iCat, the Chess Player
Evaluating User Enjoyment in a Pervasive Chess Game

André Tiago Abelho Pereira

Resumo da Dissertação para obtenção do Grau de Mestre em

Engenharia Informática e de Computadores

Orientadora: Doutora Ana Maria Severino de Almeida e Paiva

Setembro 2007
ABSTRACT
In this paper, we present a preliminary study showing that the gaming experience involved in playing a pervasive game of chess on an electronic chessboard against a robotic agent is classified as more enjoyable by the users than playing the same game against a virtual embodied agent presented on a computer screen.

We describe a model for computerized pervasive chess game and an application implemented with such model. We detail two different setups: a setup using the iCat social robot, and another using a virtual tridimensional representation of the same robot. Using the two different setups, we present some preliminary results based on the pervasive game flow model that establish the gaming experience with the robotic agent as more enjoyable.

Keywords
Computerized Chess, Pervasive Gaming, Embodiment, Agent, User Enjoyment.

1. INTRODUCTION
Chess is usually a two player game, but computerized chess offers the opportunity to play against digital opponents of different strengths. Also for solo entertainment, many people nowadays recur to online chess in search of human opponents. There are many positive cognitive effects of chess instructions on students and there is a wide variety of research articles that demonstrate them [3] [4]. Research suggests that chess helps any person to elaborate exact methods of thinking [5]. So it would be particularly useful to start playing chess from the early school days. Most children prefer to learn something while playing, rather than to learn it formally. Children that received systematic instructions in chess improved their school efficiency in different subjects, in contrast with those who did not receive that kind of instructions. Computerized chess helps to spread the same positive cognitive effects of traditional chess instructions while surpassing the original game by allowing the user to play at any time against an opponent of custom strength, to analyse moves, consult game statistics, record games or consult saved games. Nevertheless, not everything in computerized chess is positive. By using a Graphical User Interface (GUI) the social possibilities of the game, in which both opposing players are able to interact, become limited and we cannot see hesitation or any expressed emotion on the virtual opponent. Playing chess in its original form, with a real chessboard and against a real opponent, share the social advantages, physical controls and physical information offered by most of the board games. To take advantage of both computerized chess and traditional chess game elements, it would be interesting to maintain the experience as close as possible to the traditional where we face an opponent and play in a real physical chessboard. Using the concepts of pervasive gaming, we aimed at restoring such aspects of the chess playing experience. Pervasive Gaming intends to create new and exciting gaming experiences that profit by the blend of real and virtual game elements [6]. The advances in pervasive computing technologies are slowly being used to improve some aspects of our daily activities such as entertainment. By researching the field of pervasive gaming we will seek for ways in which we can bring computerized chess back to the real world. In this article we present a model and a concrete implementation that shows how we can create a pervasive computerized chess game. Understanding what makes players enjoy a game is perhaps the most important issue in successful game design [7]. Thus, we want to evaluate if a more pervasive environment influences the user’s enjoyment in a chess game. Given that, the main research question addressed by this article is: Do people have more fun using the version of our implementation with a more pervasive (i.e. physically embodied) opponent? Or have they higher enjoyment while playing against a less pervasive one (i.e. virtually embodied)? To do so, we will assemble two different setups of our pervasive chess game: one using a physical embodied opponent, and other one with a virtual embodied opponent. Both of these setups use an electronic chessboard that automatically detects the pieces. The hypothesis of this article is: If we alter between these two different setups, we will be able to take conclusions about which scenario is more enjoyable.

2. RELATED WORK
This section presents two relevant fields in the context of our work: computerized chess and pervasive gaming.

2.1 Computerized Chess
When developing a chess application, there are essential components, such as the use of a graphical or tangible user interface and the use of a chess engine.

