Physically Interactive Robogames: Definition and Design Guidelines

Diego Martinoia, Daniele Calandriello, Andrea Bonarini*

AI and Robotics Lab
Department of Electronics, Information, and Bioengineering - Politecnico di Milano
Piazza Leonardo da Vinci, 32 – 20133 Milan – ITALY

Abstract

There is evidence that people expects to be able to play games with autonomous robots, so that robogames could be one of the next killer applications for Robotics. Physically Interactive RoboGames (PIRG) is a new application field where autonomous robots are involved in games requiring physical interaction with people. Since research in this field is moving its first steps, definitions and design guidelines are still largely missing.
n this paper, a definition for PIRG is proposed, together with guidelines for their design. Physically Interactive, Competitive RoboGames (PICoRG) are also introduced. They are a particular kind of PIRG where human players are involved in a challenging, highly interactive and competitive game activity with autonomous robots.
The development process of a PICoRG, *Jedi Trainer*, is presented to show a practical application of the proposed guidelines. The game has been successfully played in different unstructured environments, by general public; feedback is reported and analysed.

Keywords: Human-Robot Interaction, Autonomous Robot Design, Robogame, Robot Entertainment, Robotic Game

*Phone: +39 02 23993525 – Fax: : +39 02 23993411
Email addresses: diego.martinoia@mail.polimi.it (Diego Martinoia), daniele.calandriello@mail.polimi.it (Daniele Calandriello), andrea.bonarini@polimi.it (Andrea Bonarini)

1. Introduction

This paper presents the definition and design guidelines for a new application field for autonomous robots: Physically Interactive RoboGames (PIRG). The research activity on this topic was started to explore a possible application for autonomous robots that could reach a potential market in a relatively short time, so to contribute to make one more step toward the realization of the Bill Gates prophecy [1] that autonomous robots will be the next technological object in our home, after PCs and cell phones.

One of the most interesting technological mass markets is that of entertainment, in particular video games. In the last years, video game companies have focused on changing the basic paradigm that involved a player sit in front of a screen, manually operating a device; they have started to make the player actively moving in front of the screen, breaking the "fourth wall" and interacting with the game in a three-dimensional space. A step forward is immersive virtual reality game, where a player is plunged in an artificial world reproduced for its senses by ad-hoc devices [2]. Although this enables to play in impressive artificial worlds, it requires to wear special devices, to "produce the (virtual) reality" needed to play, which, at the current state of the art, limit the movement possibilities and quality.

The natural evolution of this process brings to the elimination of screen and devices, and will give human players the possibility to interact directly with autonomous, physical objects in their home and unstructured or dedicated environments (e.g., laser games settings). These games do not need to "produce a reality", since they exploit the real world as environment and real, physical, autonomous devices as game companions. Technology is now mature to support the development of autonomous robots that could play with people in their common life environments, and match at the same time the market requirements. Robogames can be one of the next robotic products for the technological market. To achieve this goal, the development of robogames should be supported by a sound methodology aimed at producing successful systems in a time as short as possible. The guidelines provided in this paper are a first contribution to point out issues, and to stimulate technological and scientific efforts in this direction.

The term “robogame” is commonly related to four main concepts, as it can be checked by a web search.

The first one concerns video games where robot characters are simulated. Here, robots provide a context to the game, and allow to escape all the limitations and problems of physical robots: the robots in the game are possibly autonomous, but they operate in a virtual world completely known. The success of such games supports the fact that robots are expected to be game companions in the collective imagination; the game design technology developed for this type of games can support the development of games with real robots.

Another main stream concerns tele-operated physical robots, where the ability of the player is mainly in the manipulation of remote controllers similar to the ones used in video games, or, eventually, in the design and implementation of tele-operated robots, as it happens in RoboWars [3].

A third implementation of the idea of robogame concerns robots that have been developed by roboticists to autonomously play games (e.g., RoboCup [4]). Here, the accent is on the ability to program robots to be autonomous, but little effort is spent in the eventual, playful interaction with people, often explicitly avoided, as it is in most of the RoboCup leagues.

The last set of robogames concerns robots presented as games, which, in most cases, act more or less as mobile pets (e.g., [5, 6]). In this case, interaction is often limited to almost static positions, not exploiting rich movement, nor high autonomy; the credibility of these toys to really engage lively people, such as kids, cannot be high [7]. A notable exception is Kiro [8], a robot able to successfully play table soccer even with experienced users: the movement is limited to the control of the table bars, and the game is the robotic version of a classical bar game of the past years. Here, the interaction is relatively simple, in a very structured situation, but it matches exactly the user's expectations.

1.1. The PIRG definition

Given this great variety for possible interpretations of the term "robogame", a definition is needed to describe what this paper is focused on: *"A Physically Interactive RoboGame consists of a number of autonomous agents (including software, hardware, and physical agents) of which at least one is an autonomous robot, and one is a person. These agents interact with each other in a possibly variable and unknown environment, by following some game rules, so that the human players can have fun."*

From this definition, it is possible to notice that the focus is on a specific type of game where a physical interaction between people and autonomous robots is envisioned. Although games with tele-operated robots are possible, and are common practice since many years, in this paper the focus is on games where at least one robot is autonomous, so to exploit the possibility to engage with an entity which is not directly controlled by human playmates. This introduces a new way of playing that can stimulate research activity to make it possible.

In a PIRG at least two physical agents are engaged in the game. They can be robots (at least one), software, persons (at least one), or even pets. The important aspect is that they have to be "autonomous agents", i.e., their participation in the game must be active.

The interaction between human players and robotic agents is one of the subtle aspects of PIRGs: they are games that create some sort of interaction, either competitive or cooperative, among users and robotic agents, in particular a physical, and, possibly, lively interaction: the agents should move, exchange signals, participate in interactive actions.

The environment where the game takes place can change and it is in principle unknown, as most of the users' homes are for a robot bought from the shelf, and pulled out of its box. "Variable" has a double meaning: the environment can change during the game, or the game can take place in many different environmental conditions: you can imagine a game arena whose shape and elements change during the game, or where, for example, light conditions can dramatically change.

