set of interaction technique patterns applied to the button device.

Keywords: Patterns, interaction, technique, games, button

Author's contact:
{fbb3, jk}@cin.ufpe.br
*eduardo@inf.ufg.br
**lalv4401@gmail.com

1. Introduction

The role of the game designer is to create ludic experiences for their audience through the games he or she conceives. Games are not relevant unless they are played and the players have a good experience during the game [Schell, 2008]. The player experience, in turn, is unique and non-transferable and, although the game designer indirectly designs them, he must have tools to guide the players during their journeys. Interaction design is the border between the game and the player, i.e. the border where the designer can approach the audience, and this is where the efforts of this research are focused.

One of the differences between production software and digital games is that software should reduce the workload of the user while games should make it more difficult over time to maintain player interest [Lazzaro and Keeker, 2004]. Therefore, the game designer can explore interaction possibilities that would be unacceptable in the context of production software. More specifically, the aim of this paper is to identify these unconventional forms of interaction to provide greater consistency in the designer choices during the conception phase of the game design.

The remaining sections of this paper are outlined as follows. Section 2 briefly describes the concept of interaction techniques and classification of interaction devices. Section 3 analyzes the ways of classifying formal elements that will structure the results to be obtained in the research. Section 4 discusses the current status of pattern languages usage applied to the design of digital games. Section 5 explains the object of study and the methodology applied in the experiment. The patterns proposed in this article are described in section 6. Section 7 contains the main conclusions and discusses future work to be carried out to extend this analysis.

2. Interaction Techniques and Interaction Devices Classification

Hinckley, Jacob and Ware [2007] define interaction techniques as a way to use a particular interaction device to perform a generic task interaction. For example, the pop-up menu is an interaction technique to allow the user to select a specific command from a small set. This action is accomplished through interaction devices that are peripheral to the computer, and allow user interaction, such as mouse and monitor. In an atomic way, an Interaction Task is the user-supplied primitive input at the lowest level, such as choosing a command or typing a line of text.

According to Breyer et al [2008] we can classify the interaction devices according to twelve senses: touch, life, movement, balance, smell, taste, vision, warmth, hearing, sense of language or word, thought, and ego. This article is restricted on the tactile input devices, a category that includes devices that are common to most computers, such as keyboards.

A digital signal is the smallest amount of information a user can send to a system and the simplest device used to trigger the transmission is the button. There are devices that provide different amounts and combinations of buttons from the QWERTY keyboard to a variety of joysticks for video games.

The main features observed in continuous inputs are the type of measurement, the type of movement, the

physical property registered and the number of degrees of freedom. Continuous information can be modified using a direct assignment of a new absolute value or an increase to the old value, indicating a relative transform. There are two types of movements: linear as in a slider or rotatory like a knob. A linear device can measure the position of the marker or the force applied and the corresponding properties for the rotary devices are angle and torque. Typically, a system can record from one to three linear dimensions and/or from one to three angular dimensions, respectively translation and rotation in the three dimensional axes X, Y, Z.

The possibilities of using the button as an interaction device have also been the subject of the work by Green [2005], however, his work describes only the interaction possibilities in the abstract, such as: "Keep button pressed = activates action = A. Release button = activates action B. " On the other hand, the research here described adopts an analysis centered on the user experience, giving meaning to different forms of interaction, rather than abstractions like "Action A" or "Action B". Therefore we try to identify the actions taken within the games and the transactions associated with them, for example, to make a character perform the action of jumping, the player must press a button. And from the survey of these associations, we are able to structure the knowledge surrounding the interaction techniques, as we shall see in more detail in the results presented in section 6.

3. Classification of Formal Elements: Ontologies, Tropes and Patterns

It is of interest to the game designers community to build a formal language that facilitates communication between its members as Costikyan [1994] and Church [1999] reported. In order to sort and arrange the formal elements that make up a game you can use ontologies, patterns or tropes, as described below.