Graphical user interfaces enable us to visualize a virtual chess board and interact with it playing against chess engines or other players. One of the most known graphical user interface for chess software is Winboard or Xboard (its Linux version) an open-source Freeware by Tim Mann [8]. A comprehensive discussion of tangible user interfaces in game design has been provided by Ullmer and Ishii [9]. Tangible interfaces are more direct and natural to use than a GUI, whenever a possible virtual representation matches the physical representation of the interface. Using a tangible user interface the player alters the physical state of the chess board simply by moving a chess piece, as he would do in a normal two player chess game. The Cheesster tangible interface to computerized chess [10] can be used to play against a computer or a geographically remote opponent. Cheesster uses a large chessboard and thirty-two chess pieces with the top painted red so that the vision sensor can distinguish them from the board itself. A robotic arm with pincers is used to move the pieces around the chess board. DGT Projects [11] created a commercial board that looks like any standard chess set with a standard set of pieces. This board is available in two versions: Universal Serial Bus (USB) and serial. Each one uses a connector located at the side of the board to communicate with the computer. After initial setup (driver installation) and when using compatible software, moving a piece on the board simply moves the same piece on the screen/application.

The term chess engine refers to a chess playing system that does not have an interface, since it is the “thinking” part of a chess program. A chess engine system has to consider a number of fundamental implementation issues, including: Board representation (how a single position is represented in data structures); Search techniques (how to identify the possible moves and select the most promising ones for further examination); Leaf evaluation (how to evaluate the value of a board position, if no further search will be done from that position); The need for opening books and/or endgame tablebases. Tom Kerrigan’s
Simple Chess Program (TSCP) is a freeware chess engine intended for people who want to learn about chess programming. Its source code is designed to be very easy to understand [12].

2.2 Pervasive Gaming
Pervasive gaming brings computer entertainment back to the real world and overcomes some restrictions of conventional computer games. Players no longer have to be tied to computer screens and human computer interaction is not constrained by graphical user interfaces. Pervasive games have both social and physical aspects that mixture real world entertainment with digital entertainment creating new game experiences. Magerkurth et al [6] categorizes pervasive gaming in five main genres according to the gaming experience: computer augmented tabletop games, affective games, smart toys, location-aware games and augmented reality games.

Computer augmented tabletop games gives us the best of two worlds: the interaction and communication between the players, who sit around the same table, facing each other at an intimate distance; and the computing support that can relieve the players of tasks, such as score keeping or dice rolling, and can enhance the game with visual and audio effects.

Computers are becoming more ubiquitous, and users’ requirements include design for engagement, enjoyment, fun and playability, as well as usability. This can be accomplished by having technologies that benefit from emotional design [13]. Capturing how a player is feeling at any given moment and integrating this very personal representation of context into a game is the goal of affective gaming. Affective technologies are still far to approach human abilities to detect and respond to emotional expressions. Relatively few studies have therefore attempted to study user interactions with affective systems that respond to human emotion [14].

The shape of a toy suggests the way it should be played, but contrary to games, they are not bound by rules or limitations [15]. If we apply Pervasive Gaming to toys preserving their usual ease of use, it is a perfect way to extract principles of user interaction, which allows learning new gaming experiences that might emerge. Smart toys take advantages of two worlds by integrating the power of computers and electronic chips with traditional toys.

Location-aware games use the world we live in as a game board, and can make objects and body movements as an integrating part of our digital games. Doing so we are mixing the virtual space with elements of the real world and the player has to play in two worlds and consider his virtual game consequences as well as the consequences of his actions in the real world.

Augmented Reality (AR) is a technology that brings virtual objects into the real world. AR environments provide more enhanced immersion by seamless merging of real and virtual worlds. It also provides realism through interaction with augmented objects [16]. Currently, AR games are more focused on the technology rather than on game design, but the technology is becoming more stable and novel game designs are beginning to appear.

3. CONCEPTUAL MODEL
In this section, we present a model for designing pervasive games of chess. With this model, we intend to put together the advantages of both real world and computerized chess. First, in section 3.1, we depict a conceptual model for a traditional chess game played between two human players with a regular chess board. In section 3.2, we follow by drawing the model of a computer application of a chess game for one player only. Finally, in section 3.3, we merge the two earlier identified models to conceptualize a pervasive chess game.

Since the objective of this model is to design a pervasive game, we will divide all the models described below in widely used domains of pervasive gaming [17]: physical, social and virtual.

Physical Domain – The physical domain lies in the world around us. Gaming elements in this world generally stand for Tangible User Interfaces (TUI). TUIs allow us to physically represent and control the underlying virtual gaming world. In the physical domain everyday life properties can be used in a game. For instance, we can take information from the weather outside our window and use that data to change the virtual world accordingly.