Finally, the most important aspect: the purpose of a PIRG is to make people have a pleasant experience. PIRGs can be finalized to pure entertainment, but the above definition also applies to therapeutic or serious games, where pleasure is meant as a main driver to engage people and to get them to the therapeutic or didactic goal. This type of experience should be provided to the largest number of users, so adaptation to the specific player's skills is of primary importance.

PIRG development is an interesting research field, since it is characterized by many challenging and still open problems, such as: autonomy in an unknown and dynamic real world, environmental robustness, ease of use by unskilled users, physical robustness, reliability, safety, high–quality interaction. Moreover, a correct cost of the final product is an essential feature that has to be kept into account from the early design phases to target the market. PIRGs can be seen as an extension of video games, where an autonomous robot takes the place of a console. A target price comparable to that of a video game console can be considered as reasonable; this is a driver for PIRG design (and related research), and makes it even more challenging and interesting.

Designing a PIRG is a challenging activity, still largely left to craftspeople. It has not to do only with the technical design issues of a robotic application, but it has to consider also other important aspects, including playability and usability of the game. Insights on these topics have been developed in the video game community. In the last years, some heuristic guidelines to support the design of successful video games have been proposed (e.g., [9, 10, 11, 12, 13]), as well as Human-Robot Interaction (HRI) principles (e.g., [14]): the guidelines presented in this paper take inspiration from them, as well as from standard software engineering and robotics design principles. Some interest is also rising about using robogames as a tool to investigate HRI issues [15]: although this is not the focus of this paper, an important part of the envisioned design process is dedicated to HRI.

In this paper, Physically Interactive, Competitive RoboGames (PICoRG) is also introduced; this is a sub-field of the more general PIRGs, characterized by situations where players are involved in a challenging and physically interactive, competitive game activity with autonomous robots. The competition issue introduces characteristic aspects that will be discussed in the following sections.

In the next section, guidelines for the design of PIRGs are introduced and discussed. In the following, how they have supported the development of the "Jedi Trainer" PICoRG is reported.

2. General Guidelines

In this section, some guidelines for the design of PIRGs are defined, based on the experience accumulated in about two years of PIRG design that have produced more than ten different games, involving very different robots, and aimed at different target users. The guidelines include common sense principles, and principles from other disciplines, specifically re-tuned for this application. In Appendix A is reported an example of another game developed before the definition of the proposed guidelines, to show how these can actually support an effective design.

The design process develops from a first *conceptual description* of the game, expressed by an informal, written text, through a first phase dedicated to the design of the functional aspects of the game, and producing what is called a *functional description*, then to the interaction design phase that produces the *interaction description*, and, finally, the implementation and test phases, where the game is actually realized in all details, and tested with a sample of the final users.

The guidelines are proposed in the rest of this section.

Consider the PIRG definition

The PIRG definition given above includes all the elements that should be considered to identify requirements and constraints. The identification of the environment characteristics and of the agents' roles in the game, and the corresponding required skills are fundamental. In particular, the skills required from the autonomous robot to play a meaningful role in the game should be identified, and the specifications should be possibly adapted to the actual possibilities of the robots that can match the other specifications and constraints.

Define a story for the game and stick to it

A story that justifies goals, roles, and actions of the agents has to be defined. Goals should be declared at the beginning, as well as the actions that all the agents can do. The plot should be simple enough to be easily understood, but at the same time it should stimulate the human players to take seriously their role. This should also include some challenge for the physical and/or cognitive abilities of the players, compatible with their expected skills. Each action and behavior for each agent should be integrated in the story. The story should contain elements familiar to the players.

R^3, or Reduce, Reuse, Recycle

The assumptions considered for PIRGs require to face the design process by looking for solutions to problems (i.e., ways to implement functionalities), while satisfying the market constraints. For example, potential solutions including sensors like a Laser Range Finder cannot be considered at the moment, since there is no way to include such a device at a cost compatible with a commercial robogame. The problem solution is highly constrained: most of the time, COTS (Components Off The Shelf) will have to be used (possibly in a smart way), rather than developing customized, and, possibly, more effective and reliable components. This approach is aimed at the development of market-oriented applications in a relatively short time; this changes completely the robot design approach often adopted in more basic-research-oriented projects. The story of technology is plenty of components that have lowered their price when reaching the mass market, but many more are the components that have been kicked-off from the mass market (or never reached it) because of their lack of cost-effectiveness.

Reduce. In order to keep the cost low, the performance of the product might have to be reduced to the minimum compatible with the enjoyability of the game, rather than looking for a perfect solution at any cost. This might be quite unnatural for a researcher, who usually strive for the best solution for the problem, but it is normal practice for products. It may happen also that some functionality of the game has to be changed to match the features of admissible sensors.

Reuse. It is extremely important to find new ways to reduce the cost of materials and components, even using items that are a priori not related to robotics, as it has been already done in robotic applications with smart phones (e.g., [16, 17]), video game controllers (e.g., [18, 19]), computer mice (e.g., [20, 21]), and many other COTS.

Recycle. Sensors might be quite expensive. It is important to try to get the maximum amount of information out of the data we can obtain from the selected sensors. Sometimes, it is possible to get information by processing data that we already have, rather than getting richer information by adding other (or more expensive) sensors.

Play it safely

It is very important to put in the design process extra care about safety requirements, trying to foresee all the potential problems, and possibly providing automatic solutions for them. The norms ruling the use of autonomous robots in generic environments are still under discussion [22], and, in the meanwhile, safety is up to the designers. This might also affect the conception itself of the game. For instance, it should be avoided to ask a human player to run after a mobile robot launched at high speed, even if this might be a relevant feature to make the game interesting.