Ontologies are intended to verify the existence of real or abstract elements, categorize them and identify their relationships [Aristotle apud Cochiarella, 2007]. In the context of digital game design, Zagal et al [2005] propose an ontology of games, the Game Ontology Project (G.O.P.). It is divided into four categories: interface, rules, entity manipulation and objectives. Each element of the ontology has the following properties: name, parent element, children elements, description, strong example and weak example. The items "parent element" and "children elements" are used to create relationships between entities. Altogether, there are approximately 180 elements that make up the ontology with examples that note only the presence of the element in a particular game. However, the ontology describes only the existence of elements and not how to use them. In this manner, the ontology does not provide the necessary structure for the study of interaction techniques applied to games.

In turn, tropes are described in literature as the main idea of a story, like the *cliche*, but function as figures of speech [Merriam-Webster Dictionary, 2011]. The project TvTropes [TvTropes, 2009] is open to various media such as animation, comics, television, literature, cinema and digital games, unlike what its name might suggest and it is proposed to be a library of tropes open to the community. Tropes have only a textual description and fields of examples, despite the vast database, which has registered about 700 records, the lack of categorization and information architecture makes it difficult to query and use these data.

The pattern language was created by Christopher Alexander [1977] to the area of architecture, with the aim of structuring the knowledge gained and describe solutions to known issues thus allowing future reference. However, it was with the growth of software production that language patterns were popularized mainly in the context of software engineering, for example, the patterns of Freeman et al [2004]. For the Design, the patterns language found application in the areas of interface design with Tidwell [2005] and interaction design with Borchers [2001] and Schummer, Lukosch [2007]. To define a pattern, the language requires that a field with the description of the solution, the context in which it is used, the consequences of its implementation and a practical example must be filled. Through the application of the pattern language, we can understand how to apply a certain solution and to which scenario it applies. Thus, in the context of this research, the pattern language presented the characteristics necessary for structuring knowledge, enabling an analysis of the functions of the object of study in a given context of use.

4. Game Design Patterns

In the context of digital game design, the research on design patterns was performed by Björk, Holopainen [2004] and extended by Folmer [2007]. Björk and Holopainen instituted eleven categories of patterns:

Game elements - physical or logical components that contain the state of the game as "avatars" and "enemies".

Resource and resource management - define the domestic economy of the game, as "consumers".

Information, communication and presentation - include the interface patterns dedicated to contain the game information, such as "progress indicators".

Actions and Events - describe the actions that the player can do during the game and changes caused to the game state, such as "aim and shoot".

Narrative structures, predictability and immersion - group the patterns used in games that have narrative purposes, such as "cut scenes".

Social interaction - describe the possibilities of social interaction mediated by the game, such as "cooperation".

Goals - motivations for establishing the behavior of the player in the game, like "rescue."

Goal structures - are patterns that determine the relationships between the goals of the game as "incompatible goals".

Game sessions - includes patterns for the preparation and closing steps of the game, such as "time limits".

Game mastery and balancing - are related to the balance of opportunities between the institutions participating in the game, such as "randomness".

Meta games, replayability and learning curves - including patterns that transcend the game environment, such as "extra-game consequences".

Björk and Holopainen distributed approximately 300 patterns within these categories and even with the eleven categories and a vast collection of patterns, authors still consider the library incomplete and encourage other authors to identify new patterns. Folmer [2007] contributed with a description of 26 new patterns for usability in games.

5. Object of Study and Research Methodology

The focus of this research is to identify and structure the interaction techniques applied to digital games through different interaction devices and associate them with their semantic functions during the game. In the scope of this article, however, it will be presented interaction techniques based solely on a single button.

As research subjects it was selected games of the genre action and adventure in real time in which the player controls a human avatar or with humanoid characteristics. The analysis involved several game platforms, among them as different generations of consoles such as Sega Genesis, Super Nintendo Entertainment System (SNES), PlayStation2, Xbox, Nintendo Game Cube, PlayStation3, Xbox 360, PlayStation Portable (PSP) and Nintendo DS. During the research, the question “game environment” that could take two or three dimensions, showed no significant impact by observing the actions of the characters in the games, so this item was discarded as a point of analysis.

