Maggie: A Social Robot as a Gaming Platform

V. Gonzalez-Pacheco · Arnaud Ramey · F. Alonso-Martin · A. Castro-Gonzalez · Miguel A. Salichs

Abstract Edutainment robots are robots designed to participate in people's education and in their entertainment. One of the tasks of edutainment robots is to play with their human partners, but most of them offer a limited pool of games. Moreover, it is difficult to add new games to them. This lack of flexibility could shorten their life cycle. This paper presents a social robot on which several robotic games have been developed. Our robot uses a flexible and modular architecture that allows the creation of new skills by the composition of existing and simpler skills. With this architecture, the development of a new game mainly consists in the composition of the skills that are needed for this specific game. In this paper, we present the robot, its hardware and its software architecture, including its interaction capabilities. We also provide a detailed description of the development of five of the games the robot can play.

1 Introduction

Playing games is one of the activities in which children spend the most time and show the most interest. Nowadays, children are starting to incorporate robots into their games. For instance, the Pleo robot plays a very basic game called tug-of-war and it is just a substitute pet. NEC, a Japanese firm, also performs research on entertainment robots with Papero, an entertainment robot, which can perform dances, mimicry, riddles, quizzes, fortune telling and other games.

University Carlos III of Madrid, Systems Engineering and Automation Department, 281911 Leganés (Madrid), Spain

Other works try to mix entertainment and education in robotics leading to edutainment robots. In [23], preliminary experiments on remote education have shown that students interacting with robots show pleasure and interest. [10] shows that children understand better the gaming situations when the robot shows emotional behaviors. Moreover, children seem to enjoy the game more when they play with embodied characters than when they play with screen-printed characters. [17] describes how children play with Roball [12], a plastic spherical robot, which adapts its behavior to increase and sustain interaction.

In addition, it has been shown that robots have a psychological effect on patients, improving their motivation, as demonstrated with the Paro robot. Furthermore, some studies use robots for the therapy of disabled children. For instance, children who suffer from severe disabilities use robots with the aim to learn and improve their quality of life [3]. Depending on the games, the robots will help the users to develop and improve their abilities. In [8], [14] a remote-controlled furry robot is used to guide the therapy of disabled children. The robot is controlled by sensors, adjusted on the body of the child. Through those means, she can move the robot in order to make it tell a story. Similarly, other works have focused in the help of autistic children [17], [18], [7].

Gaming seems to be promising as a mechanism to facilitate the research in Human-Robot Interaction (HRI) field [22]. However, almost all of these works use robots that are limited in the range of games, actions or applications they can play. The lack of such software flexibility results in robots with shorter life cycles than others which have a more flexible architecture [9]. For example, in the field of toy robots, a robot that is capable of playing numerous types of games can be addressed to a

wider range of people than another robot with a smaller pool of games. Extending the gaming capabilities of social robots could enlarge their life cycle, especially if we increase the number of games they can play as well as we give the user different ways of playing each game.

We present an easily extensible robotic platform for the development of board games as well as educational applications. The platform consists of the robot Maggie [19] and its software architecture. Both are the base for the creation of several skills that allow the robot to play a wide range of games, for instance board games.

The remainder of the paper is as follows: next section shows a description of the robot Maggie introducing its hardware and software. Section 3 presents the interaction mechanisms of the robot. In section 4 we discuss the robot as a gaming platform and present a description of some of the games that we have developed. Finally, in section 5 a brief conclusion and future issues are discussed.

2 Description of the robot

The robot Maggie [19] is a research platform aimed for study human-robot interaction (HRI). Research topics with this robot focus on finding new ways to adapt the robotics potential to provide to human users new ways of working, learning and entertaining. Following, software and hardware included in Maggie are briefly introduced.

2.1 Hardware

Maggie is designed as a 1.35 meters tall girl-like doll. Its base is motorized by two actuated wheels and a caster wheel. The base is equipped with 12 bumpers, which detect contacts with objects. Above the base, a laser range finder (Sick LMS 200) has been added. Inside Maggie's belly, an infrared emitter/receiver allows the robot to control home appliances.

The upper part of the robot incorporates some interaction modules: several touch sensors are located in the surface of the body. A laptop touch screen, located in the chest, provides bidirectional communication between the robot and the users. Two one-degree-of-freedom (DoF) arms are attached to both sides of the body. On top of the platform is located a two DoF robot head, which presents an attractive design. Inside the head is placed an RFID (Radio Frequency IDentification) antenna to read RFID tags. For image acquisition, the robot has a Logitech QuickCam Pro 9000 camera located in its mouth. On both sides of this mouth is located an array of LEDs (Light Emitting Diodes), which

light up when the robot speaks. This, together with two animated eyelids, endows Maggie with a life-like face. These features are illustrated in Fig. 1.

Recently, the robot has been expanded with a Microsoft Kinect device [15]. The Kinect sensor is a sensor capable of providing color images and depth maps of the environment simultaneously. Among other things, it can be used to track targets and to obtain the pose of the human at which is looking at.

The audio capturing system is based in a wireless microphone headset. The robot can speak with people using a loudspeaker built-in Maggie's neck.