KISS: Keep It Stupid Simple

PIRGs may be targeted to children and other (at least initially) unskilled and untrained users. This must be considered not only when designing the interaction process, but also for the maintenance procedures (if any) or the pre/post game set-up operations. It should be possible to understand how to play the game with a minimal instruction, eventually provided by the system itself. All the steps of the game and the states of the players should be clear at any time, and dedicated, reliable communication channels have to be set. Obscure error messages have to be avoided. The user manual, which is expected to be read, should provide detailed explanations and include non-technical examples and similarities, but the game should be understood even by only reading a summary of few lines, as it usually happens for video games.

Test the assumptions

No matter how well-founded and reliable the taken assumptions are considered, they have always to be tested by an appropriate user pool, in the real environment. For instance, it may be possible to assume that the interaction signals are easy enough to be produced by all the users, but to understand only in the test phase that they require skills that may be developed by playing the game, but that may not be assumed a-priori to be shared by the potential users.

Do it for the fun

The final goal of a PIRG is to make the user enjoy it and have a good time. The design of a PIRG is driven in this direction and actual achievement of this goal has to be tested on the prototypes and the final product. Enjoyment is a complex mechanism that involves challenge, self-motivation, self-realization (e.g., [23, 24]), and should be reached by a large variety of people playing the same game. It may require a mix of cognitive and physical involvement, challenge, reward. Often, it is needed to have the game adapting to the skills of the specific user, or, at least, matching a broad set of skills.

Robot as a rational agent

The robot(s) participating in the game have to be perceived as rational agents, possibly smart enough to play the role of effective opponents or teammates. This rises the interest in the interaction (and so in the game), since the player tries to set up a relationship with the robot, and to load it with the emotion asked for by the game. Moreover, nobody likes to play with or against a dull playmate. Eventual limitations in robot abilities should be turned into features of the game (e.g., a low speed should appear as need for attention and care in movements). Decisions about actions should be taken rationally. Even randomized actions eventually included in the behavior should be perceived as expression of some intention of the robot, as mentioned in the next section.

Sense exploitation

People usually interact by three main modalities: visual, aural, and kinaesthetic (e.g., [25]). Many of the top–selling video games give the users, in some measure, the possibility of experience the game not just with their eyes, but using more than one sense at a time. PIRGs put the accent on kinaesthetic interaction, by providing an artificial body to interact with, but it is important to remember also the other modalities, which play a major role in traditional video games and are important to deliver a full game experience. For instance, a silent robot might make sense in few games, while in others the user expects to hear, at least, sound expressions from the robot. Gestures, lights, and proper movements may complete the interaction kit.

Agile Implementation

The robotic design process is usually characterized by a very well defined step-by-step approach, where a complete analysis of user requirements is followed by careful and detailed planning, followed by implementation. However, this approach may be inappropriate to design PIRGs, for many reasons: first of all, given that we are considering the development of a new game, there is usually no user that have experienced it and that can be interviewed to get requirements at the beginning of the design process; the designer has to guess how users might like the game. Secondly, the use of low-budget components and COTS can rise problems that may be hard to foresee before the actual implementation. Finally, good add-ons or adjustments can become obvious only when a prototype has been realized and can be tested. Agile implementation means being ready to start all over again at any given step of the development process, whenever a better solution, in terms of costs or user experience, arises.

3. Applying the guidelines to the development of a PIRG: Jedi Trainer

In this section, an application of the above introduced guidelines to develop a real PIRG is reported, together with an example of the development process organized in phases: *project overview*, *functional design*, *interaction design*, *implementation*, and *testing*. In each of them, the name of the applied guidelines is mentioned in italics. The presentation is rather informal, so to maintain the feeling of the development process actually followed.

3.1. Project Overview

Before starting the design process, the idea of the game, its *conceptual description*, has to be drafted (*Define a story for the game and stick to it*). In this example, the inspiration came from a scene of "Episode IV - A New hope", the first movie of the "Star Wars" saga. "Jedi Trainer" is a single human player PICoRG. It recreates the training of Luke Skywalker on board of the Millenium Falcon. A drone flies around the player (Luke as a Jedi Knight trainee, in the movie), and shoots him with its laser blast. The player has to parry the shots with his light–saber. If the shot is parried, one point is assigned to the player, otherwise to the drone. The first one that reaches a given score wins the game. The drone is completely autonomous once the game is started. It should be possible to have different behaviors to implement game experiences at different levels of difficulty. To match the inspiring scene, it should be possible to play the most difficult levels of the game wearing a mask on the eyes, so to exclude any visual contact, and to rely only on the player's sense of hearing (*Sense exploitation*).

3.2. Functional Design

The first step of the design process is to consider the project overview to identify the specific elements of the system and the required functionalities (*Consider the PIRG definition*).

Environment

No environment information is provided in the overview. It was possible to select a controlled environment with stable and known conditions, but, remembering that the ultimate targets are fun and interaction with as less limitations as possible, the only selected constraint was the need for enough open space to have the drone flying around the player at a distance a little bit greater than the distance reachable with the light–saber, as it was on the Millenium Falcon (*Define a story for the game and stick to it* and *Play it safely*). This leaves the players a lot of freedom about the place to play the game, but it also means that almost nothing can be assumed about the environment (indoor/outdoor, light conditions, shape, background, and so on), and the implemented PIRG should be able to play in all these situations. It would be possible to ask the players to input information about the environment at each game start, but this would require that the players define a number of parameters, and it would likely overload them, thus reducing fun (*KISS*). The robot's sensors may gather information about the environment, but technical and budget constraints (R^3) have to be taken into account. The final choice, as described in the next section, was a compromise between all these needs.

Agents

Two agents are playing the game: the drone and the human player. An acceptable constraint is that the human player should be able to stand and manipulate the light–saber.

The drone could be implemented as a flying object, namely a quadricopter. A commercial one, the Parrot ARDrone [16], was selected on the basis of its characteristics and price, comparable with that of a video game console. The choice of a product on the market also gave the benefit of working with a tested and certified device (*Play it safely*).