The study followed the inductive methodology of research that is divided into three phases, namely: observation of phenomena, discovering the relationships between them, and generalization of the relationships [Marconi and Lakatos, 1982]. Following these phases, the first step in the research consisted of the record of the actions taken within the games together with the configuration of the controls assigned to each activity. For the inductive method to be used it is required a sample of the set under review. Thus, a group of over 50 action and adventure games was selected as listed below: Assassin’s Creed, Assassin’s Creed – Brotherhood, Assassin’s Creed 2, Batman: Arkham Asylum, Blackthorne, Brutal Legend, Bully, Castlevania – Dawn of Sorrow, Castlevania – Order of Ecclesia, Castlevania – Portrait of Ruin, Darksiders, Dead Space, Dead Space 2, Devil May Cry 3 – Dante’s Awakening, Devil May Cry 4, Flashback, Freedom Fighters, Gauntlet: Seven Sorrows, God of War, God of War 2, God of War 3, God of War: Chains of Olympus, Grand Theft Auto: San Andreas, Hitman: Blood Money, Ico, InFamous, Lord of the Rings: Return of the King, Marvel Ultimate Alliance, Megaman X, Megaman X2, Mercenaries, New Super Mario Bros, Ninja Gaiden Sigma 2, Okami, Onimusha: Dawn of Dreams, Prince of Persia, Prince of Persia: Sands of Time, Prince of Persia: Two Thrones, Prince of Persia: Warrior Within, Psychonauts, Resident Evil 4, Resident Evil 5, Rogue Galaxy, Shadow of Rome, Shadow of the Colossus, Sonic – The Hedgehog, Star Wars: Battlefront 2, Star Wars: Episode 3 – Revenge of the Sith, Star Wars: The Force Unleashed, The Matrix – Path of Neo, Tomb Raider Legend, Transformers - War for Cybertron, Uncharted – Drake’s Fortune, Uncharted 2 – Among Thieves, Vanquish, X-Men Legends, X-Men Legends 2: Rise of Apocalypse.

Thenceforth, using the empirical mapping of the relationships between the characters' actions and the control scheme assigned to each one, three parameters for classifications were defined: the effort required by the character to perform a certain action, the time length that the action demand and the context in which the action can be triggered, illustrated as variations of axes in Figure 1.

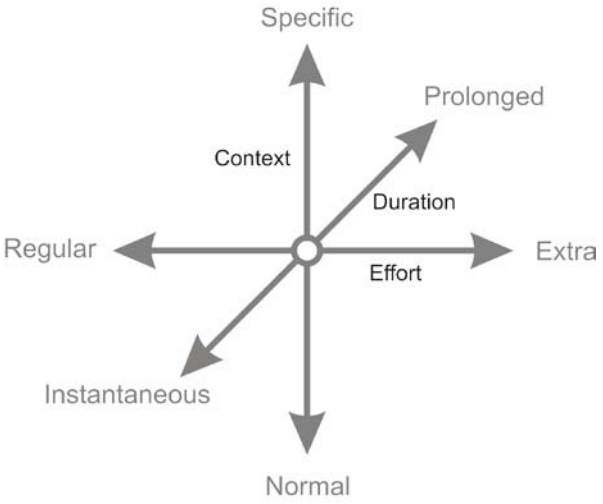

Figure 1: Axis of analysis categories and classification variations

Considering the effort in performing the actions it is possible to classify them as regular in which actions semantically associated to a low level of effort in relation to the capabilities of the character within the ambiance of the game, or as extra, so the actions of this category of tasks point toward approaching or even exceeding the limits of the capabilities of the heroes of the games. It is important to notice that the effort can be interpreted in different ways according to the characteristics of each game. For example, the action "jump" can be considered an action that takes effort to normal people, but for the character Sonic, in the game Sonic - The Hedgehog, where jumping is one of its main features in the game, this action does not represent nearly any effort. Thus, one action can be interpreted through different interaction techniques according to the theme of the game.

On the issue of action's duration, the tasks were categorized as instantaneous or prolonged. The instantaneous actions are those that require little more than a moment to happen, such as squeezing a trigger. Likewise, the extended actions are those that extend over time. Finally, we divide the actions in accordance with the context of the game in which they occur, and then, separating the regular actions that can be performed at any time by the player, and those that require a specific context to be accessed. From these observations, this research identified eight game interaction techniques patterns assigned to the device button. They are: press, pressing in the rhythm, keep pressed, press and release, precise pressing, rapid events, press and hold for limited time length and pump, with their properties described in Table 1 and positioned according to their characteristics in the analysis space in Figure 2.