Maggie is controlled by a laptop hidden inside her body. In this computer resides the software control architecture of the robot, which is described in the next section.
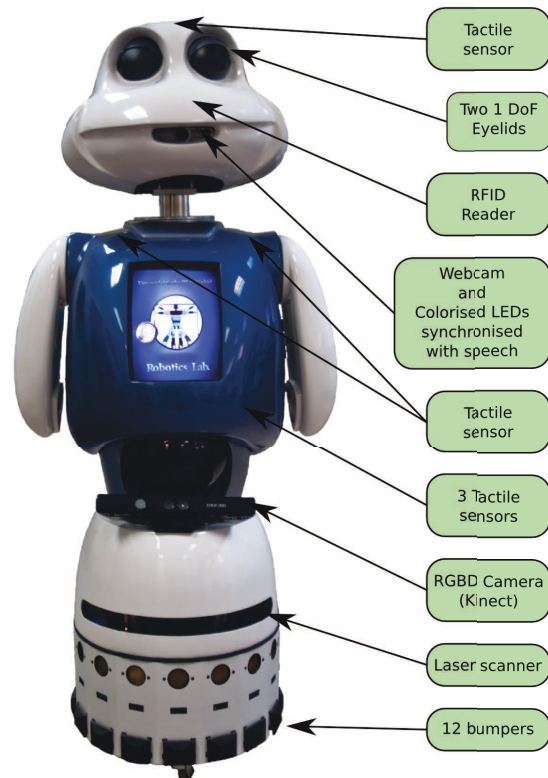

Fig. 1 The hardware equipping Maggie.

2.2 Software architecture

The software architecture of the robot is the Automatic-Deliberative (AD) architecture [1]. AD is composed of two levels, the automatic level and the deliberative level. The automatic level is where the low-level control is

done. Here are located the modules that provide communication and control of the sensors, motors and other hardware. The deliberative level is where reasoning and decision processes are placed.

The essential component of the AD architecture is the skill. A skill is an entity with the capacity of reasoning, processing data or carrying out actions as well as communicating with other skills. For example, the *laserSkill* manages the laser range finder of the robot, formats its data and shares it with the rest of the skills. Thus, other skills can benefit from the data obtained from the *laserSkill* to, for example, build a map that, in turn, can be shared with other skills. A detailed description of the AD architecture can be found in [1] and [16].

3 Interaction with Maggie

To date, the robot has several mechanisms of interaction that will be detailed in the next section. All these interaction mechanisms can be used within the games to improve the experience of the human user while playing with the robot.

3.1 Voice System

One of the most important interaction mechanisms of the robot is the voice system.

It allows the robot to speak with and listen to humans. The voice system is composed of a set of skills that give the robot a complex and powerful communication system. The main voice skills are: Automatic Speech Recognition (ASR), Emotional Text To Speech (eTTS), Speaker Identification (SI), and DialogSkill, which is based on the standard VoiceXML. [1]

3.1.1 Emotional Text To Speech

Emotional Text To Speech (ETTS) is a technology which converts written texts into speech. The skill uses an external software framework, Loquendo, which is wrapped in the form of a skill. This enables the remote use of the Loquendo framework by other skills. In other words, any other skill of the architecture can use the functionalities provided by the ETTS skill. Some of these functionalities are adding and removing a prompt into a prompt queue; knowing when a prompt has started or finished; controlling the speed and volume of the speech; starting, pausing and resuming a speech; entering into quiet-mode; and controlling the emotion and the language of the speech.

This skill allows synthesizing voices in several languages. Currently, Spanish, British and American English languages are supported.

The voice generated is clear and easily understandable by humans. The system produces a voice similar to the human voice rather than a "robotic" voice. Moreover, through varying the voice tone and speed, the robot show interrogative or affirmative intentions and express emotions like happiness, sadness, nervousness or tranquility.

Additionally, the ETTS system allows generating voice gestures such as: laughter, yawning, whistles, tickles, sighing, etc. This increases the expression of the voice prompts, which can help the user understand better what feeling the robot wants to express.

3.1.2 Automatic Speech Recognition

ASR refers to the process of capturing an acoustic signal from a microphone input and converting it into a string of written words. It is a subcomponent of speech understanding, which transforms speech acoustic signals into semantic-pragmatic representations. This process is composed of three states: speech recognition, literal understanding and contextual understanding.

ASR based technologies are classified into two basic types: *Speaker-dependent systems* and *Speaker-independent systems*. *Speaker-dependent systems* refers to a configuration where the system is trained to recognize the voice of one specific user. Hence, the system can only recognize the voice of this user. The recognition is said to be open, that is, any sentence is possible and allowed. These systems are commonly known as dictating machines. On the other hand, *Speaker-Independent Systems* are capable of recognizing sentences that meet specific sets of grammatical rules. They are also called grammar-based systems. On the other hand, any user can use such a system without having previously undergone training with it.

We mainly use this second type of system within our architecture. Any human can hence talk to the robot in a natural language and without training phase. Again, a professional software vendor, Loquendo, provides the software for this feature. We wrapped it into a skill in order to enable its use by the other skills. Using this skill, Maggie can understand Spanish, American English and British English.

Important improvements in this skill are under development in several research areas:

– Noise cancellation: the ASR engine and the microphone are able to eliminate the residual noise.

---

[1] http://www.w3.org/TR/voicexml20/

– Natural Language Understanding: the semantics grammar allows to get the semantic interpretation of the recognition of a whole utterance.
– Partial results: the skill is able to provide partial recognition results before the user speech has ended.