Social Domain – Games with a social domain use for the most part face to face interactions to include elements such as directed speech, gestures and showing or capturing emotions. In social experiences players interact in a natural fashion, therefore making the experience more instinctive and socially rewarding.

Virtual Domain – This is the typically used environment in nowadays gaming. Any game involving a computerized scenario has a virtual domain, where the digital game logic is displayed usually by a Graphical User Interface. The term virtual often means “something that is almost something else”. Virtual is not Real but can display all the qualities of Real and therefore it is widely used in computer gaming where it tries on numerous occasions it tries to mimic the Real.

3.1 Normal Chess Game
In figure 3.1 we can see a conceptual model for a regular chess game between two players. This type of human to human interaction presents game elements from the physical and social domains as integral parts of the gaming experience.

![Figure 1. Conceptual model of a traditional chess game](image)

3.1.1 Social Domain
In a classic chess game between two human players, there is a strong component of social interaction. One player interacts with the other, for example, by asking to take back one move. The opponent chooses to accept or deny that request and may express an emotional reaction. Another example of social interaction in a chess game happens when we try to study our opponent’s “mind” by, for instance, detecting that false signal he just made, representing a terrible gaffe, with the intention of bluffing. Some players think that the priceless look on the face of a defeated opponent is what makes, across the board chess, the game it is.
3.1.2 Physical Domain
The player and his opponent are present in the physical domain because it is where they interact with the chessboard. Since a chessboard is shared by both players, when a player inputs a new move, that move becomes the output of the adversary.

3.2 Computerized Chess Game

![Figure 2. Conceptual model of a traditional chess game](image)

In Figure 2, a conceptual model for most of the computer chess games available today is presented. Although the player could be represented in the physical domain, there is no interaction in that domain, so there is no need to represent it. The virtual domain, on the other hand, is where all the interactions of a computerized chess game take place.

3.2.1 Virtual Domain
Most chess computer games use a mouse driven graphical user interface, providing the user with the control of the pieces in a virtually displayed chess board. The GUI enables the player to input his move into the virtual domain. Afterwards a chess engine analyses the state of the chessboard and computes its move. Finally, the GUI updates the virtual chessboard with the engine’s move, in order to offer output to the user. With the presented model, we are able to play a one player chess game with the assistance of the controls and representation offered by the GUI, combined with the computing power of a chess engine. The term opponent does not appear in this model because even though there is a chess engine that “thinks” about the best move to make, the embodiment of such opponent is not physically or virtually represented.

3.3 Pervasive Chess Game

3.3.1 Advantages of the Previous Models
From the analysis of the two previous models, we conclude that: a traditional chess game has the advantage of providing innumerable social possibilities between a player and an opponent, rewards the player with a physical experience provided by a real chess board where the representation matches the controls. On the other hand, computer chess games give us the opportunity of playing a one player game using a chess engine which allows us to play a chess match any time of the day, any number of times we desire, it also provide users with a large number of tasks that were only possible recurring to a chess instructor or a chess book. Some examples of these tasks are move advices, a large database of played games, video tutorials, chess problems and chess puzzles.

3.3.2 Pervasive Chess Game Model
To bring together the advantages mentioned in the previous subsection into one single model, we show in Figure 3, a model of a pervasive chess game. This model comprises the virtual domain deriving from the computer chess game model, and both the physical and social domains coming from the traditional table top chess game.

3.3.3 Chess Playing Agent / Opponent
This model is possible by developing a social software agent that interacts with a computer augmented table top chess game. A software agent is an artificial software system situated in a computerized environment, which senses that environment and acts on it, over time, achieving its goals. Social agents can interpret human cues or imitate them; such social cues are, for instance, natural language speech or showing emotions via facial expression or gestures. Therefore, to perform such social cues, the agent must be virtually or physically embodied. In our model, the chess playing agent/opponent appears in a dashed line across the virtual and physical domains because it can be physically or virtually embodied. If the opponent is physically embodied it coexists in the physical domain as well, but the agent’s mind is always present in the virtual domain.