Table 1: Classification of interaction techniques patterns for a button.

Pattern	Effort	Duration	Context
Press	Regular	Instantaneous	Normal
Rhythm MultiPress	Extra	Instantaneous	Normal
Hold	Regular	Prolonged	Normal
Hold and release	Extra	Prolonged	Normal
Precision press	Regular	Instantaneous	Specific
Quick time events (Q.T.E.)	Extra	Instantaneous	Specific
Time limited hold	Regular	Prolonged	Specific
Pump	Extra	Prolonged	Specific

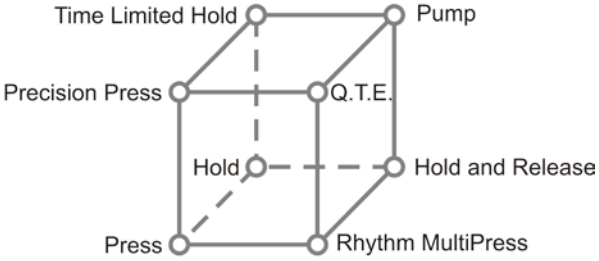

Figure 2: Positioning of the patterns of interaction techniques according to the variation of the axes in space analysis.

6. Interaction Techniques Patterns in Games

This section presents the patterns defined in our research, containing its description, usage and consequences of their application and example.

6.1 Press Pattern

Description: the player presses and releases the button quickly.

Usage: this interaction technique pattern should be used to a low effort action to the player character, with instantaneous response regardless of the context.

Consequences: due to its versatility, this pattern is the most appropriate to different actions, such as making the avatar jump or attack.

Example: in the game no Prince of Persia: The Sands of Time [Ubisoft, 2003], the player's avatar charges with sword blows with the simple press of a button as illustrated by Figure 3.

Figure 3: Prince attacks when the player presses a button.

6.2 Rhythm MultiPress Pattern

Description: The player presses the button in a specific rhythm so that the actions in the sequence have their results changed.

Usage: This pattern should be applied to modify, usually amplifying, the actions of the character in the game. By requiring the player to press buttons in a predetermined pace the player's character follows a different flow of actions. Normally this stream can be chained together in different combinations of actions which will continue as the player keeps the pace of pushing the button in the right rhythm.

Consequences: the designer must ensure that the response is clear to the user so that the player realizes that he is pressing the button in the correct rhythm.

Example: in the game Darksiders [Vigil Games, 2010], the player's character can deliver different strikes according to the rhythm the player presses the "X" button in the control configuration for Playstation 3, as described in Table 2 and illustrated in Figure 4.

Table 2: List of blow variations of the main character of Darksiders.

Strike	Commands
Forehand Strike	X
Return Slash	X, X
Double Slash	X, X, X
Reverse Backhand Strike	X, X, X, X
Whirl Wind Combo	X, “pause”, X
Tornado Slash	X, X, “pause”, X
Dragon’s Breath	X, X, X, “pause”, X


Figure 4: The main character of Darksiders throws different strikes according to rhythm the player presses the button.

6.3 Hold Pattern

Description: The player must press and hold the button. While the button state is changed the action will remain enabled and so is released the action must cease immediately.

Usage: This pattern should be used to replace low stress load, prolonged and accessible actions regardless of the context in which the player character is.

Consequences: When using this pattern the game designer should adjust the layout of the joystick functions so that the restriction of movement caused by holding the button pressed does not cause discomfort to the player.

Example: in the game Resident Evil 5 [Capcom, 2009], to make the character raise his weapon, the player must keep a button pressed as in Figure 5.

Figure 5: To raise his weapon in Resident Evil 5, the player must hold the button pressed.

6.4 Hold and Release Pattern

Description: The player must press and hold the button down for a while and then release it so that the action is triggered.

Usage: This pattern should be used to represent a long-acting action that requires extra effort from the player's character.