3.1.3 Speaker Identification

Maggie is also able to identify the person who is talking to her using a previously recorded voiceprint database. This functionality is called *Speaker Verification* and is powered by the Loquendo speech system and it is integrated into the ASR Skill. In order to identify the speaker, the Speaker Verification system first needs to learn the user voice tone. For this, the robot requires the user to utter some sentences. Then, the user can identify herself using her voice.

3.1.4 Dialog Manager

A dialog system is an automatic system which emulates a human in a natural conversation with another human. Therefore the task of the DialogSkill (or dialogue manager) are controlling the flow of the conversation, accepting spoken inputs from the user, producing messages (to clarify, disambiguate, suggest control, assist and constrain the conversation) which will be conveyed to the user, and interacting with the internal and external resources. Dialogue strategy is a subcomponent of the dialog skill in which the dialogue strategy to be followed is chosen. There are several possible strategies:

– Based on Finite States Machine.
– Based on Form-filling (information slots) [13].
– Based on Agents.
– Stochastics:
  – Markov Decision Process (MDP) with reinforcement learning. [6]
  – Partial Observable MDP (POMDP) [21].

Currently, DialogSkill is based on the VoiceXML standard, which implements form-filling strategy. Nowadays, it is the most popular strategy due to a good trade-off between flexibility and accuracy.

Remark that in a spoken dialogue system, determining which action a machine should take in a given situation is a difficult problem because automatic speech recognition is not fully reliable. Hence, the state of the conversation cannot be known with certainty.

3.2 Computer Vision

The camera in the mouth of the robot enables it to "see" its environment. A skill is in charge of acquiring the images from the camera and sharing them with the other skills, thus creating a distributed image processing system.

Most of the vision skills process this image stream thanks to the OpenCV library. [2] A few applications that these skills perform are face detection, tracking of a user in the environment, detecting some objects, etc. We will detail more later on.

The recent addition of the Kinect depth-camera adds some potential to the vision system. We can obtain a 3-dimensional reconstruction of the robot environment. Thus, we can detect the user's body gestures and movements, or obstacles such as pieces of furniture, in real time and with a low processing cost.

3.2.1 A tool for games: playzones and playzone detection

One of our goals concerning the gaming experience is to make Maggie a good partner for playing classical board games, such as tic-tac-toe or hangman.

However, locating a board game in an image provided by the camera is not easy. The image is full of pieces of information, including useless ones: the camera integrated in the robot supplies images containing the whole environment of the robot, as it is not only focused on the play zone. Furthermore, as we search a solution adapted for every board game, we cannot make any assumption about the global shape of the game board. Indeed, its content is random, and we must keep this assumption: it can contain any type of game inside of it, which means random colors and shapes. Thus, it is impossible to use the usual object finding methods, which are based on the object content and color such as the SIFT descriptors [11] or, more recently, the SURF features [2].

Consequently, we have to add stable and invariant information to our board. The solution we propose here is to add a thick rectangular border around the game, with some blank padding between the actual game and the border. We call this border the **"playzone"**. This creates a kind of "marker" around our game, which is independent of the actual game, and it is easier to localize in an image. For most of the games, the playzone is drawn with a thick marker on a sheet of paper which lies on a table. It must stand in the cone of vision of the camera to let the robot see it with the camera located in the mouth. For instance, an 80 cm high table, one meter away from the robot, perfectly fits this condition.

---

[2] OpenCV (Open Source Computer Vision - http://opencv.willowgarage.com/) is a popular library of programming functions for real time computer vision. It is released under a BSD license, and is free for both academic and commercial use.

The algorithm for detecting the playzone in the image supplied by the camera is made of several steps. Here is an overview. An illustration of the different steps is visible in figure 2.

1. We transform the input frame to a monochrome version. Each pixel is transformed to black or white depending of its lightness value.
2. We extract all the connected components of the monochrome frame.
3. We evaluate the likeliness of each connected component with the square shape. For this, we use a specific geometrical distance, called Modified Hausdorff Distance (also known as MHD, cf [5]). The lower this specific distance returns between both compared components, the more similar they are. This enables us to spot the component corresponding to the thick black square, if any.
4. Using some geometrical methods, we detect the corners of the connected component corresponding to the playzone.
5. We rectify the zone delimited by the four corners to a square shape of the wanted size, for instance, a 300x300 RGB frame. As the resolution of the playzone in the input frame is often low, the missing pixel values are obtained through a linear interpolation in this corrected version.

3.3 Radio Frequency Identification

Another mechanism to interact with the environment is the Radio Frequency Identification (RFID) system [4].

This type of interaction is not a type of interaction which is naturally used by humans. It relies on the exchange of data via radio waves between the reader of the robot and tags located in objects. It allows the robot to identify and retrieve information about these objects when they are presented to the robot by the human user.

Maggie has a RFID reader located in her body. It is next to the mouth and invisible from outside. When the human presents the object to the robot, the robot is not only able to identify the object itself, but also is able to retrieve more information related to it. We have developed several skills that use this kind of interaction. Maggie can recognize and retrieve information about certain products like drugs and toys.

Note that many utilities developed with this identification system may be processed by either the vision system or the RFID detection system. While the former is similar to the human sensors, the latter makes it is easier to develop applications that are robust to certain conditions like low light, etc.