Once selected the robotic base, the robot design issues had to be faced: low-budget limitations may push the designer to use COTS. This means that the following design steps are bound to the performance of the available components. In our case, the drone sensors include a frontal camera able to transmit QVGA (320x240 pixel images) highly compressed video stream via wireless link , and a battery charge sensor. The battery lifetime (about 15 minutes flight) suggested that the game had to last less than 10 minutes per round. The drone comes with a self–stabilization functionality implemented, and can be driven via WI-FI along 4 degrees of freedom: X,Y,Z axis movement, and rotation around the Z axis.

Functional description

The definition of the environment and agents enabled the definition of some qualitative functional description together with some first technical choices. A draft of the game description at the end of this first phase is reported hereafter.

The player will be free to move all over the available space, wielding the light–saber. The drone will fly around the player, sometimes changing direction, and trying to keep a constant distance.
When the player is not shielded by the light–saber, and the chest of the player is in the center of the drone's camera view, the drone can decide to shoot its laser beam. The player will try to parry the laser beam with its light–saber, and an evaluation of the outcome of this parry stance will assign a point either to the drone or to the player. When one of the opponents reaches the target score the game is over, and the player with the highest score wins the match.

Notice that this definition is still open to many different implementations, and this is typical of this phase, where decisions are taken to be tested ad eventually changed in the following steps.

3.3. Interaction Design

The second phase concerns the design of the human–robot interaction, and the specification of game parameters. Questions such as these had to be answered: "How can the light–saber be implemented?" "How can the player be detected?" "What distance should the drone fly from the player?" "How can such a distance can be estimated?"

The definition produced in the previous phase was analysed step by step (*Define a story for the game and stick to it*) to design the interaction.

Movement

The first sentence states that the player can move freely, always wielding her/his light–saber. According to the game definition, the light–saber should be detected in any situation with the precision needed to evaluate the parry stance. A "real" light–saber (a realistic item found on the market (R^3)) resulted to be too thin for being detected with such a low resolution camera, and its flickering light altered the colour of the player's robe reducing the possibility to recognize it (*Test the assumptions*). Something visible, light, light–saber–looking, and safe had to be identified. Finally, the light–saber was implemented as a swimming noodle, opportunely dressed (to be opaque) and reinforced (R^3). Albeit it might be argued that this is quite far from how a light–saber looks like, it has color and dimensions similar to the real light–saber, and users that tested it enjoyed a lot the solution (*Test the assumptions*), and considered it as a realistic substitute for the real light–saber (*Define a story for the game and stick to it*), which was evident to everybody that it had been impossible to have. This is an important aspect that should not be under–estimated: the game should have a story and reference points that match something known and well present in the player's mind.

The second sentence of the game overview states that the drone has to constantly track the player, and to keep itself at a constant distance, sometimes changing direction. Here, the problems are player recognition and distance estimation.

Distance estimation is particularly hard: since the drone doesn't have any frontal distance sensors, this information has to be derived by the video stream, rather than installing a dedicated sensor (R^3). The player size is not known a priori (*KISS*). To standardize this variety, the player is asked to wear a universal size game uniform. Knowing the real size of the uniform, it is possible to estimate the player's distance from the frontal camera in terms of number of pixels in the image, with a precision good enough for the game purpose. A single-colour uniform shaped as a Jedi robe was adopted (*Define a story for the game and stick to it*), and body detection was implemented by colour detection: the user is identified as the largest blob in the image having the robe colour, and compatible geometric features. The movement of the user makes the robe changing shape, and this gave an interesting effect, put in evidence only during tests: due to the variation of the detected robe blob size the drone moves back and forth, thus adding a new dimension to the a priori designed behavior. This makes the drone appearing even more autonomous (*Robot as a rational agent*), and provides a sort of emotional feedback to the players (*Do it for the fun*): very active players induce a very active movement in the drone, while still players induce a more quiet and less aggressive behavior in the drone. This sort of emotional matching is also an important aspect to build a good relationship [25], and implements a kind of adaptation to the user's mood (*Do it for the fun*). This implementation of the distance estimation may bring to poor performance (in strictly functional terms), especially when light is changing, but the behavior of the drone seems rational also when most of the color detection problems occur, since they make the drone looking as if it was trying to find a way to shot the player, which is fully compatible with its role (*Define a story for the game and stick to it* and *Robot as a rational agent*).

The direction of movement over the virtual circle around the player should change from time to time, so to show the drone challenging the player's stance. The first approach was a random selection of movement direction, but the users didn't like it (*Test the assumptions*), because the drone seemed "too dull": it was not possible to recognize any intentionality in its behaviour. Finally, an algorithm was implemented to select randomly when to take the next direction change over a set of non-uniformly distributed finite time values, obtained after several user-appreciation tests. With this, users tended to perceive the drone's movement as if it was trying to challenge them (*Test the assumptions* and *Robot as a rational agent*).

Some emphasis is put on this aspect. In any kind of robogame, people expect from their robot companions a very broad range of behaviours, but, if the robot shows too many different strategies to achieve its goal, these might be perceived as random choices, and this should be avoided: good companions should be perceived as smart, while random actions are perceived as not purposeful. From another game that we have developed we have learnt that even a quite simple game, such as driving a robot to the back of another, autonomous one, may become interesting when the human player is challenged either cognitively or physically (in that game the remote control was implemented through gestures interpreted by a WIIMote® device). Understanding the behavior of the opponent is a cognitive challenge performed under emotional pressure during the game. If players perceive that they cannot understand how the opponent is moving (either because it is moving randomly, or because the strategy is too complex for the cognitive or perception levels of the player), and this is relevant for their performance, then they may lose interest in the game (*Do it for the fun* and *Robot as a rational agent*). It is responsibility of the designer to balance smartness and complexity, usually testing alternative solutions with a representative set of users (*Test the assumptions*).

