Elective In Artificial Intelligence

Human Robot Interaction Module:

Project Overview

Child-Pepper Interaction in Pediatric Emergency Rooms

Ratini Riccardo†, 1656801
Simionato Giada†, 1822614

† All the students contributed equally to this project.

Contents

Abstract

In this work we have addressed the task of developing a social robot to reassure and entertain children facing the, usually long, wait in pediatric emergency room of international hospitals. We have considered children hospitalized for non-severe injuries or for routine exams as to make sense for this interaction to exist. Using the Pepper robot by Softbanks, we have designed and implemented a *short-term, multilingual, multimodal customized interactive* system that is able to form social relationships, through the mutual exchange of personal information, and to entertain the children with its storytelling ability and a wide set of games that can be safely performed in this environment. We have exploited the findings of several *Child-Robot Interaction* studies, such as [1, 2, 3, 4] to create a valuable resource for the enhancement of the physical and psychological wellness of ill children.

Chapter 1

Introduction

Having to wait several hours to be checked or treated by a doctor in an emergency room of an hospital is a tough and difficult situation, both from the physical and psychological point of view. This situation if even more painful when it is faced by children. Due to the lack of medical staff and sufficient healthcare resources, the overall procedure may last longer that they can support, and unfortunately, their mental wellness outside the ambulatory is of secondary importance. In order to provide to these young patients enough relief to not only easily wait for their turn but also to increase the efficacy of the cures, we have developed a social robot able to keep them company directly in the pediatric emergency rooms. This robot could also be an useful resource for the children's parents that are sometimes powerless in front of their children suffering.
We have used the Pepper robot to build a *short-term multilingual customized multimodal interactive* system that is able to approach the user respecting the proxemics rules, to establish a social relationship through a set of conversations about the respective personal information and about the reasons behind the hospitalization of the child. In this work we have assumed that the interactions take place with children that are hospitalized for non-severe injuries or just for routine exams or checks. This was motivated by the fact that whenever the problem of which the child is suffering is more dangerous, the urgency of its treatment increases, therefore removing the necessity, or the utility, of such interaction. Pepper then estimates the emotion of its partner as to discover which is the best way in which it can reassure him/her. Finally, it offers a series of services that could entertain the child as to make the wait bearable. These functionalities were adapted on the target user as to find the most suitable ones and currently include the multimodal storytelling ability and a set of games that can be safely performed in the hospital environment, such as the candy game, quizzes and a brand new *guess the pattern* game that exploits the unique functionalities of Pepper. The whole interaction is supposed to happen in an international hospital therefore the child could easily select which language Pepper should speak and understand. It is worth to anticipate that we have used several characteristics of Pepper, from its body posture, to the voice parameters and eye colors to show its internal state as to enrich the social relationship with empathy and to use this channel to convey positive feelings to its interaction partner. In addition the multimodal nature of this interaction should increase the efficacy of the interaction itself in terms of both getting through the children and having a more pleasant interaction.
This report is organized as follows: in Chapter 2 the papers that have inspired our work will be described, putting particular attention to the way in which we have used them in the project. In Chapter 3 we will deeply cover the overall structure of this work, along as its functionalities, while in Chapter 4, all the tools that we have used and the details of the implementation will be disclosed. In Chapter 5 we will comment how the customized interaction took place and, finally, in Chapter 6 conclusion will be drawn.

Chapter 2

Related Work

Due to the high importance of the task addressed in this work, it was possible to retrieve several studies in the literature, focused on analysing different aspects of *child-robot interaction*, especially when dealing with such complex environments as hospitals.
A clear example of this is [1]. This study aims at identifying the benefits of performing child-robot interaction with young patients suffering from cancer. Although the environment in which this interaction takes place is the same for both [1] and our work, the severity of either the illnesses and injuries strongly differ. Moreover, the former is more intended for long-term social relationships rather than short-term ones as in our case. In order to claim that this kind of interaction is beneficial for young hospitalized children, the authors have conducted a study in which the level of depression, anxiety, pain and psychological distress were measured before and after each session of interaction. In these sessions the robot was used under different roles, from which we have taken inspiration for our work, such as a doctor, a *chemo-hero* to learn more about this therapy in a positive way and as a nurse. The robot was able to speak, dance, play sounds and music and replicate body gestures as to induce positive feelings to the suffering patients. By analysing the results it was clear that this kind of interaction partner would help to reduce the negative feelings thus increasing the efficiency of the treatments, while providing also an interesting source from which to learn more about the disease they are suffering of and the possible cures. Since this kind of interaction appeared to be successful also in other medical areas, such as dealing with *Autism Spectrum Disorders*, it was possible to augment the extent of these results also to young emergency room patients. The idea behind [1] represents exactly the purpose of our work: through the cooperation with the robot, the young user can reduce the negative impact that feelings like fear and concern provide. By knowing more about medical procedures, the children could become more cooperative, but could also find entertainment to help them wait the huge amount of time that usually is required for non-urgent cases in emergency rooms.
Another fundamental study in the development of this project was [2]. It presents the findings of an experiment that includes a collaborative task, i.e. playing a turn-taking quiz game, between a group of children and two robots: the former, the *affective*, implemented an *adaptive emotional expression* system, while the latter, the *non-affective*, did not. This system, created by the authors of the study, is a four-stages apparatus that receives in input a series of information on the emotions felt by the child, which the robot is interacting with, and gives as output a precise behaviour aimed at emulating emotions felt by the robot itself. Expressive behaviours, such as emotions and gestures, are important in showing the proper internal state to an interaction partner and therefore to create a social relationship. For this reason we followed the same means to convey emotional expressions of [2], which sets the focus on gestures, voice characteristics and body poses. The connection between this study and our work is additionally corroborated by the fact of having used robots of the same family, i.e. the NAO robot and Pepper robot,