(a) Input frame for the playzone detection; the playzone is the black square on the left of the frame
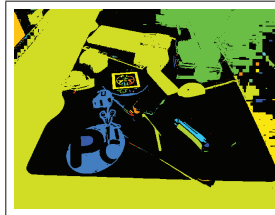
(b) thresholded version of the input frame, where every connected component is drawn with a different color
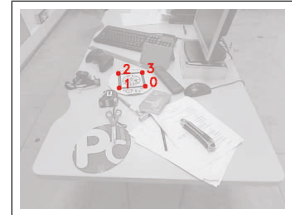
(c) illustration of the detected corners for the connected component symbolizing the playzone

(d) the final version of the playzone detector: rectified and cropped

Fig. 2 The different steps of the playzone detection

3.4 Touch sensors

Maggie can detect when a person touches certain parts of its body (head, arms and upper body) thanks to a dozen capacitive touch sensors placed over her body. This skill notifies the architecture whenever a human

touches a sensor. It is useful in games as another interaction mode. To date, these sensors are only binary sensors. Therefore they cannot specify the intensity of the contact, but only its presence or not.

3.5 Built-in touch screen

Maggie is equipped with a touch screen located in her chest. We use the distributed capacities of the AD architecture to exchange data between the touch screen and the main computer in a flexible and robust way.

The touch screen can be used as an input as well as an output device. As output, skills can supply information that needs to be displayed to the tablet. It can for instance show the image stream acquired by the camera or the Kinect.

As input, the user can control what is running inside the robot through graphical menus. He can also draw figures with his fingers. In this scenario, the image will be passed to the main computer, with a faster CPU, for the image analysis. A picture of a user interacting in such a way is visible in fig 3.

Fig. 3 An user interacting with Maggie through the touch screen in her chest. Here, he is using a very simple drawing application.

3.6 Interaction via smartphones

Another recent improvement enables the control of the robot through smartphones (such as the iPhone). With it, it is possible to enable or disable skills within the robot through an iPhone application, as well as control the robot position and orientation with your fingertips, thus providing a natural mechanism of tele-operation. The main difference between the tablet-based and this is that the latter is not equipped inside Maggie robot.

3.7 Engagement gestures

Interaction between humans heavily relies on gestures. In a similar fashion, Maggie performs gestures with her body members to express emotions and feelings.

An example is head movement. Maggie's head tracks the human movement, so that the robot always looks directly to the human player. This is made by the detection of objects thanks to the laser range-finder and the location of the sound source with the array-microphone.

Another one is eye movement. The robot can open and close its eyelids. When it is in sleep mode, the eyelids are mainly closed. They slightly open now and then. If it is a activate state, its eyelids are completely open, and Maggie blinks from time to time. They close halfway to show that Maggie is perplexed in some skills.

4 Gaming platform

Maggie is able to play several games thanks to the design of her software architecture. Since AD enables the construction of new skills that make use of previously built skills, the development of new games becomes a task where most of the work has already been done by other skills. For example, to create a new game, it is possible to use the data provided by the vision skill that locates the game board and returns a rectified image only containing the current state of the game. After that, the new game skill has to process this normalized image and can calculate the next movement the robot will play. It then uses the communication skills to tell the human what is going to be the robot's next movement. In other words, creating a new game consists in only developing the algorithm of the new game, and after that complement it with the necessary skills to allow the robot to interact with the world and the human player.

The robot interaction capabilities, seen in the previous section, are used to allow the robot to play board games with humans in the similar manner they play with other humans.

The voice system is not only used during the games. During the complete operation of the robot, it can be operated by voice. The robot understands several dialogues that allow the user to activate and deactivate by voice the available skills like the games or other robot utilities. Our aim is to have the robot completely operated by voice in order to increase the feeling of being interacting with a real person.

The following sections show some of the games we have developed so far to test the interaction capabilities of the robot and are an example of how to create a complex gaming robotic platform by the addition of

several aggregated skills. Several of the described games are based on classical board games.

4.1 Peekaboo

Fig. 4 Maggie playing peekaboo

Description of the Game. Peekaboo is a game played with babies where the older player hides her face and pops it again into the baby's field of view. The game takes advantage of the baby's inability to understand the permanence of objects.

How to play with the robot. In the robotic version, the robot plays the role of the baby. Then, a person hides her face from the robot in the same way as the classical version. The purpose of the game is hiding the face to make it undetectable by the robot's computer vision system. If the robot does not detect any faces, then it says that it cannot see anybody. When the person shows her face to the robot, it then tells that now it is seeing her. More than one person can play the game at the same time. In this case, the robot tells to the group the number of faces she is seeing.

A sample is shown in fig 4.

Under the hood. The game was developed as a test of one of the AD face detection skills. After that, the game was integrated into the pool of games that Maggie can play.

The face detection algorithm is made through an object detector. It uses a cascade classifier with Haar-like features [20]. These features represent local differences of contrasts in the image values, altogether with their orientation and scales. This classifier returns the number of faces in the frame, together with the confidence for each detection. We only keep the faces with high confidence detection.

Additionally, the game uses some interaction mechanisms to make the game more engaging. The main one is the voice system, which allows the player to know whether he is being detected or not.

4.2 Guessing a character

Description of the Game. This game is a version of the popular Twenty Questions game. There is one player commonly called as *answerer*, and one or several called as *questioners*. The game is played as follows: the answerer thinks of a fictional or a real character, without revealing who it is. The answerers try to guess who this character is. To do so, they can ask to the player up to twenty questions about this character. The only answers that the player can give to these questions are "yes", "no" and "I don't know", for instance, *"Is the character a man?"*.