Actions

From the analysis of the game overview, the next step was to design how "the drone can decide to shoot" (*Robot as a rational agent*). As mentioned above, the game duration should be no longer than 10 minutes per match, because of the battery life time. Each shot will assign a point either to the drone or to the human player, so there can be at most $(N-1)+N$ shots per match, where N is the final score. Thus, the drone must calibrate its shots so that the average time between them is no longer than $\frac{10}{(2N-1)}$ minutes, possibly shorter. But, again, just having the drone to shoot at random intervals cannot be accepted by the user (*Do it for the fun* and *Robot as a rational agent*). The final algorithm took such average time between shots when the player parry stance was at a normal level of quality. If the player opened up her/his guard, the likelihood of shooting increased, while if the player had a very good parry stance, the drone would more likely refrain to shoot. The quality of the guard stance is evaluated as proportional to the area of a specific part of the uniform (corresponding to the player's chest) covered by the light–saber. Combining this with direction changes gives the users the impression that, when their stance is very good, the drone does not fire and tries to change direction to make them open up (*Robot as a rational agent*). When the battery is almost over, the drone blasts its last shots and lands, so to avoid sudden interrupts due to exhausted batteries. This seems as it was trying its last chance, being not perfect, so being an opponent at a comparable level. The calibration of the robot abilities is also very important, because it keeps the challenge (and motivation) high (*Do it for the fun*).

The next step concerns the definition of how to implement a shot. The drone has no laser beam shooters in its equipment, therefore, a way had to be identified to represent the shot so that it cannot be misinterpreted (*KISS*). A real laser, even a laser pointer, would bring two problems: the weight (this drone cannot mount any overload) and the fact that, on the opposite of what happens in movies, laser beams are not visible in open, dry air (only the dot they produce on the target is). After some brainstorming and testing, it was decided that the drone perform a standard animation just before shooting. This consists of a rather violent shaking, which noise could be easily associated to the laser noise people are used to hear in movies (*Sense exploitation* and *Define a story for the game and stick to it*). This can be used also for the most challenging version of the game, when the players' eyes are shielded, and they have to play blindly. A few moments after the animation, the image taken by the frontal camera is checked: if the light–saber covers the chest, then the point is assigned to the user; if the chest is towards the center of the image (and it is not covered), then the point is assigned to the drone. The time between animation and image checking can be modulated to make the game more or less challenging, but paying attention to leave enough time to the players to react to the sound, in order to make them enjoy the challenge (*Do it for the fun*).

Scoring system

The last part of the interaction design phase concerns the scoring system and, in general, the game interface. Simplicity is the key concept here, and the usage of a personal computer as interface can bring many benefits, such as: effectiveness of communication, user familiarity, ease of implementation, and the possibility to implement changes in a short time (*Agile implementation*). All the commands (take off, land, emergency, start game, stop game) are given via a software interface, which also shows, and clearly announces the score when appropriate. In general, it is recommended to avoid direct interactions between user and robot outside the game roles, since these may reduce the engagement, because the robot loses its role in the story (*Define a story for the game and stick to it*), to take the one of interface or score table (*Robot as a rational agent*). The only exception that might be accepted is when the score affects the behaviour of the robot, e.g., by making it moving more brightly when winning and more slowly, or staggering, when losing; it is not the case in this game, and this possibility was exploited in other games, such as the one presented in appendix.

Interaction description

Combining the information derived from this second design phase, the following game description was obtained.

The player has to wear a standard-size game uniform, recalling the Jedi robe, and wields a light–saber, i.e. a solid, coloured, and light tube arranged to look like a light–saber.
The game starts with the player ready in front of the drone, and the drone on the ground. As the game is started on the computer, the drone takes off. The drone flies around the player, keeping itself at a constant distance. Distance is estimated from the blob size in pixels, knowing the actual robe size. Direction is maintained for a given amount of time, randomly selected over a set of feasible, non uniformly distributed, values.
The drone shots are represented by a unique animation in the drone flight, such as a violent shaking. The shot will occur according to the quality of the player's parry stance.

Parry evaluation occurs by checking the amount of area of a specific part of the robe (i.e., the estimated player's chest) covered by the light–saber. The drone will always aim for the chest. Evaluation occurs a certain number of frames after the animation standing for laser shot, to give the user enough time to move her/his light–saber to parry. Score is assigned accordingly. Score is constantly, clearly visible on the user interface running on an external computer, and announced at each change.
When the time is over, the player with the highest score wins the match.

3.4. Implementation

The above-mentioned description was implemented in a prototype.

In particular, the strategy implemented by the drone was the result of three overlapping kinetic actions, controlled independently by three different software agents: lateral movement, frontal movement, and shooting. Technical details are not in the scope of this paper, but it is interesting to mention that the rules governing the drone are actually quite simple, that all three agents act independently of each other, in a distributed fashion, and that the perceived result is yet that of a coordinated, centralized, intelligent strategy, which results in a funny, challenging and engaging experience.

3.5. Testing

The performance of the game during alpha-testing seemed acceptable: when played by the developers, the game resulted enjoyable and reliable, but as the guidelines say, no matter how much you like something, you need to test it with the users (*Test the assumptions*). Testing is one of the most important phases of the design of a robogame: as already mentioned, the final goal of a robogame is its enjoyability by the users. Unfortunately, there is no absolute scale to measure such a thing, being its perception extremely different from user to user, and also from the same user in different emotional settings. Like for video games, a set of users would test the game, but, due to the difference in the physicality between a robotic game and a virtual one, the users' variety should cover also all the possible spectrum of physical characteristics (e.g., tall, short, fat, slim, male, female). Moreover, as many people with small familiarity with robots and computers as possible was included in this pool, since it was assumed they might have brought a greater variety of reactions to the new game.

The beta-testing pool consisted of a set of both male and female subjects, distributed on different social backgrounds and physical characteristics, ageing between 19 and 23. After the first round of testing, all users seemed to appreciate the game concept, but some problems emerged. The drone was too slow, both as movement and reactivity, and did not appear to challenge enough (*Do it for the fun*). Moreover, the movement range, was too broad, due to the intrinsic imprecision of distance estimate and to unpredictable delays in image transmission. When the drone goes away, the players tend to follow it, and this is acceptable (*Robot as a rational agent*), but when the drone rushes towards the players, they may got scared, an undesired effect in this game, where the drone is expected to play at a given distance from the player (*Robot as a rational agent* and *Define a story for the game and stick to it*). Other problems came from the partitioning of the robe color blob due to shadows produced by some bodies (e.g., some girls or fat persons), solved by joining blobs that satisfy topological and dimensional constraints. Once solved the technical causes originating these problems, and after having tuned parameters such as speed and critical distance, the game was ready to go on the field.