Consequences: the time that the button is pressed should be proportional to the intensity of the resulting action. There is always a limit for this intensity and time though.

Example: in the game Megaman X 2 [Capcom, 1994], the player can hold and release the button causing the character to fire beams with different energy levels (Figure 6).

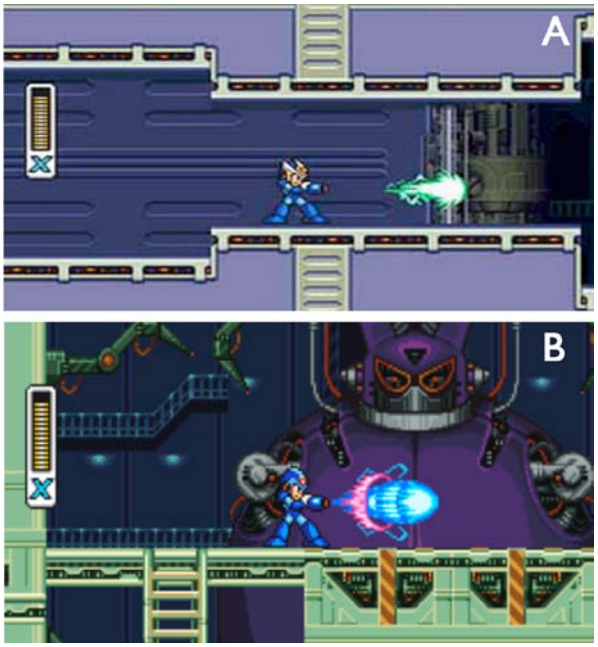

Figure 6: Shooting with the second level of stored energy (A) and shooting with the third level of stored energy (B).

6.5 Precision Press Pattern

Description: The player presses the button at a given moment in a specific context to activate a different action.

Usage: This pattern should be applied to instantaneous, low-effort and accessible only in special contexts actions. The rule that modifies the environment should be clear and previously presented to the player and also be prepared in advance. Like this, the user can wait for the right moment to press the button and successfully activate the action Precision Press.

Consequences: the game designer must determine which penalty the player should suffer in the event of failure of trying to run the Precision Press pattern. Thus, the game will remain balanced, requiring the player to analyze the risk of their choices.

Example: in the game Assassin's Creed 2, Ezio, the player's character, when being attacked has the ability to run a special defense that will leave the opponent momentarily helpless if the button is pressed at precisely the moment when the opponent strikes his blow, see Figure 7.

Figure 7: Ezio character evades the enemy attack with precision.

6.6 Quick Time Event (Q.T.E.) Pattern

Description: The player is prompted to press a button in a specific context in a short period of time.

Usage: This pattern should be used to represent actions that require extra effort from the player's character, more precisely, for actions that require quick reflexes in critical moments.

Consequences: When creating a scene that uses quick time events, the game designer should consider the possible division of the player's attention, since the player should keep their attention on events that take place both in the foreground, where the cue to push the button will appear, and in the background, where special animations are being shown. Therefore, the player can issue the required commands while being able to understand what is happening in the scene. It is essential to emphasize the importance of the timing the cue to press the button will appear, in order not to make it obvious, thus reducing the reflex demand of the player, turning the interaction ineffective.

Example: Kratos, character of the game series God of War [SCE Santa Monica, 2007], performs a series of risky maneuvers that require superhuman reflexes to defeat an enemy of epic scale. The player must press the button in a short period of time for the action to be successful, as illustrated in Figure 8.

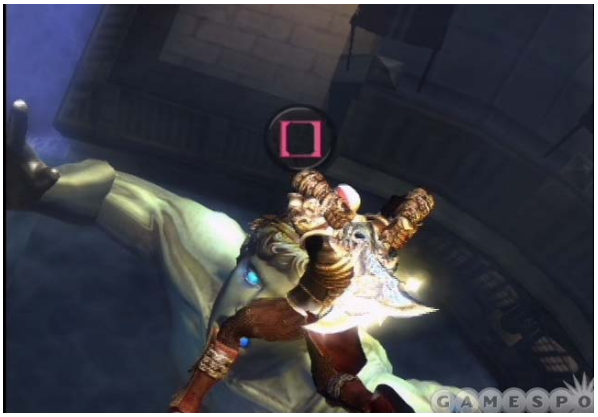

Figure 8: Kratos faces the Colossus of Rhodes in God of War II.