How to play with the robot. The robotic version of the game is similar to the classic one, but with the robot acting as the questioner part. The robot begins the game asking questions to the player using its voice system. The human answers the questions talking (with her own voice) directly to the robot. When the robot estimates it has gathered enough information about the character, it tries to guess its identity by proposing a name.

Under the hood. Three main components of the robot are used in the game. The first two components are the ASR and the TTS skills, for the voice interaction with the human player.

The third component of the game is a skill that, using the robot's built-in WiFi system, connects to a public web server (Akinator [3]). This server maintains a wide database of characters, altogether with a list of questions and their answer that can lead to each character. Using machine-learning techniques, the web server selects questions so that the answer prunes as many potential characters as possible. The web server then returns to the skill the appropriate question, or a character suggestion if the information gathered is sufficient.

4.3 Tic-tac-toe

Description of the Game. This game is a classical turn-based board game, which consists in two players fighting for putting three aligned counters in a 3x3

---

[3] Akinator is the name of the game, http://en.akinator.com/

board before the rival does. Each player must wait her turn to put a single counter in a free cell of the game board. The first player who manages to form a row of three consecutive counters wins the game.

How to play with the robot. In the game, the robot and the human play against the other. The robot can play with either crosses or circles. The player can start the game or let the robot do it. If the human starts the game, it does it by putting down a counter of one kind. Then, the robot uses the other type of counters. If the robot starts, it always chooses crosses.

The human can put its own counter in a free space, and then tells the robot it is her turn. The robot then chooses a free space in the board and tells the human to put its counter in that space. When one player wins the game or when there are no more free spaces in the game board, the robot tells the human that the game is finished and reports the result of it.

It is also possible to play through the touch screen located in the chest of Maggie. The image of the play-zone is supplied to it through the communication mechanisms of the AD architecture. The user directly draws with his fingers on this image, and so does Maggie: the tic-tac-toe skill draws Maggie moves on this shared image.

Both these methods are shown in figure 5.

Under the hood.

The vocal interaction between the robot and the human is done by the ASR and TTS skills. The robot tells the human the rules of the game, asks the human to put a counter in a certain position on the game board and updates the human with the state of the game (i.e. is finished, who has won, etc.).

The robot uses the playzone mechanism, explained in section 3.2.1 to recognize the game board.

The specific tic-tac-toe analysis is made in several steps. First, the image of the playzone is segmented according to a 3x3-grid pattern, as is a real tic-tac-toe game. For every cell of this grid, we compute the percentage of black pixels. If it is lower than a given threshold, the cell is considered as empty, that is, not played yet. Otherwise, we extract the biggest connected component of black pixels of that cell. It is then compared to the reference cross and round patterns using the geometrical MHD distances [5].

Once the position of the game counters is defined, it is necessary to apply an appropriate algorithm to decide the next move. The game algorithm is based on minimizing the "damage" that you can receive from the adversary and maximize your chances of winning.
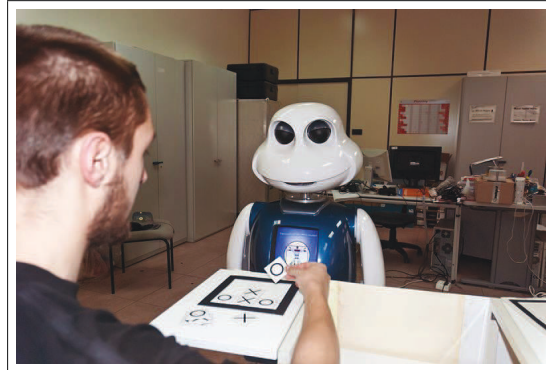
(a) Maggie playing to tic-tac-toe with cardboard counters
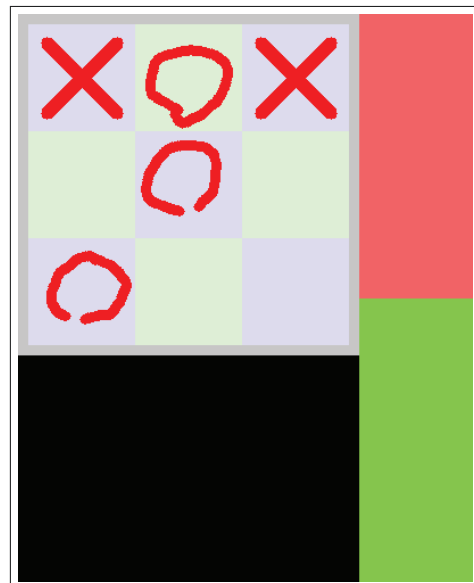
(b) The same game, using tablet: a sample of game. The rounds are drawn by the user fingers, while the crosses are drawn by Maggie. The red and green buttons enable to respectively correct and send the user drawing. The shape of the round is not perfect, but still easily recognized by the image analysis algorithm.
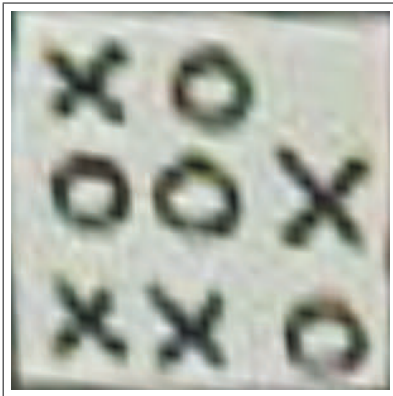
Fig. 5 The steps of the tic-tac-toe algorithm.