Although the drone could now keep a quite constant distance from the players, they tended to react lively to the drone's movements, and to move freely to face it, according also to personal feelings. This witness the involvement of players in the game, which makes them forgetting anything but the drone (*Do it for the fun*). Therefore, since the space needed for the game would be virtually unbounded, but this is not always possible, recommendation had to be included to the players to try to stay more or less in the middle of the available space.

In Figure 1, a user playing the game in an outdoor environment[1] is shown. In Figure 2, another user, with other feelings and attitude, is playing the game in a room with artificial light.

The game was presented at a major robotics exhibition in Italy, Robotica2011 [26]. Besides demos held by the developers, 35 persons among the general public accepted to try the game, and to provide structured feed-

[1] Videos are available from:
`http://airwiki.ws.dei.polimi.it/index.php/Robogames`

Figure 1: Facing the drone(left) and parrying (right) in outdoor environment. On the lower part, on the right the image from the drone camera, on the left the corresponding colour classification.

Figure 2: Facing the drone (left) and parrying (right) in indoor environment: different game conditions and player.

back. They did not know anything about the game, the developers, and the presenters. Their distribution by age and gender is reported in table 3.5.

Gender\Age	*<18*	*18-23*	*>23*	*Tot*
F	**0** (0%)	**3** (8.6%)	**2** (5.7%)	*5 (14.3%)*
M	**7** (20%)	**19** (54.2%)	**4** (11.4%)	*30 (85.7%)*
Tot	*7 (20%)*	*22 (62.9%)*	*6 (17.1%)*	*35 (100%)*

Table 1: The distribution of the sample by gender and age. Percentages are computed w.r.t. the grand total (35).

Although it was not possible to select the sample according to proper statistical criteria, this distribution is not far from the one expected for potential customers, possibly due to the kind of exhibition where we recruited them. The game was designed to target young people, able to move fast in front of the drone, mainly males, due to the inspiring situation in the movie and the nature of the game. In the exhibition setting there was a lot of space to leave people free of moving according to their feeling, and the length of the game was limited to three minutes, to allow more people to try it.

At the end of each match, players were asked to assign a degree of agreement with proposed assertions, by putting autonomously a sign on a horizontal bar whose extremes were marked as 0 and 10. Among the assertions proposed in the questionnaire, the ones relevant for this paper are reported in Table 3.5

A	"I was wishing to play"
B	"I enjoyed the game"
C	"The game was engaging"
D	"I was focused on the game only"
E	"I was focused on the opponent only"
F	"I improved during the game"
G	"The game was too slow"
H	"The game was too long"

Table 2: The assertions in the questionnaire, and the corresponding labels.

The degrees of agreement have been discretized into 21 labels equally distributed from 0 to 10 (step 0.5); the results are summarized in Figure 3.

The Jedi Trainer game was appreciated by the large majority of the subjects, as shown by the distribution and the particularly high median of the assessment of agreement with assertions B and C. It appears that involvement is definitely high, meaning that this game was able to engage the players. It can be noticed that most of the players had high expectations from the game, as it appears from the high median agreement with question A. Engagement is confirmed by answers to questions D and E, which show that most of the players were concentrating on the game or on what the opponent was doing. Many players also felt to have improved during the game (question F); this is another important result, since it contributes to leave a positive impression of the experience, and to motivate further playing. Figure 4 reports the

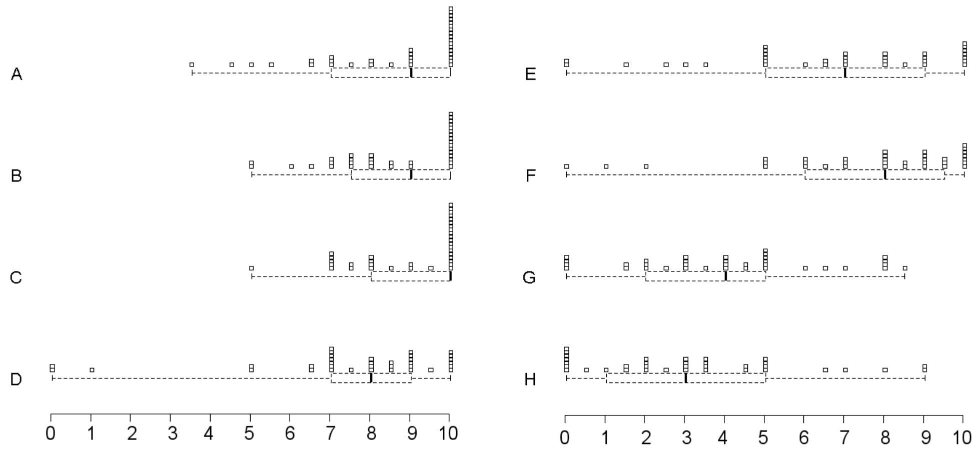

Figure 3: Summary of questionnaire results. For each question, the actual answers are represented by means of a stapled dot plot, where the answer of each subject corresponds to a squared dot; in addition, being the data ordinal, a synthetic representation has been plotted, showing the following modalities: lowest and highest (extremes of the dashed lines), 25% and 75% percentiles (boundaries of the dashed boxes), and median (bold line).