respectively. Even if these robots differ in size, number of degrees of freedom and in way to move[1], the channels in which they convey emotions are the same: the body poses comprehensive of gestures, the voice characteristics such as volume, speech speed and pitch and the eye colors, since both possess controllable LEDs for both eyes. Differently from our work, [2] uses a combination of two values, the *arousal* and the *valence*, provided by a *Wizard of Oz*, to describe the emotion of the children and to influence the robot internal parameters as to provide a grater variety of expressive options. However, we relied on a predefined set of moods: *calm*, *bored*, *sad*, *angry* and *worried*, to comply with the integrated ability of Pepper in estimating such quantity. From this set it is clear that we have focused more on negative feelings, motivated by the context of application of our work. In order to use in the best possible way the findings of this study, we have kept the correspondences among the arousal and valence values and the voice, eye and poses characteristics, thanks to the regroup of the grid results of the former values into the predetermined sets. For the sake of precision, we have mapped intense emotions to hot colors for the eyes, such as red and yellow, and louder and quicker voice, while for more balanced emotions we have used cold colors for the eyes, such as blue and green and lower and slower voice. As regards the body poses, we have used, as in [2], a set of predefined movements associated to the emotion to feel. The values used in this study influenced also the size of these gestures, however since it was not clear how to categorize such quantity into groups, we have not affected this parameter. Another information used in [2] that was not used in this work was the extroversion level of neither the robot nor the child, since it will be not clearly possible to retrieve the extroversion index for each possible future user. The main idea underneath this paper that convinced us to use it in this work is that emotions are contagious, hence it would be possible to use Pepper's emotions to ideally influence the children's ones for giving them greater relief. In addition to some insights of the actual way to implement an adaptive emotional expression system, we also have taken advantage of the findings of this study: since both robots scored very high, it was possible to see the presence of the ceiling effect[2], despite this is a negative phenomenon that could invalidate the study, we could exploit it to induce a positive feeling to the hospitalized children just with Pepper presence. This study claims that the affective robot was the preferred one, since it was able to inspire friendship, empathy and overall positive feelings, that is exactly the purpose of this work. However, [2], described also that some children preferred the non-affective robot, due to the fact that it have not shown high fluctuations in the voice with respect to the affective one, ending up to be more understandable. We have therefore mitigated the voice characteristics settings in such a way that they could still convey emotions, but preventing the robot from becoming not understandable or untrasted as highlighted by the children in [2].

We have not relied only on [2] to understand how and which channels to use to convey emotions, also [3] provided interesting hints about how to successfully express the robot internal state as to mimic most the human behaviour. It points to gestures and the overall posture as the most informative aspect for a social interaction and confirms what [2] suggested as characteristics in the center of attention. However, while the previous study was mostly oriented in disclosing the robot internal state as well as providing empathy in the interaction, this research is more oriented towards how adopting general social behaviours. In fact it suggested to adopt in this work an *inclusive* posture for the Pepper robot, that is a set of frontal positions, with respect to the child with which it is interacting, that indicates an higher awareness of the presence of other people in the interaction. Similarly, a *face-to-face* orientation was taken into account to show more active and engaging interactions. [3] also pointed out some interesting findings about voice characteristics not taken into account in [2]: the presence of *linguistic* and *non-linguistic*

[1]The former is legged while the latter has a mobile base to move within the environment.