4.4 Hangman

Description of the Game. This classical board game is played with a pencil and a piece of paper. The game consists in one player thinking of a word and another trying to guess that word by suggesting letters. The word to guess is represented by a row of as many dashes as letters the word has. For example, if the word has seven letters, the player writes seven dashes, each one corresponding to one letter. If the guesser suggests a letter that is a part of the word, the other player writes this letter in all the correct positions of the word. If

(a) Input image for the robot (color perspective image)
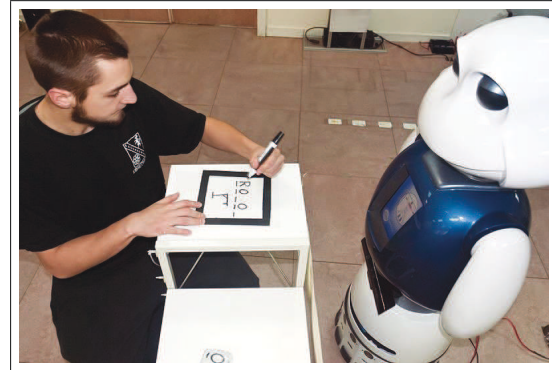

(b) Image corrected in plant view

(c) Results of the image analysis by the tic-tac-toe algorithm

Fig. 6 The different steps of the tic-tac-toe algorithm.

Fig. 7 Maggie playing hangman

the suggested letter is not in the word, the other player draws one element of a hangman diagram indicating a failure of the guesser. Both players agree at the beginning of the game on the number of elements of the hangman diagram, i. e., the number of wrong letters the guesser can try. The game ends when the guesser completes the word or when the drawer completes the hangman diagram.

How to play with the robot.

The human player is responsible for thinking of the word, and writing the dashes corresponding to the num-

ber of letters of the word in the piece of paper. The robot is responsible for guessing the word. [4]

The game board is marked with the playzone mechanism seen in part 3.2.1. When the human puts the paper with the dashes on the table, she has to tell the robot to start the game by touching its shoulder or by giving a vocal order. Then, the robot tries to guess the word. To do this the robot has several rounds. In each round, the robot proposes a letter that might be in the word and announces it using the voice system.

If the proposed letter is in the word, the human must write it on the sheet of paper, just above the corresponding dash. Once the human has done this, she warns the robot to make another attempt using a voice command. After that, the robot analyzes the game board, counting the number of dashes and the letters that it has guessed. This is visible in figure 7.

For instance, imagine that the user choses the word *"robot"*. If the robot asks for the letter *"o"*, the human must write an *"o"* above the second and the fourth dashes. Doing that, the robot is able to detect that it has guessed a correct letter and to detect that this letter appears twice in the word.

The game ends when the robot guesses the word or when it reaches a maximum number of failures. The allowed number of failures is six. If Maggie reaches that point, she notifies the player that she has lost the game.

Like in the tic-tac-toe game, it is necessary to put the table near the robot to allow it to see the game board (Fig. 7).

As for the tic-tac-toe game, it is possible to use the touch screen as an input. This substitutes the playzone mechanism: the user directly draws on the frame on

---

[4] Indeed, as Maggie is not able to grasp objects, she is not capable of drawing on the sheet of paper the number of letters of the picked word, the occurrences of the letters the user proposes, etc. This could however be performed by giving vocal orders to the player, but this wouldn't be a very natural way of playing hangman.

the screen the letters that Maggie proposes, if they are present in the word. He can also draw the hangman diagram on the screen. The tactile version enables to change roles, as the hangman algorithm can directly draw on the image exchanged with the user.

Under the hood.

As we use the playzone system, the detection of the game position inside the frame supplied by the camera is easy. Hence, the core challenges of this application are the detection of the number of letters at the beginning of the game and the recognition of the hand-written letters.

The detection of dashes is made at the first turn. We find all the black connected components in the frame. We remove the ones that do not have the appropriate rectangle shape and size. Then we search for an alignment between the remaining ones.

In each turn, the user must write clearly and with a black marker the letters that the robot has guessed. The recognition of the letters could be performed with an external optical character recognition (OCR) application. However, it would be dependent on the way the user writes: an "O" letter whose stroke slightly open on top would result as a U. Consequently, we chose to use the modified Hausdorff distance [5] to compare the connected component corresponding with every allowed letter. The allowed letters are the letters Maggie has already proposed. It is for instance impossible that the user writes an "M" if Maggie has suggested "A, B, C".

In Fig. 8a, we can see the game board in the top right corner. In Fig. 8b, the robot has obtained the rectified image from the playing area (plant view) and has detected and identified the letters written on it.

The algorithm used in the game is based on finding the words that can match with the current state of the game. We have two dictionaries of words, one in English and another in Spanish, each one with approximately 100,000 words. These words are the most common in both languages.

The human part of human-robot interaction in this game consists in writing in the board game and talking to the robot to indicate it the end of each turn. In the other part, the robot reads the letters written by the human and asks relevant questions to the human.

4.5 Animal Quiz

Description of the Game. This game is designed for studying how to improve the interaction between robots and children. In this game, the robot plays with a group of children. A range of soft toys is disposed on a table next to the children. The role of the children

(a) Table with the board game

(b) OCR in the rectified image

Fig. 8 The different steps of the hangman algorithm.

is to understand the questions Maggie asks about the soft toys. They then have to bring to Maggie the soft toy she is asking about. Each soft toy has three unique properties that make it different from the rest: color, shape, and name, that is, there is no pair of soft toys with the same color, shape or name.

How to play with the robot.