3.3.4 Social Domain
The social elements of a classic two player table top chess game can also be achieved through an embodied social agent in a one player computerized chess game. The idea is to maintain the social interaction between the player and the agent and by that simulate the real human to human interaction, described in the social domain of the first model. For instance, while waiting for the opponent’s move, the player can try to read the embodied agent’s “mind”, or even look at its defeated expression when it loses.

3.3.5 Virtual Domain
When comparing the virtual domain of this model to the one of computer chess model, it loses the GUI to display the virtual chessboard since the objective is to have a chess tangible interface in the physical domain to replace it. Although not represented in this model, a GUI might be also included to display a virtually embodied opponent and data such as tips and advices. The chess engine still subsists, but we considered it as a part of the agent’s
mind and have included it in the chess playing agent. The objective of the chess engine is the same, which is to provide a single player mode and all the advantages of computerized chess gaming. The input of the chess engine this time is not provided by the GUI but by the computer augmented chessboard. After the computation, the chess engine’s move is transmitted to the chess playing agent.

3.3.6 Physical Domain
In the direction of maintaining the physical domain of a traditional table top chess game and preserve the virtual possibilities offered by computers, we need one chess tangible user interface to replace the virtual chessboard of computer chess applications with a real chessboard with augmented computation. While using a tangible user interface, we can input the user’s move into the virtual domain simply by moving a piece in the chessboard. After knowing the agent’s move, the agent has to act on the chessboard like a real life opponent. The agent may have the capability of moving its pieces with, for example, a mechanical arm. If so, he plays the move, otherwise if the agent does not have that capability, the player must make the agent’s move.

4. APPLICATION
In the previous section, a conceptual model for a pervasive chess game was presented. Following that model, we implemented a chess game named “iCat, the Chess Player”. In this chapter we will describe the chess game architecture (see Figure 4).

“iCat, the Chess Player” is a game where a user plays a chess game against a embodied social agent. The game is played on an electronic chessboard that automatically detects the user’s moves. This game can begin from the initial chess position or from any valid position (e.g. a chess problem). The user can take hints about the state of the game by analysing iCat’s facial expressions. For example, if iCat is expressing a very happy face it means that it is winning. After each move played, iCat gives feedback about what it “thinks” about that move. If the user played a bad move, iCat reacts with a happy animation. When this happens the user can take back the move and play another one. By repeating this process, chess can be learned by distinguishing the bad from the good moves. Since iCat does not have any mechanism that allows it to move the chess pieces it verbalizes its move to the user who plays it on the chessboard.

Figure 4. iCat chess player architecture

4.1 Pervasive Elements
The pervasive elements in iCat The Chess Player game are the physical components that allow us to bring the social and physical domains to a computerized chess game. To build a pervasive chess game based on the model presented on section 3.3, we had to choose the technology of the pervasive elements, which are the embodiment of the chess playing agent and the chess tangible user interface. In this section, we introduce and describe the technology of the chosen pervasive elements.

4.1.1 Chess Playing Agent Embodiment
The chess playing agent chosen for the implementation had to give the user the ability of understanding what the agent is thinking, which would allow us to simulate the human to human interaction of a traditional two player chess game. The natural manner used to perform these interactions is by body gestures and facial expressions. Given the above, the chess playing agent must have social skills and therefore it must be a social agent. In our conceptual model we state that the chess playing agent could have a physical or a virtual embodiment. In the implementation we had to have both, this happens because we want to test if the users have more fun while playing with a physical opponent than with a virtual one. The selected embodiment for both the physical and virtual opponent was a social robot named iCat, a user-interface robot developed by Phillips Research [19] that is capable of mechanically rendering facial expressions. iCat uses 11 servos and 2 motors, used for controlling individual parts of the iCat body, such as its eyelids and body. A webcam is located on iCat’s nose and it can be used for computer vision algorithms, such as recognizing faces. Located on the robot’s front there is a loudspeaker controlled by an internal soundcard device. The speaker is used for speech and sounds. With all these capabilities iCat can be considered a social robot since it has many of the characteristics needed to simulate human-to-human interaction. Therefore it was a good choice to embody our social chess playing agent.