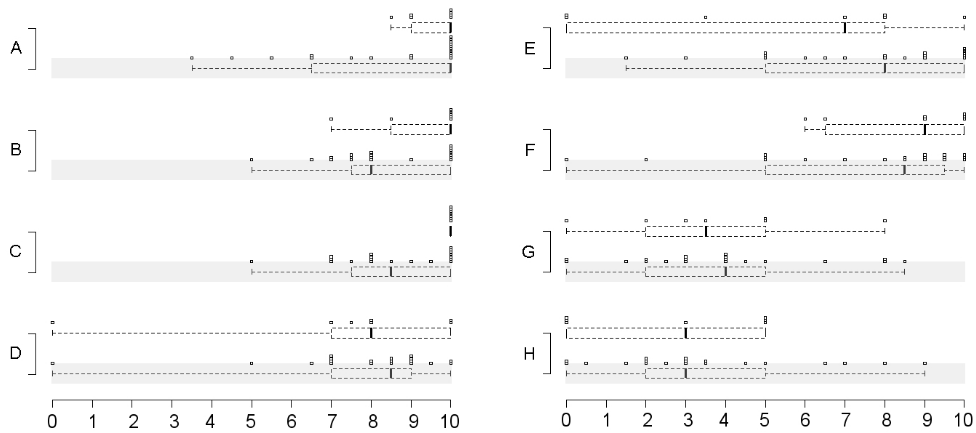

Figure 4: Comparison among the distributions of answers given by males under 18 (no shade), and between 18 and 23 (gray shade). The data are represented as in Figure 3.

agreement with assertions of the two more important sub-groups that represent the core target of the game: males under 18, and between 19 and 23. It seems that a larger percentage of younger people was more involved in the game, while the other answers seem comparable; this reinforces the original assumption about the target population, which included people from kids up to 23 years old with no substantial difference in the game design.

The players were also asked to report emotions felt when playing, as free text answers; among the elicited ones there are: "enjoyment", "happiness", "involvement", "anxiety", "tension", and "satisfaction". These emotions do not seem to be related to the other answers. For instance, it is interesting to notice that players expressing a similar agreement with the assertions about engagement and enjoyment declared to have felt different emotions. This puts in evidence, once more, that the game experience is unique for each player, and a game has to trigger the emotions that make the highest number of potential customers to enjoy it, goal apparently achieved by Jedi Trainer.

4. Conclusion

This paper presents a set of guidelines to design Physically Interactive RoboGames (PIRG), and introduces as well a first category of robogames: Physically Interactive, Competitive RoboGames (PICoRG). The application of the proposed guidelines to the development of a successful PICoRG (Jedi Trainer) is reported, to show how they can effectively support the definition of such challenging robotic applications, where low–cost, autonomous robots should show top level capabilities in interacting with people in competitive games. In particular, by following the proposed guidelines, it is possible to implement autonomous robotic systems able to show intelligent behavior even if the equipment on board is limited by costs and technology reliability.

Many issues are related to setting and maintenance of effective communications channels between the player and the robot. These issues can be faced by a careful design of the game and the interaction modalities, which can make use of different interaction channels. The user can be engaged thanks to the appropriate design of the game, which takes into account both the physical and cognitive aspects of interaction, together with the devices that can be exploited, given the application constraints. In the game context, any apparently rational behavior of the robot may help to accept it as a robotic companion. This set of guidelines, albeit simple, expandable and amendable, is already a valuable tool to support the implementation of simple, inexpensive, yet smart robots that can be well accepted as game companions.

The interdisciplinary activity that supports this research line is continuing. Other games are under development, both to evaluate the generality of the proposed guidelines, and, possibly, adapt them to other contexts. In these new games, new situations and robots, as well as new abilities and kind of games are exploited.

More complex cognitive abilities in robogame robots are under development, for instance, the ability of the robot to detect a place where it is possible to hide, which requires a good integration between (low–cost) sensing and reasoning: the conceptual definition itself of the features of such a place is problematic. Once this ability will reliably work, it will make possible the implementation of games like hide–and–seek, and sophisticated shooter games, where surprising actions and hunting will be enabled. Moreover, guidelines for co–design of robots and games, where a game idea can influence also the design of the robot base and its capabilities, are under development. For instance, a fast robot on omni–wheels, might engage the human player in real chasing and pursuit, opening the way to new games. This will also introduce new emphasis on issues such as safety for the robot and the environment; for instance, the player should not be induced to run after a robot at 3 m/sec, so losing the perception of potentially dangerous corners or slippery floors.

A final development direction concerns multi-player games, where more than one human player and/or robot are involved. This increases the playing possibilities, and includes in the game issues related to roles, dynamic role exchanging, teamwork, and much more.

5. ACKNOWLEDGMENTS

The authors thank the organizations that have made the game tests possible (Artenergy and Fieramilano), Roberta Carabalona for data analysis, and the authors of the game reported in the appendix, Christian Mandelli and Deborah Zamponi, who did a great job at that time.

References

[1] B. Gates, A robot in every home, Scientific American Magazine 296 (1) (2007) 58–65.

[2] M. Zyda, From visual simulation to virtual reality to games, Computer 38 (9) (2005) 25–32.

[3] M. Thorpe, Robowars, `http://www.robowars.org` (2011).

[4] M. M. Veloso, Entertainment robotics, Communications of ACM 45 (3) (2002) 59–63.

[5] M. Fujita, K. Kageyama, An open architecture for robot entertainment, in: AGENTS '97: Proceedings of the first international conference on Autonomous agents, ACM, New York, NY, USA, 1997, pp. 435–442.

[6] T. Shibata, K. Inoue, R. Irie, Emotional robot for intelligent system-artificial emotional creature project, in: Proceedings of 5th IEEE International Workshop on Robot and Human Communication, IEEE Computer Press, Piscataway, NJ, USA, 1996, pp. 466–471.

[7] D. Engber, Rrrrrrrobots! The search for the ideal animatronic pet, `http://www.slate.com/id/2206604/?y=` (2008).

[8] T. Weigel, Kiro–a table soccer robot ready for the market, in: ICRA 2005 – Proceedings of the 2005 IEEE International Conference on Robotics and Automation, IEEE computer Press, Piscataway, NJ, USA, 2005, pp. 4266–4271.

[9] H. Desurvire, C. Wiberg, Game usability heuristics (PLAY) for evaluating and designing better games: The next iteration, in: Online Communities and Social Computing, Vol. Lecture Notes in Computer Science 5621, Springer-Verlag, Berlin, Heidelberg, 2009, pp. 557–566.