[2]The ceiling effect is the phenomenon of the fact that children like so much the robots that is difficult to see differences among conditions.

vocalizations. Due to the relevant position that these aspects occupy in the interaction process, we considered to add some members of the first categories to augment the human-like behavior, while the latter category was not taken into consideration due to the difficulty in synchronizing such sounds with the partner's speech and its necessary understanding to avoid lack of context. Finally, this study also provided some useful information on the criteria with which to start the interaction: the use of proxemics. In this work the interaction starts whenever Pepper detects a human within a certain range through its frontal sonar, stops and starts talking. In order to determine the most correct threshold, i.e. the distance between Pepper and the child, we have relied on the data in [3], choosing a range of 1.2 - 2m as it represents the *socio-consultive* zone, hence the zone devoted to formal relationships. In this way Peppers leaves the choice to the user of getting closer to interact with the tablet or with its sensors, entering progressively into more intimate zones. This choice is additionally supported by the multimodal interaction of the first stages that always allow also spoken responses.
The last paper used in this work was [4]. In particular we have borrowed examples of use and ideas to perform multimodal interaction. The employment of this paper was motivated also by the kind of interaction reported in this study that matches the one expected in our work, that is *personalized short-term multi-modal* interaction. In fact, in the environment in which this project will be used, Pepper will interact with different users every time, therefore it is possible to avoid to remember details needed in long-term relationships. Moreover, the variability in the ages and needs of the children in pediatric emergency rooms is high enough to require personalized interactions that could adapt to the user characteristics. An example of this phenomenon could be taking into account the fact that the children under 6 y.o. are not able to read, therefore all the writings must be also replayed through voice, or being substituted by images or numbers. The last hint taken from this study, that is its essential, is the multimodal nature: in our work the overall interaction happen in a multimodal fashion, by using text to speech and specific graphical user interfaces on the Pepper's tablet, as in [4].
Each of the above mentioned papers contributed to make the child-robot interaction for this work as much as possible effective.