The game consists in the robot asking to a group of children to bring her a series of soft toys. To do that the robot asks for one of the properties of the soft toy (shape, color or name). One child picks the corresponding soft toy and brings it to the robot. If the child does not understand the question, she can ask the robot to repeat it. If the child does not pick up the correct soft toy, she may try repeatedly until she gets the correct answer. At the end of the game, the robot tells the number of right and wrong answers. An illustration of the robot playing is shown figure 9.

Under the hood. In order to let the robot detect the soft toy, each soft toy has an RFID tag inside of it. The child gives the soft toy to the robot by bringing it near to the nose of the robot, where she has its RFID reader installed. By reading the RFID tag of the soft toy, the robot detects which animal the child has brought.

When the robot asks for a soft toy, it waits until an RFID tag is detected, i.e. it waits until the child

Fig. 9 Maggie playing animal quiz with two children

puts the toy in front of the robot's nose, typically at a distance of about 20 cm. When the robot detects the RFID tag, it reads its data and then compares the number stored in the tag with the expected answer in order to know if the toy is the correct one or not.

Using RFID tags allows us to play a game without being dependent on the light conditions. This is an advantage when the light it is not good enough to use vision based methods.

5 Conclusions

We have presented a social robot, with strong HRI capabilities and a flexible software architecture, being used as a gaming platform. The paper shows how we have created several games seizing the capabilities offered by the robot hardware and, especially, its software architecture. Having a flexible behaviour-based software architecture facilitates the creation of new applications such as games by composing previously developed skills into new ones.

A social robot with the capability of running an extensible pool of applications is able to adapt to scenarios that were not initially intended or designed for it. In that way, the life cycle of the robot can be increased considerably. The constant development of new applications allow, at the same time, to enlighten new scenarios and areas of use of the robot. In this case, the new area is the gaming area.

Following the exploration of this area, we plan to conduct some experiments to analyse how people react and behave when they play with a robot that tries to behave in the same way a human would do in such game situations. Our hypothesis is that people tends to be more involved when a robotic character shows emotions during the game. In addition, our preliminary results show that robots with more interaction capabilities make people feel more comfortable and, as a result, they tend to spend more time playing with the robot.

References

1. R Barber and Ma Salichs. *A new human based architecture for intelligent autonomous robots*, pages 85–90. Elsevier, 2002.
2. Herbert Bay, Andreas Ess, Tinne Tuytelaars, and Luc Van Gool. Speeded-up robust features (surf). *Comput. Vis. Image Underst.*, 110:346–359, June 2008.
3. Albert M Cook, Max Q H Meng, Jason J Gu, and Kathy Howery. Development of a robotic device for facilitating learning by children who have severe disabilities. *IEEE Transactions on Neural and Rehabilitation Systems Engineering*, 10(3):178–187, 2002.
4. A. Corrales and M. A. Salichs. Integration of a rfid system in a social robot. In Jong-Hwan Kim, Shuzhi Sam Ge, Prahlad Vadakkepat, Norbert Jesse, Abdullah Al Manum, Sadasivan Puthusserypady K, Ulrich Rückert, Joaquin Sitte, Ulf Witkowski, Ryohei Nakatsu, Thomas Braunl, Jacky Baltes, John Anderson, Ching-Chang Wong, Igor Verner, and David Ahlgren, editors, *Progress in Robotics*, volume 44 of *Communications in Computer and Information Science*, pages 63–73. Springer Berlin Heidelberg, 2009.
5. M.-P. Dubuisson and A.K. Jain. A modified hausdorff distance for object matching. In *Pattern Recognition, 1994. Vol. 1 - Conference A: Computer Vision Image Processing., Proceedings of the 12th IAPR International Conference on*, volume 1, pages 566 –568 vol.1, October 1994.
6. D. Goddeau and J. Pineau. *Fast reinforcement learning of dialog strategies*. IEEE, 2000.
7. H. Kozima, C. Nakagawa, and Y. Yasuda. Interactive robots for communication-care: a case-study in autism therapy. In *ROMAN 2005. IEEE International Workshop on Robot and Human Interactive Communication, 2005.*, pages 341–346. IEEE, 2005.
8. Corinna Lathan, Jack Maxwell Vice, Michael Tracey, Catherine Plaisant, Allison Druin, Kris Ed-

ward, and Jaime Montemayor. *Therapeutic play with a storytelling robot.* Number figure 1. ACM Press, New York, New York, USA, 2001.

9. WP Lee, JW Kuo, and PC Lai. Building Adaptive Emotion-Based Pet Robots. In S. I. Ao, Len Gelman, David WL Hukins, Andrew Hunter, and A. M. Korsunsky, editors, *Proceedings of the World Congress on Engineering*, volume I, pages 85–90, London, U.K., 2008. Newswood Limited.

10. Iolanda Leite, Andre Pereira, Carlos Martinho, and Ana Paiva. Are emotional robots more fun to play with? In *RO-MAN 2008 - The 17th IEEE International Symposium on Robot and Human Interactive Communication*, pages 77–82. IEEE, August 2008.

11. David G. Lowe. Distinctive image features from scale-invariant keypoints. *Int. J. Comput. Vision*, 60:91–110, November 2004.

12. F Michaud and Serge Caron. Roball, the rolling robot. *Autonomous robots*, 12(2):211–222, 2002.

13. Eric Nyberg, Teruko Mitamura, Paul Placeway, Michael Duggan, and San Francisco. DialogXML: Extending VoiceXML for Dynamic Dialog Management. *Proceedings of the second international conference on Human Language Technology Research*, pages 298–302, 2002.