4.1.2 Chess Tangible User Interface
For our chess tangible user interface we have selected the only programmable commercial tangible user interface available on the market, the DGT Electronic Chessboard from DGT Projects. As we have related before, there are few chess tangible user interfaces that possess a physical mechanism useful to move its chess pieces. DGT electronic chessboard doesn’t have that mechanism. By using this board the user has to play the computer’s move on the chessboard. Therefore, the user has to know the computer’s move to play it, and for that, we have used the speech and gestures provided from the chess playing agent. However, after some tests with users, we have concluded that they sometimes needed to hear the computer’s move more than one time. Consequently, we did not want to annoy the users by constantly repeating the computer’s move. To address this situation we came up with the solution of connecting a DGT XL clock to the chessboard, using it to display the computer’s last move.

4.2 iCat the Chess Player Module
The OPPR (Open Platform for Personal Robotics) software development environment was created by Phillips to provide programmers with easy development of building blocks used for creating applications for social robots like the iCat. iCat the chess player module is a time-based execution module created to run on the OPPR system. This module comprises the “mind” of our game, a state-driven social agent that plays chess with the user via an electronic chessboard with the goal of winning the game. The
agent also engages users by expressing its affective state during the game through its embodiment. Being a state-driven agent, its “mind” is controlled by a finite state machine. The state machine controls three components (see Figure 4): the chess engine, emotion system and the electronic board interface.

4.2.1 Integration with a Chess Engine
To build a one player computerized chess game, and to use the board evaluation value in the emotion system, we need to integrate our application with a chess engine. In our application we did not have the need of a very strong virtual chess opponent. The focus was to add the social and physical domains to the game. Therefore, the engine did not have to play at grandmaster level and we integrated Tom Kerrigan’s Simple Chess engine with our application.

4.2.2 Emotion System
The emotion system is triggered after the user plays his move. It receives the board evaluation from the chess engine and after processing those values the state of the agent is updated. The affective state will then be reflected in the embodied agent’s behaviour, which is controlled by the OPPR animation module. The complete design and implementation of this system is described in Leite’s work [20]. In this emotion system, the agent has two main components: emotional reactions (computed after every user’s move) and mood (longer lasting affective state controlled by a valence variable).

4.2.3 State Machine of iCat the Chess Player
Finite state machines (FSMs) have been for many years the AI coder’s instrument of choice to imbue a game agent with the illusion of intelligence [21]. In this section we will depict the state machine of our agent (Figure 5).

![Figure 5. Game state machine on the iCat’s perspective](image)

4.2.3.1 Begin State
In this state iCat plays a welcome animation. In this animation, the iCat “wakes up” and says the “Let’s play” sentence to the user. From begin state, iCat, can transit to two other states. In chess whoever has the white pieces is the first to move, so if iCat has the white pieces it moves to the thinking state. If on the other hand iCat is destined to have the black pieces it has first to wait for the user’s move and therefore transit to the Waiting for User’s move state.

4.2.3.2 Thinking State
This state uses the chess engine to choose the best move for iCat and to evaluate the value of the current board position. The current board’s evaluation is the value returned, after a depth search. When the move is chosen we pass along to the next state, reacting, where the returned value and the chosen move are used.

4.2.3.3 Reacting State
This state uses the emotion system to react to the user’s move by changing iCat’s affective state. We have created several idle animations that are randomly used to simulate iCat looking at the chess board thinking about what move to make. These animations happen after the iCat’s animation of the corresponding emotional reaction finishes playing and provide a more continuous and believable affective feedback to the user over time. The final step in order to transit to the next state is to verbalize the previously chosen move. This is achieved by communicating text to speech sentences into the animation module. After saying its move a transition to the waiting for my move state is reached. Also in this state the user take back her last move.