[10] K. Jegers, Investigating the applicability of usability and playability heuristics for evaluation of pervasive games, in: ICIW '08: Proceedings of the 2008 Third International Conference on Internet and Web Applications and Services, IEEE Computer Society, Washington, DC, USA, 2008, pp. 656–661.

[11] D. Pinelle, N. Wong, T. Stach, C. Gutwin, Usability heuristics for networked multiplayer games, in: GROUP '09: Proceedings of the ACM 2009 international conference on Supporting group work, ACM, New York, NY, USA, 2009, pp. 169–178.

[12] J. Schell, The art of game design: a book of lenses, Elsevier/Morgan Kaufmann Publishers, Burlington, MA, 2008.

[13] Game design workshop, Elsevier/Morgan Kaufmann Publishers, Burlington, MA, 2008.

[14] T. Fong, I. Nourbakhsh, D. Kerstin, A survey of socially interactive robots, Robotics and Autonomous Systems 42 (2003) 143–166.

[15] M. Xin, E. Sharlin, Playing games with robots - A method for Evaluating Human-Robot Interaction, Itech Education and Publishing, Vienna, A, 2007.

[16] P.-J. Bristeau, F. Callou, D. Vissire, N. Petit, The navigation and control technology inside the AR.Drone micro UAV, in: Proceedings of the 18th IFAC World Congress, Elsevier, Amsterdam, NL, 2011, pp. 1477–1488.

[17] Orbotics, Sphero, the iphone controlled ball, `http://gosphero.com/` (2011).

[18] M. Lapping-Carr, O. Jenkins, D. Grollman, J. Schwertfeger, T. Hinkle, Wiimote interfaces for lifelong robot learning, in: Proceedings of AAAI Symposium on using AI to Motivate Greater Participation in Computer Science, AAAI, 2008.

[19] C. Smith, H. Christensen, Wiimote robot control using human motion models, in: Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems. IROS2009., IEEE Computer press, Piscataway, NJ, 2009, pp. 5509–5515.

[20] A. Bonarini, M. Matteucci, M. Restelli, Automatic error detection and reduction for an odometric sensor based on two optical mice, in: Proceedings of the 2005 IEEE International Conference on Robotics and Automation. ICRA 2005., IEEE Computer press, Piscataway, NJ, 2005, pp. 1675–1680.

[21] M. Dille, B. Grocholsky, S. Singh, Outdoor downward-facing optical flow odometry with commodity sensors, in: Field and Service Robotics, Springer Verlag, Berlin, D, 2010, pp. 183–193.

[22] TC184-SC2, Robots and robotic devices, http://www.iso.org/iso/iso_catalogue/catalogue_tc/catalogue_tc_browse.htm?commid=54138, ISO - International Standard Organization (2012).

[23] M. Csikszentmihalyi, Flow: The Psychology of Optimal Experience, Harper Perennial, London, UK, 1990.

[24] E. L. Deci, R. Flaste, Why we do what we do: understanding self-motivation, Penguins Books, New York, 1996.

[25] R. Bandler, J. Grinder, The Structure of Magic Volume 1; A Book About language and therapy, Science and Behavior Books, Inc., Palo Alto, CA, 1975.

[26] HTE, Robotica 2011, http://www.hitechexpo.eu/it_hte/index_rbt.asp (2011).

Appendix A. A game developed without guidelines

In this section, the development of another PIRG when the presented guidelines were not yet available is summarized. The problems occurred, and where the guidelines might have supported the game development (missing guidelines in italics) are pointed out. The development of such games taught lessons that have been included in the guidelines presented in this paper.

Appendix A.1. Game concept

The mission of the robot is to bring a secret message to its home base. The human player has to prevent this by staying on its way and eventually using one of his/her weapons. When injured by the weapon, the robot loses some energy and, with this, some of its capabilities, up to complete immobility and death. In the environment, it can find some “Friendship bases”, where it can safely stay just for the time needed to recover its injuries and re–gain a full amount of energy. The robot is also equipped with a special “ray gun” able to block any offending device for a given time, including the human player’s weapons. The use of this ray gun needs energy and the robot strategy is to use it only when really needed, e.g., when the opponent stays on the robot way for too long time. When this ray is active, the human player's weapons are deactivated for a certain amount of time.

Appendix A.2. Game Implementation

It was decided to implement the game by using Spykee, a two-track differential drive robot by Meccano®, enhanced to detect obstacles and to be detectable by a WIIMote®. The robot can send low-resolution images via WI-FI and these are elaborated to detect color blobs corresponding, respectively, to the user's legs, the target base, and the friendship bases. The player uses the WIIMote® as a weapon: according to gestures to extract the WIIMote®, she can have it representing three different weapons: rifle, gun, or sword, with different effects on the robot.

The goal for both the players is clearly defined before the game: the human player should prevent the robot to reach the home base by using his/her weapons, and/or by taking positions between the robot and the base. The selected robot moves quite slowly w.r.t. the player, and it cannot really run away to reach the base. This makes the hunting quite unrealistic, since the player can easily reach the robot and keep it under control at any time (*Define a story for the game and stick to it* and *Robot as a rational agent*). The different weapons are introduced to give more cognitive load to the player, expecting that this could induce more engagement. However, the mechanism of weapon activation (based on gestures done with the WIIMote®) is slow and quite complex (*KISS*), and the effects of the different weapons are comparable, so that players do not really care about changing their weapon (*Test the assumptions* and *Do it for the fun*). The robot's weapon is not much effective in the game conditions, since the user has only to wait to be able to use her/his weapon in turn (*Do it for the fun* and *Robot as a rational agent*). In general, the game is quite complex and slow enough to prevent a real involvement of the user. Although many interesting elements have been implemented, most of them were not functional to produce an effective PIRG, and developers were not ready to reconsider implementation (*Agile implementation*).

If the guidelines were available, the game had probably been more simple, maybe with only one weapon, the selected robot had been faster, or the player's movements had been limited, e.g., by obstacles or virtual walls on the playground. This PIRG is presently under re-definition.