14. Catherine Plaisant, Allison Druin, Corinna Lathan, Kapil Dakhane, Kris Edwards, Jack Maxwell Vice, and Jaime Montemayor. A storytelling robot for pediatric rehabilitation. In *Proceedings of the fourth international ACM conference on Assistive technologies - Assets '00*, pages 50–55, New York, New York, USA, November 2000. ACM Press.

15. Arnaud Ramey, Víctor González-Pacheco, and Miguel A. Salichs. Integration of a low-cost rgb-d sensor in a social robot for gesture recognition. In *Proceedings of the 6th international conference on Human-robot interaction*, HRI '11, pages 229–230, New York, NY, USA, 2011. ACM.

16. R Rivas. Robot skill abstraction for ad architecture. *6th IFAC Symposium on Intelligent Autonomous Vehicles*, 47(4):12–13, 2007.

17. Ben Robins and Kerstin Dautenhahn. Interacting with robots: Can we encourage social interaction skills in children with autism? *Accessibility and Computing*, (80):6–10, 2004.

18. Ben Robins, Ester Ferrari, and Kerstin Dautenhahn. Developing scenarios for robot assisted play. In *RO-MAN 2008 - The 17th IEEE International Symposium on Robot and Human Interactive Communication*, pages 180–186, Munich, August 2008. Ieee.

19. Miguel Salichs, R Barber, A Khamis, M Malfaz, J Gorostiza, R Pacheco, R Rivas, A rrales, E Del-

gado, and David Garcia. Maggie: A robotic platform for human-robot social interaction. *2006 IEEE Conference on Robotics Automation and Mechatronics*, pages 1–7, 2006.

20. P. Viola and M. Jones. Rapid object detection using a boosted cascade of simple features. In *Proceedings of the 2001 IEEE Computer Society Conference on Computer Vision and Pattern Recognition. CVPR 2001*, volume 1, pages I–511–I–518, Los Alamitos, CA, USA, April 2001. IEEE Comput. Soc.

21. Jason D. Williams and Steve Young. Scaling POMDPs for Spoken Dialog Management. *IEEE Transactions on Audio, Speech and Language Processing*, 15(7):2116–2129, September 2007.

22. Min Xin and Ehud Sharlin. *Playing Games with Robots–A Method for Evaluating Human-Robot Interaction*, chapter 26, pages 469–480. Number September. Itech Education and Publishing, Viena, 2007.

23. Akihiro Yorita, Takuya Hashimoto, Hiroshi Kobayashi, and Naoyuki Kubota. Remote education based on robot edutainment. In Jong-Hwan Kim, Shuzhi Sam Ge, Prahlad Vadakkepat, Norbert Jesse, Abdullah Al Manum, Sadasivan Puthusserypady K, Ulrich Rückert, Joaquin Sitte, Ulf Witkowski, Ryohei Nakatsu, Thomas Braunl, Jacky Baltes, John Anderson, Ching-Chang Wong, Igor Verner, and David Ahlgren, editors, *Progress in Robotics*, volume 44 of *Communications in Computer and Information Science*, pages 204–213. Springer Berlin Heidelberg, 2009.

Victor Gonzalez-Pacheco holds a bachelor degree in Telematics from the Technical University of Catalonia, Spain, a Master in Science in Robotics and Automation from the University Carlos III of Madrid, Spain, and a Master in Science in Computer Science and Technology with specialisation in Artificial Intelligence from the University Carlos III of Madrid, Spain. He has worked in several research European and Spanish projects as an external consultant research engineer with Telefónica I+D, the research branch of the Spanish Telefónica Group. Currently he hods a Spanish research grant to conduct his PhD in Robotics and Automation at the University Carlos III of Madrid. His research interests include, Human-Robot Interaction (HRI), Personal and Social Robots, and Machine Learning applied to HRI.

Arnaud Ramey holds a double diploma from Ecole Polytechnique, France, and the Royal Institute of Technology (KTH), Sweden. He is now a research engineer at Direction Generale de l'Armement, a section of French Ministry of Defense, while being a PhD student at Carlos III University of Madrid, Spain. His research focuses

on image processing applied to human-robot interaction.

Álvaro Castro González obtained his degree as Computer Engineer from the University of León, Spain, in 2005. After he became a MSc in Robotics and Automation in 2008 at the University Carlos III of Madrid, Spain, where he is currently a PhD candidate and a member of the RoboticsLab research group. Álvaro has been a teaching assistant for several courses at the Department of Systems Engineering and Automation of the University Carlos III of Madrid. His research interests include personal and social robots, human-robot interaction, decision making systems, emotions and motivations on robots.

Prof. Dr. Miguel Angel Salichs is a full professor of the Systems Engineering and Automation Department at the Carlos III University of Madrid. He received the Electrical Engineering and Ph.D. degrees from Polytechnic University of Madrid. His research interests include autonomous social robots, multimodal human-robot interaction, mind models and cognitive architectures. He was member of the Policy Committee of the International Federation of Automatic Control (IFAC), chairman of the Technical Committee on Intelligent Autonomous Vehicles of IFAC, the responsible of the Spanish National Research Program on Industrial Design and Production, the President of the Spanish Society on Automation and Control (CEA) and the Spanish representative at the European Robotics Research Network (EURON).