4.2.3.4 Waiting for My Move State
Waiting for my move state presents a time waiting period until the user plays iCat’s move on the chessboard. In this time waiting period iCat is animated by idle animations. From this state we can jump to two other states. The most usual jump is to the Waiting for User’s Move State. This jump happens when the user plays iCat’s move correctly. Another possible transition is if the iCat’s move is a move that ends the game. If so, the current state of iCat state machine transits to the Game Over State. If the user does not play the verbally requested move, iCat warns her that she has played incorrectly. This warning appears with the same probability in two shapes, verbal sentences and iCat animations. If the warning is via verbal sentences iCat randomly chooses one of several predetermined sentences, “That was not my move” is one of the possible sentences. The warning can also come from choosing one of many animations that indicate that the move was incorrect. A sample animation can be nodding the head sideways. Otherwise, if the user plays the move correctly, other animations and sentences were created, in order to give her indication that she is in the right way.

4.2.3.5 Waiting for User’s Move State
The time waiting period present in this state is animated with the same animations of the previous one. Likewise the sentences and animations for illegal and legal moves are the same of the earlier described state. From this state we can only transit to the thinking state. This happens after the user thinks and makes her move.

4.2.3.6 Game Over State
When we reach this state three possible situations might had happened. For all the three different scenarios, losing, winning or drawing, a animation was created. One of these animations plays correspondingly after verifying if iCat has just lost, won or drawn. Finally iCat plays a final animation where it falls asleep, making it clear that the interaction has ended.
5. EVALUATION

This section describes the preliminary experiment we performed in order to evaluate our implementation of our chess game, in both its physical and virtual embodiments. This experiment consists of testing our hypothesis by measuring the user’s enjoyment while varying between a physical and a virtual setup of our game in order to answer our proposed research question. The following sub-sections will discuss the methodology involved in our experiment and the obtained results.

5.1 Methodology

A preliminary experiment has been performed. In that experiment, the participants had the opportunity to play a chess game against the iCat (physical or virtual) using a DGT electronic chessboard.

5.1.1 Measurements

Each participant played chess in our application against either a physical or a virtual opponent. Our independent variable was the type of test (physical or virtual). Our dependent variable is user enjoyment and to measure it we followed the design of an independent test where each participant only played against one of the two possible embodiments.

5.1.2 Gameflow

To evaluate our dependent variable, the user enjoyment, we have chosen to use a model designed to evaluate user enjoyment in games, the Gameflow [12] model. Gameflow is based in the flow theory analysed by Csikszentmihalyi, he affirmed that “Flow experiences can be divided in eight elements and a combination of these elements provokes a sensation of deep enjoyment that people, just for the sake of feeling it, spend a great deal of energy and time” [13]. The flow model can be used to evaluate user enjoyment in any task, but since we want to measure user enjoyment specifically in a game we used the adaptation constructed for games, the Game Flow model. This model maps the eight elements from flow to elements of the game literature and concludes that to have flow (user enjoyment) in a game the elements that must be present are: concentration (Games should require concentration and the player should be able to concentrate on the game); challenge (Games should be sufficiently challenging and match the player’s skill level); player Skills (Games must support player skill development and mastery); control (Players should feel a sense of control over their actions in the game); clear Goals (Games should provide the player with clear goals at appropriate times); feedback (Players must receive appropriate feedback at appropriate times); immersion (Players should experience deep but effortless involvement in the game); social Interaction (Games should support and create opportunities for social interaction).

To evaluate each one of these elements we have used and extended some criteria formulated in the Game Flow model. Clear goals element was removed because the only existing goal that exists in our game is very clear which is to win a chess game against iCat. Similarly to the Game Flow model each question is a five point evaluation Likert scale where the user may choose to strongly disagree, disagree, neither agrees or disagrees, agree or strongly agree. User enjoyment is calculated by a simple mean of the seven element’s value. These elements can also be interpreted as dependent variables of our experiment.

5.1.3 Participants

There was a total of 18 participants, all of them knew how to play chess and had some experience with computerized chess. Since a regular chess game without time restrictions usually takes up to 2 hours or more to play, we have chosen to test users in a predefined more advanced position. This position gave some advantage to the user and was designed with the intention of entering early in the end part of the game. We tested our application in 2 different locations. The first two sessions of testing occurred on a chess club, “Clube de Xadrez de Sintra”, where we had 5 participants with the target age from 8 to 12 years old. The second location was in our university, “Instituto Superior Técnico - Taguspark”, where we tested the remaining 10 participants with subjects of ages between 19 and 32.