6.7 Time Limited Hold Pattern

Description: The player must keep the button pressed for a predetermined time so that the action is executed.

Usage: This pattern should be applied for long run actions with low effort in a specific context in the game environment. It can serve as a preparation or to represent the time that an action requires, however, the positive result is achieved only at the end of the time previously stipulated. In some cases, time can serve to check whether the user selected the correct action or not.

Consequences: It is important to understand the difference between this pattern from Hold Pattern. While in Hold Pattern the action remains active throughout the time that the player keeps the button pressed, the action of "Time Limited Hold" is only triggered when the timeout is reached. If the character suffers some type of interruption, the action will be canceled and the player must repeat the entire procedure.

Example: In multiplayer mode, to activate a system in a computer terminal, the characters in Dead Space 2 [Visceral Games, 2011] shall keep a button pressed for about 10 seconds, see Figure 9.

Figure 9: Characters in Dead Space 2 should activate computers by holding a button for 10 seconds.

6.8 Pump Pattern

Description: the player must rapidly and repeatedly press the same button to execute an action.

Usage: this pattern should be used to represent a prolonged action, that demands a great deal of effort by the player's character, and usually such action is available only in a particular situation. The amount of effort delivered by the character must be compatible with how many times the player must press the button in order to be successful in his attempt.

Consequences: due to the effort required by the pumping, the player may have difficulties performing other concurrent activities.

Example: in the game Bully [Rockstar, 2006], the Pump pattern is applied to fast pedaling while riding a bike, contextualizing the extra effort required by the action of the character, illustrated by Figure 10.

Figure 10: The player must pump the button to fast pedal while riding a bike to win the races in Bully.

7. Conclusions and Future Works

Although the player experience is unique, particular and non-transferable, it is necessary as well as feasible to infer user's behavior in order to foresee error events and even to provide a greater gaming experience. Due to the increasing market competition related to digital games, it is also possible to notice the growth of player's demand for games which offer higher levels of fun, in such a way that even the least development error can lead to public depreciation of a given product causing players to migrate to another similar game.

Related to the game design process as a whole, this research had its focus on the problem specification for interaction techniques, which are applied for the game controls scheme definition. Such approach is important since it aims to improve both understanding and formalization of the interaction-planning task between player and game. Supported by this method, the process of decision-making undertaken by the design team in order to associate a meaning or action to a given device will become more effective and efficient, so that players will associate more easily commands they want to execute to the interaction devices as well as to the feedback displayed on the game environment, providing a more pleasurable experience.

The inductive research methodology has achieved its purpose and made itself fundamental for defining analysis parameters of the players' avatars actions inside each cataloged game world as well as for the modeling of players' activities related to the interaction device chosen as research object. The 57 game samples were considered sufficient for extracting the information needed for performing this research. Observing the interaction techniques became redundant after approximately the 20th game analyzed, therefore the remaining elements on the control group were used

to verify registered associations on the first samples. Next, the pattern language was considered adequate for describing the interaction techniques functions, this way creating a neat association between tasks executed by the players and its meaning inside the game world.

In a broader view, adopting the pattern language brought visible benefits for this project, since the extension of a well-established game design library facilitates the understanding and appliance of the obtained results with this work on other research performed by professionals which already make use of game design patters for their games.

Despite the possibility of appliance of the proposed patterns for most of the analyzed games, some minor exception could be observed as in the God of War series, as shown in Figure 11. In these games, the main character Kratos may open chests placed along the levels, however for this particular case is applied the interaction technique of Time Limited Hold for a given action that could be easily performed with Pressing.

By analyzing the context, the player's avatar action would not have a long duration, inasmuch as the character has godlike powers and the chest has no special feature for greater locking. Nevertheless, the contents of chests vary from blue or green essences in cycles of a few seconds, and to inform the player of its current state, the chest brightens in a corresponding color. Changing the interaction technique in this case proves to be a wise decision of the design team, since it favors the player who makes time to check if he is opening the chest with the correct color, or whether he should stop the operation.