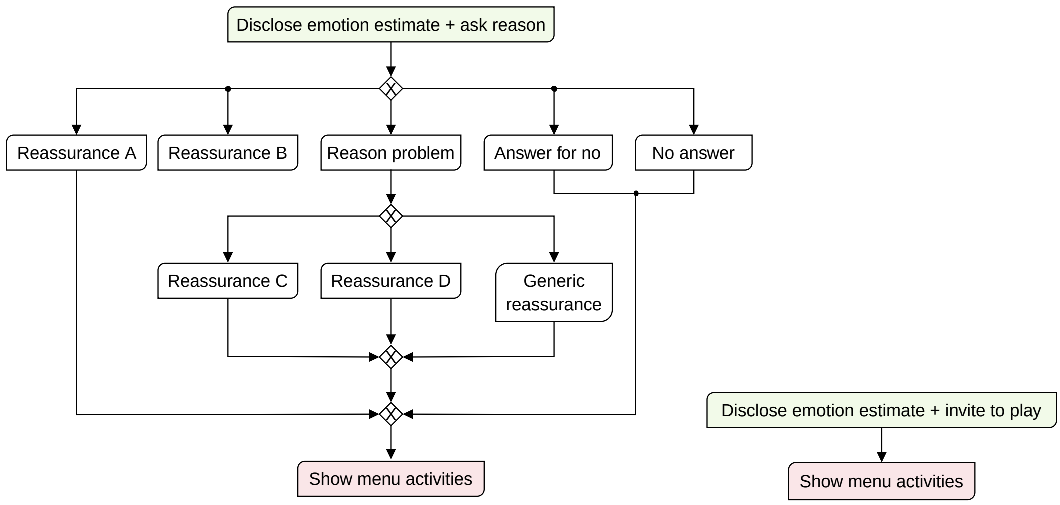
Chapter 3

Solution Architecture

To address the problem of how to perform child-robot interaction in pediatric emergency rooms, we have designed the solution whose schema is reported in Fig. 3.8.
The interaction starts with the *proxemics stage* in which Pepper moves around the environment until it detects an user with which start the interaction. The distance accepted to consider the user as interaction partner is dictated by the socio-consultive area, as described in Chapter 2. When this measurement satisfies this criteria, Pepper stops and starts the actual interaction. Since this project is aimed to be used in international hospitals, we expect that the children speak different languages, therefore, despite English is considered the default language, other possibilities are offered, namely: Italian, French, German and Spanish, thus offering a multilingual experience. Another key feature of this project is the multi-modalities of the actions, chosen as to guarantee a more effective interaction. An example of such functioning is the first dialog representing the *welcoming stage*: Pepper uses its ability to synthesize speech (TTS) to ask in which language the interaction will take place while showing the same question also written on the tablet. The user can then answer in English or in his/her own language through voice of by clicking on the corresponding flag onto Pepper's tablet[1]. Once the language has been selected, Pepper can begin the *information gathering stage* in which introduces itself and asks the name to the user, retrieved by Pepper with its automated speech recognition (ASR) module. NAOQi APIs allow Pepper to estimate the gender and the age of a person, that we have exploited to increase the knowledge about the child as to provide more customized interactions. Despite these functionalities are not allowed in the simulated environment, we have in any case coded and simulated these features (more details on this process will be reported in Chapter 4). Even if the gender does not strongly affects the kind of interaction that Pepper may offer, the age instead is crucial because it is also the quantity estimated with the highest uncertainty. In order to get a correct data, the age is asked directly to the child, by disclosing Pepper's estimate and asking to its partner if the real age is simply greater or lower. Accordingly to the child's answer, Pepper will say another age and this game is repeated until Pepper guesses the correct data. Pepper's side of this interactive game is also replicated using its tablet where each estimate of the age is shown. This latter stage is also the first step employed to form a social bond between the child and the robot to induce positive feelings as deeply explained in Chapter 2. The major connection is expected to be created during the next stage, i.e. *dialog stage*. During this phase a set of predefined dialogues will take place in order to primarily learn the reason why the user was hospitalized. As described in Chapter 1, our work covers non-urgent injuries, as to make sense for this interaction to exist. We have therefore considered two main flows of reasons: to cure a non-severe injury or for a visit of control. Based on the belonging of the user to one of these categories, Pepper will give an appropriate answer. The next step within

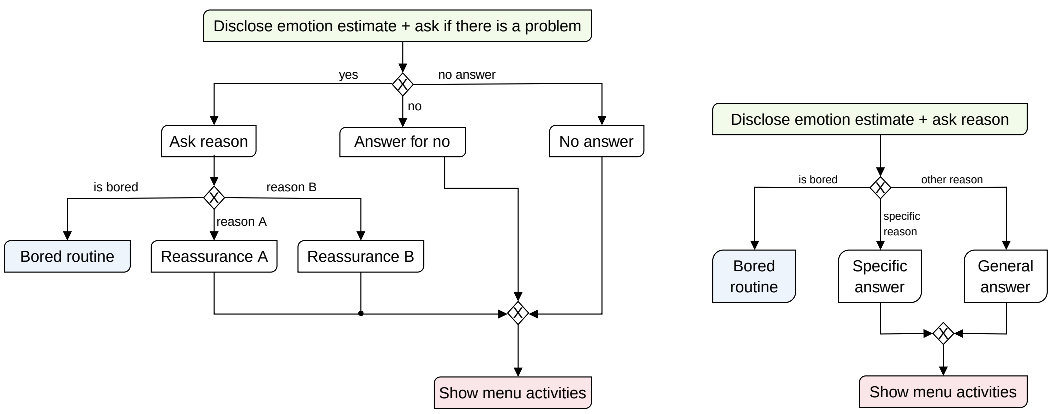
[1]Please note that we have decided to use flags instead of writing the name of the language to avoid stressing the knowledge of English and for the youngest children who are not able to read yet.

this phase is to use NAOQi APIs again to estimate the emotion that the child is feeling. To comply with the result of this method, we have considered a set of basic emotions, steering towards the most common ones induced by the context: calm, bored, angry, sad and worried[2]. Depending on which answers are given by the user, a different conversation will take place, all with the purpose of reassuring the child. We have modeled these dialogues as trees, whose general structures are reported in Fig. 3.1(a) and (b) for the worried and the calm or bored feelings respectively, and in Fig. 3.2(a) and (b) for the sad and angry moods, respectively. For more information on the actual chosen sentences please refer to the code or to the videos attached to this report.



(a) **(b)**

Figure 3.1. Dialog tree for the first set of emotions: *(a)* worried; *(b)* calm or bored.



(a) **(b)**

Figure 3.2. Dialog tree for the second set of emotions: *(a)* sad; *(b)* angry.

As it is possible to notice each of the dialogues above starts with a disclosure of the estimated emotion as to make Pepper more human-like. All the *routines* end showing the menu of the activities that is the first step of the final stage of this work. All the steps within the information gathering and dialog stages are performed by the TTS module of Pepper along

[2]The *fear*, that is the most common feeling, is directly integrated into some of the above-mentioned emotions.

with the replication of the questions onto its tablet as to help also children with non-severe hard of hearing.
The final stage, i.e. *entertainment stage*, is the most complex and interactive part of our work. It has the purpose of actually entertain the child as to keep him/her company until it is his/her turn to be medical assisted. This phase starts with the listing of the main categories of activities that Pepper can do to entertain the child. As the previous stages, also in this case, Pepper will use the TTS to disclose the options as well as showing a graphical user interface on the tablet that the user can press to see which are the further possibilities, or simply selecting the option through voice. The categories of the main menu are:

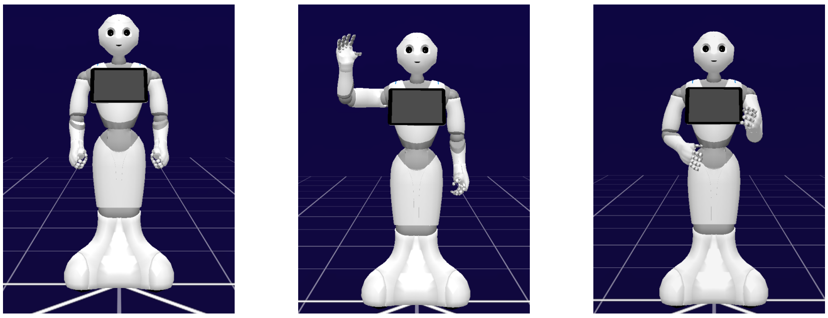
- *Stories*: this section correspond to the storytelling ability of Pepper. It is clear how important the telling of the stories for children is, especially for the youngest. By evoking good memories through well known stories it is possible to distract the child from the painful situation, resulting to be beneficial. Pepper will offer a catalogue of stories accordingly to the previously retrieved age and gender of the child, from which s/he can select an option to start the actual storytelling. This activity is performed through the TTS module as well as showing the synchronized text and images as to mirror the illustrations in children's books. More on the details of the customization of these stories will be explained in Chapter 5.

- *Games*: this section regroups all the games that we have implemented. These activities are competitive games between Pepper and the child and are extremely personalized and different among diverse user profiles (more on Chapter 5). Up to now the framework includes the *candy game*, that is well-known game in which the child has to find in which hand Pepper is hiding a virtual candy, selected through either ASR or by touching the corresponding Pepper's hand. In addition there is a *quiz* game in which the child has to answer correctly to questions that Pepper poses that are also written on the tablet, through ASR or by selecting the corresponding answer on the tablet. Some questions include images as part of the questions themselves. Another game included in this project was *guess the pattern*: through the LEDs in its eyes, Pepper shows a series of colors for a predefined amount of time each, that the child has to replicate, in the correct order by selecting the corresponding colored area on the tablet. Several parameters of this game are influenced by the age of the child, as will be explained in Chapter 5. For all the games Pepper will react with gestures and exclamations according to its win or defeat to make it more interactive.

- *Videos*: this section is a catalogue of videos of cartoons and TV series, based on the recent favourites among the children. The user can then select the desired video and enjoy it while waiting for his/her turn. Despite the content of this section was not implemented in this work, its structure was inserted anyway in the main menu among the other categories.

Once the selected activity is completed, Pepper will ask which other one the child wants to execute, by going back to the main menu. Since the children are well known to get bored quickly and get easily tired of listening to a story or playing a game, providing an exit from the currently activity, even if it has not arrived to a completion yet, becomes fundamental. For this reason we have provided the possibility of interrupting the current type of interaction either by pressing the button `Stop` on the tablet, or simply by saying a sentence containing particular keywords, e.g. "stop" or "enough". Pepper then will get back to the main menu, ready to start a new activity.
During all these stages Pepper will also use a set of body postures as to appear more human and to disclose its internal state that, as stated in Chapter 2, is fundamental to create strong social relationships. Another reason to use these gestures is that, in this way, Pepper can become even

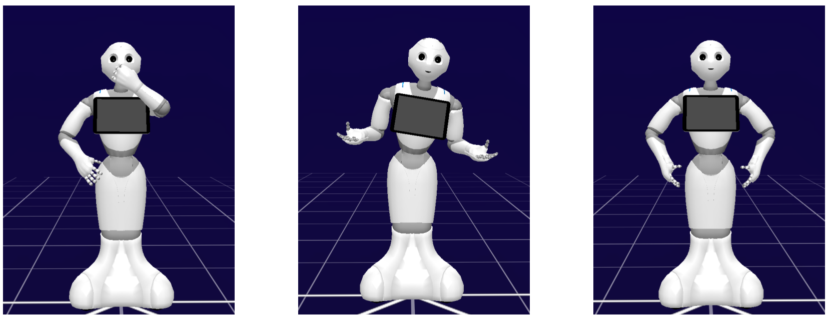
more a positive distracting element for the child. Since the implementation setting, described in Chapter 4, did not allow to use[3] already available animations for the robot to insert into the MODIM actions, we had to carefully design a set of poses that could blend well with the context within the interaction. In total we have created thirteen poses, whose first set that can be seen in Fig. 3.3: in (a) there is the base posture that Pepper uses when idle, in (b) there is the initial greeting one while in (c) there is the gesture accompanying Pepper's presentation.