5.1.4 Setting

The experiment was conducted in the two different scenarios, physical and virtual. The participants who played in the first scenario did not play on the second one and vice versa, because they would have played the same position twice and therefore harm the results of the second experiment. After each game against our agent, the user filled up the 21 evaluation the electronic chessboard and watched over iCat’s behaviour and its reactions to the same moves.

Physical Setup

In this set up (Figure 2) the participants sat on a chair in front of a table that contained the DGT electronic chessboard, DGT XL clock and the Phillips iCat. Even though the player did not interact with it, the table also contained a laptop in order to run our application and connect to both the iCat robot and electronic chessboard. In this scenario the participants played their moves on the electronic chessboard and watched over iCat’s behaviour and its reactions to the same moves.

Virtual Setup

The virtual setup (Figure 3) differs from the physical one because in this case the iCat is virtually embodied and displayed on a 17 inch TFT computer full screen.

![Figure 6. Physical setup](image-url)
5.2 Results

In this section we present the results of our preliminary testing sessions. We have used non parametric tests because we did not obtain a normal distribution with our test samples.

5.2.1 User Enjoyment – Virtual versus Physical

To study the differences of the user enjoyment between the physical and virtual embodiment of the iCat, we have used the Mann-Whitney U Test in order to analyse the data provided from the compiled data. First we ran the Mann-Whitney U Test using the Type of Test as the grouping variable and the User Enjoyment as the test Variable.

Table 1. User Enjoyment Mann-Whitney U Test Ranks

<table>
<thead>
<tr>
<th>Type of Test</th>
<th>N</th>
<th>Mean Rank</th>
<th>Sum Of Ranks</th>
</tr>
</thead>
<tbody>
<tr>
<td>Physical</td>
<td>9</td>
<td>12,1</td>
<td>108,5</td>
</tr>
<tr>
<td>Virtual</td>
<td>9</td>
<td>6,94</td>
<td>62,5</td>
</tr>
<tr>
<td>Total</td>
<td>18</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

By analysing Table 1 mean rank and sum of ranks we can conclude that the direction of the difference in user enjoyment is physical towards virtual.

Table 2. User Enjoyment Mann-Whitney U Test Statistics

<table>
<thead>
<tr>
<th>Item</th>
<th>Z-Score</th>
<th>p-Value</th>
<th>Direction</th>
</tr>
</thead>
<tbody>
<tr>
<td>User Enjoyment</td>
<td>-2,035</td>
<td>0,042</td>
<td>Physical &gt; Virtual</td>
</tr>
</tbody>
</table>

By analysing each of the Pervasive Game Flow model elements individually we have also concluded that the setup involving the physical embodied agent had a more immersive user experience, improved game feedback and more believable social interaction.

6. CONCLUSIONS

This research enabled us to conceptualize a model for a computerized pervasive chess game by joining the social and physical domains present in the model of a traditional chess game, played between two human players in a regular chess board, with the virtual domain of a single one player computer chess game. After the conceptualization of this model, we have implemented an application under the name of “iCat, the Chess Player” with two different setups. The first setup comprised the iCat robot as the physical embodiment for our chess playing agent, and the other one incorporated only the virtual representation of the iCat at a computer screen. By developing these two setups we were able to perform an experiment to answer our research question, which was: Do people have more fun using the version of our implementation with a more pervasive (i.e. physically embodied) opponent? Or have they higher enjoyment while playing against a less pervasive one (i.e. virtually embodied)? The results of the experiment suggest that our initial hypothesis was correct, since by varying between the two experimented setups we were able to take conclusions about the more enjoyable scenario. Those conclusions showed that participants who have tested the more pervasive physically embodied agent had higher enjoyment experiences than those who have tested with the virtual embodied chess playing agent. By analysing each of the Pervasive Game Flow model elements individually we have also concluded that the setup involving the physical embodied agent had a more immersive user experience, improved game feedback and more believable social interaction.

7. ACKNOWLEDGMENTS

We would like to thank DGT projects for contributing to this research project by providing us with their outstanding DGT electronic chessboard and DGT XL chess clock. We would also like to thank deeply all the members of the GAIPS group.

8. REFERENCES

20. Leite, I., iCat The Chess Tutor, in GAIPS. 2007, IST.