Figure 11: Kratos opens a chest with green essence.

For future work, it is necessary to expand the library of interaction techniques patterns to other interaction devices of which we can cite the analog joystick, touch screens with stylus pointers, devices equipped with accelerometers that record up to six degrees of freedom, multi-touch screens and body gestures tracking devices.

By expanding the library of interaction techniques patterns to the other devices, this research will contribute with a powerful tool to help game designers on building their games, hence contributing to the growth and development of the gaming industry and especially with the improvement of the player's experience.

REFERENCES

ALEXANDER, C., et al., 1977. *A Pattern Language: Towns, Buildings, Construction*, Oxford, Oxford University Press.

BJÖRK, S., HOLOPAINEN, J., 2004. *Game Design Patterns*, Hingham, Charles River Media.

BORCHERS, J., 2001. *A Pattern Approach to Interaction Design*, Chichester, John Wiley & Sons.

BREYER, F., et al., 2008. *Aplicação da Interação Natural em Sistemas de Entretenimento Virtual com Wiimote: Desenvolvendo a Bateria Virtual WillyKit*. São Luís, Anais do VIII Congresso Internacional de Ergonomia e Usabilidade de Design de Interfaces e Interação Humano-Computador.

CAPCOM, 1994. *Megaman X 2*, Osaka.

CAPCOM, 2009. *Resident Evil 5*, Osaka.

CHURCH, D., 1999. *Formal Abstract Design Tools*. Game Developer, San Francisco, CMP Media.

COCHIARELLA, N. B., 2007. *Formal Ontology and Conceptual Realism*, Dordrecht, Springer.

COSTIKYAN, G., 1994. *I have no words & I must design*. Interactive Fantasy.

FOLMER, E.. *Designing Usable and Accessible Games with Interaction Design Patterns*. Available at: http://www.gamasutra.com/view/feature/1408/designing_usable_and_accessible_.php?page=1. Accessed: December 2007.

FREEMAN, E., et al., 2004. *Head First Design Patterns*, Sebastapol, O'Reilly Media.

GREEN, B.. *One Button Games*. Available at: http://www.gamasutra.com/features/20050602/green_01.shtml. Accessed: April 2008.

HINCKLEY, K., JACOB, R. J.K., WARE, C.. *Input/Output Devices and Interaction Techniques*. Available at: http://www.cs.tufts.edu/~jacob/papers/crc2.pdf. Accessed: September 2007.

LAZZARO, N., KEEKER, K.. *What's My Method? A Game Show on Games.* Available at: http://www.xeodesign.com/whyweplaygames/whatsmymethod.pdf. Accessed: April 2009.

MARCONI, M. A.; LAKATOS, E. M., 1982. *Metodologia Científica*, São Paulo, Atlas.

MERRIAM-WEBSTER DICTIONARY. *Trope*. Available at: http://www.merriam-webster.com/dictionary/trope. Accessed: March 2011.

ROCKSTAR, 2006. *Bully*, Vancouver.

SCE SANTA MONICA, 2007. *God of War II*, Santa Monica.

SCHELL, J., 2008. *The Art of Game Design – A Book of Lenses*, Burlington, Morgan Kaufmann.

SCHÜMMER, T., LUKOSCH, S., 2007. *Patterns for Computar-Mediated Interaction*, Chichester, John Wiley & Sons.

TIDWELL, J., 2005. *Designing Interfaces*, Sebastapol, O'Reilly Media.

TVTROPE. *Main Page*. Available at: http://tvtropes.org. Accessed: April 2009.

UBISOFT, 2009. *Assassin's Creed 2*, Montreal.

UBISOFT, 2003. *Prince of Persia: The Sands of Time*, Montreal.

VIGIL GAMES, 2010. *Darksiders*, Austin.

VICERAL GAMES, 2011. *Dead Space 2*, Redwood City.

ZAGAL, J., et al., 2005. *Towards an Ontological Language for Game Analysis*. Vancouver, Proceedings of DiGRA 2005 Conference: Changing Views: Worlds in Play.