(a) **(b)** **(c)**

Figure 3.3. First set of postures: *(a)* base posture; *(b)* greetings posture; *(c) I'm Pepper* posture.

In Fig. 3.4 are reported the postures used in the information gathering stage, e.g. in (a) it is possible to see a thinking position.

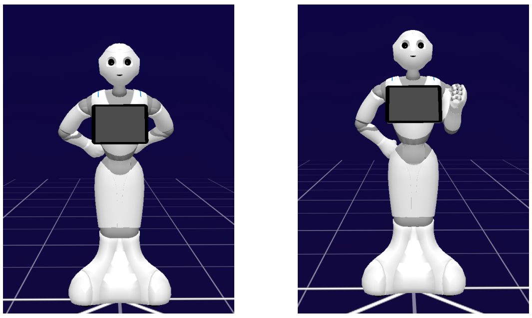


(a) **(b)** **(c)**

Figure 3.4. Second set of postures: *(a)* thinking posture; *(b)* first explanation posture; *(c)* second explanation posture.

The poses used during the conversation about the reasons behind the hospitalization of the child are shown in Fig. 3.5.

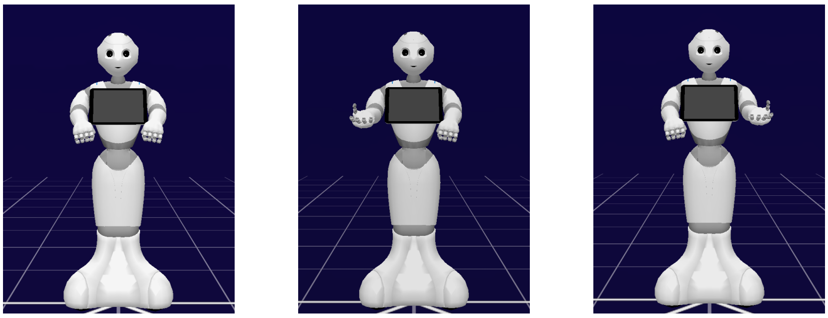
[3]More on this can be found in Chapter 4.



(a) (b)

Figure 3.5. Third set of postures: *(a)* reason-of-hospitalization-is-a-check posture; *(b)* reason-of-hospitalization-is-an-injury posture.

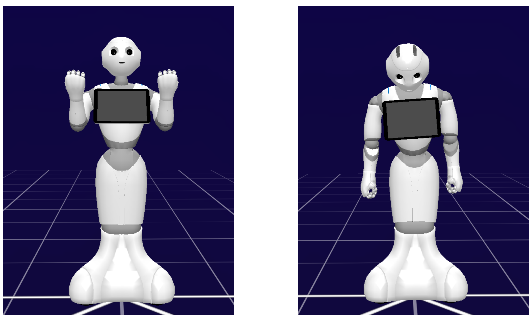
In order to implement the candy game it was necessary to design also standard poses, such as the hands behind Pepper's back and the position as to reveal the right or the left hand, once chosen by the user. The implemented postures are respectively shown in Fig. 3.6(a), (b) and (c).



(a) (b) (c)

Figure 3.6. Candy game set of postures: *(a)* initial posture; *(b)* right hand reveal posture; *(c)* left hand reveal posture.

In conclusion, it is worth reporting also the *celebratory* and the *sad* postures used to emulate a joyful and a bad emotions, used in several moments throughout this work. In Fig. 3.7 it is possible to see these postures.



(a) (b)

Figure 3.7. Fourth set of postures: *(a)* celebratory posture; *(b)* sad posture.

Finally, the interaction will end as soon as the child responds negatively to Pepper asking whether s/he would have liked to do something else together, as denoted by the red oval in Fig. 3.8.

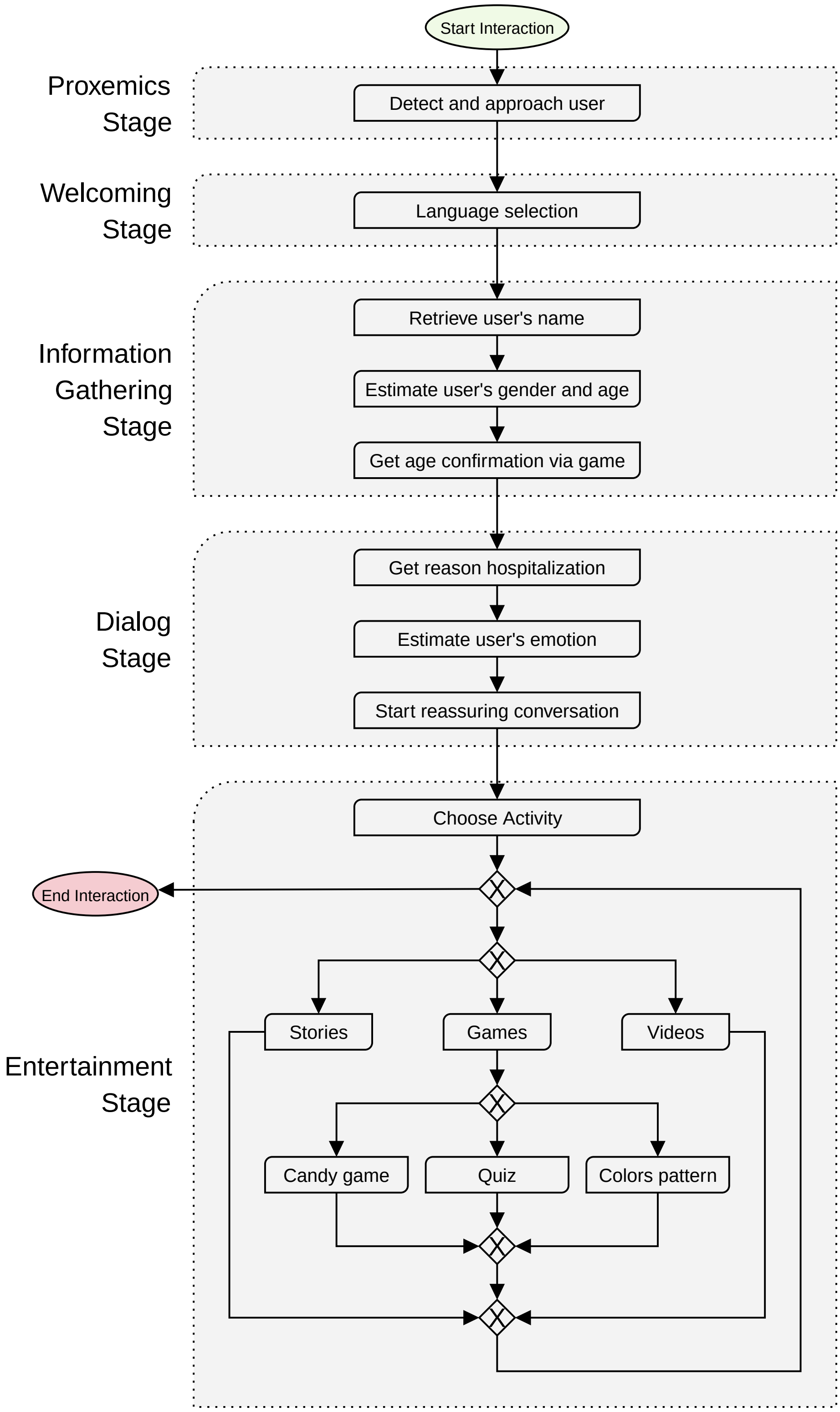


Figure 3.8. Complete diagram of the structure of our solution.

Chapter 4

Implementation Details

In this chapter we will disclose all the technical details involved in our work. In Fig. 4.1 it is possible to see the structure of the code attached to this report where, for the sake of compactness, the innermost files are not depicted.
The elements in the root folder, i.e. *Dr. Pepper* are:

- *index.html*: default layout;
- *welcome_layout.html*: layout used in the welcoming stage;
- *main_layout.html*: layout used in the remaining stages;
- *init*: file to initialize the interaction manager, contains the initial user profile i.e. $< *, *, * >$;
- *qaws.js*: javascript file to setup the MODIM module;
- *actions*: folder containing the MODIM actions used in this project;
- *img*: folder containing the used images;
- *scripts*: folder containing the Python scripts for this work;

The *actions* folder is organized as follows:

- *emotions*: folder containing the MODIM actions used within the emotion retrieval step;
- *menu*: folder containing the MODIM actions used in the activities menu;
- *stories*: folder containing the MODIM actions used in the storytelling step;
- *welcome*: folder containing the MODIM actions used in the welcoming stage;
- *games*: folder containing the MODIM actions used in the gaming step, in particular they are subdivided into:
  - *candy*: for the actions of the candy game;
  - *colorsPattern*: for the actions of the guess the pattern game;
  - *quiz*: for the actions of the quiz game;
- *not_implemented*: generic modim action for not already implemented features.

The *img* folder is organized as follows:

- *language*: folder containing the images for the language selection;
- *menu*: folder containing the images for the categories of the main menu;
- *quiz*: folder containing the images for the quiz game;
- *stories*: folder containing the images for the storytelling functionality, in particular:
  - ◇ *beautybeast*: is a folder containing the images for the Beauty and the Beast story;
  - ◇ *bighero6*: is a folder containing the images for the Big Hero 6 story;
  - ◇ *frozen*: is a folder containing the images for the Frozen story;
- *default_ image.png*: default image of Pepper.

Finally, the *scripts* folder contains:

- *interactions.py*: a Python script containing all the interactions handled by the MODIM interaction manager;
- *main.py*: the main Python script calling the interactions defined in *interactions.py*;
- *postures.py*: a Python script containing the methods for setting a new posture or getting the parameters of already defined postures;
- *routines.py*: a Python script containing the routines for the dialogues or games;
- *utils.py*: a Python script containing utility functions for supporting the implementation.

In order to develop a code able to run both on the simulated and real robot, we have used several tools: in the first place we have used the `NGINX WebServer` as local web server to handle the requests for transferring the web pages. By default the page to load is in `localhost/DrPepper/index.html`. In our work we have used this webserver as to show the content of the Pepper's tablet that in the simulated environment it is accessed through the browser, while in the real one it is shown directly on the real tablet. To have a visible information about the behaviour of Pepper, we have used `Android Studio` to create an Android application that, through a plug-in, enabled a simulated environment in which it was possible to see Pepper moving around the environment, assuming specific postures and displaying the sentences that should have been outputted through the TTS module. Unfortunately, this emulator did not provide all the functionalities usually running on the real robot. Despite this lack of features, the emulator provided also the `NAOQi server` needed for the code to work. We have taken the advantage of using the given `Docker image` in which all the libraries and modules were already available. Among all the libraries used, `pepper tools` stands out since it allowed us to easily interface with Pepper and getting the references of the robot's parameters and services, through `pepper_cmd.py`, and to simulate functionalities otherwise forbidden in the simulator, such as the ASR module and the sonars, thanks to other scripts like `human_say.py`. Finally, we have used the `MODIM server` as the server for the multimodal actions defined to create a multimodal interaction. In particular we have used the `interaction manager (im)` class to call the MODIM actions but also to use the `pepper_tools` methods through the pointer to the `cmd_server` possessed by the `im`.
As mentioned earlier, the emulator that we have employed to face the lack of a real robot, unfortunately, did not provide all the functionalities needed to create a complete work. We have relied on specific Python scripts to emulate human interaction, however was not sufficient to overcome all the disabled functionalities. In particular in this work during the information

gathering stage, as stated in Chapter 3, Pepper estimates the gender and the age of the child. We have simulated this functionality through the method `get_fake_details()` that randomly generates this part of the user profile. However, to provide a code able to run also on the real robot, we have implemented the function `get_real_details()` in the `utils.py` file. More specifically, from the NAOQi documentation of the `ALFaceCharacteristics`[1] service, it is possible to see that the age, gender and other information, are saved in the memory with the keys `PeoplePerception/Person/<ID>/AgeProperties` and `PeoplePerception/Person/<ID>/GenderProperties`, where the <ID> is the identifier of the child. This data is obtained through the event `PeoplePerception/VisiblePeopleList` of the `ALPeoplePerception`[2] service that returns the list of all the IDs of the persons in the Pepper's field of view. We have instead obtained the ID value by creating a subscriber to the `VisiblePeopleList` event and by assigning the `on_visible_people_change` callback function to the subscriber. Into this callback function we have set the ID as the first element of the input list of IDs. After having closed the subscriber, we have accessed the memory through the corresponding keys to return the tuple [gender, age].
Finally it is worth to mention that, due to the usage of the simulator, it was not possible to insert the already implemented animations, for example inside the MODIM actions. This was due to the fact that the animations are executed by the function `animation` of `pepper_tools` (accessible through the `pepper_cmd` or `im.robot.animation`), that disables the services `ALBackgroundMovement`, `ALBasicAwareness` and `ALSpeakingMovement`. However, the second service is not implemented in the NAOQi server used in this work, therefore an exception is generated and the animation is not executed. To overcome this issue we have implemented the postures, shown in Chapter 3, by overriding the method `setPosture()`, in `postures.py`, in such a way that is possible to set the interpolation time besides the joint angles, usually the only parameter of the vanilla `setPosture()` method, to regulate also the speed of the movement.
In the next chapter we will explain all the ways in which the user profile influences Pepper's behaviour in order to generate customized interactions.

[1] http://doc.aldebaran.com/2-4/naoqi/peopleperception/alfacecharacteristics-api.html#alfacecharacteristics-api.

[2] http://doc.aldebaran.com/2-4/naoqi/peopleperception/alpeopleperception-api.html#PeoplePerception/PeopleList.

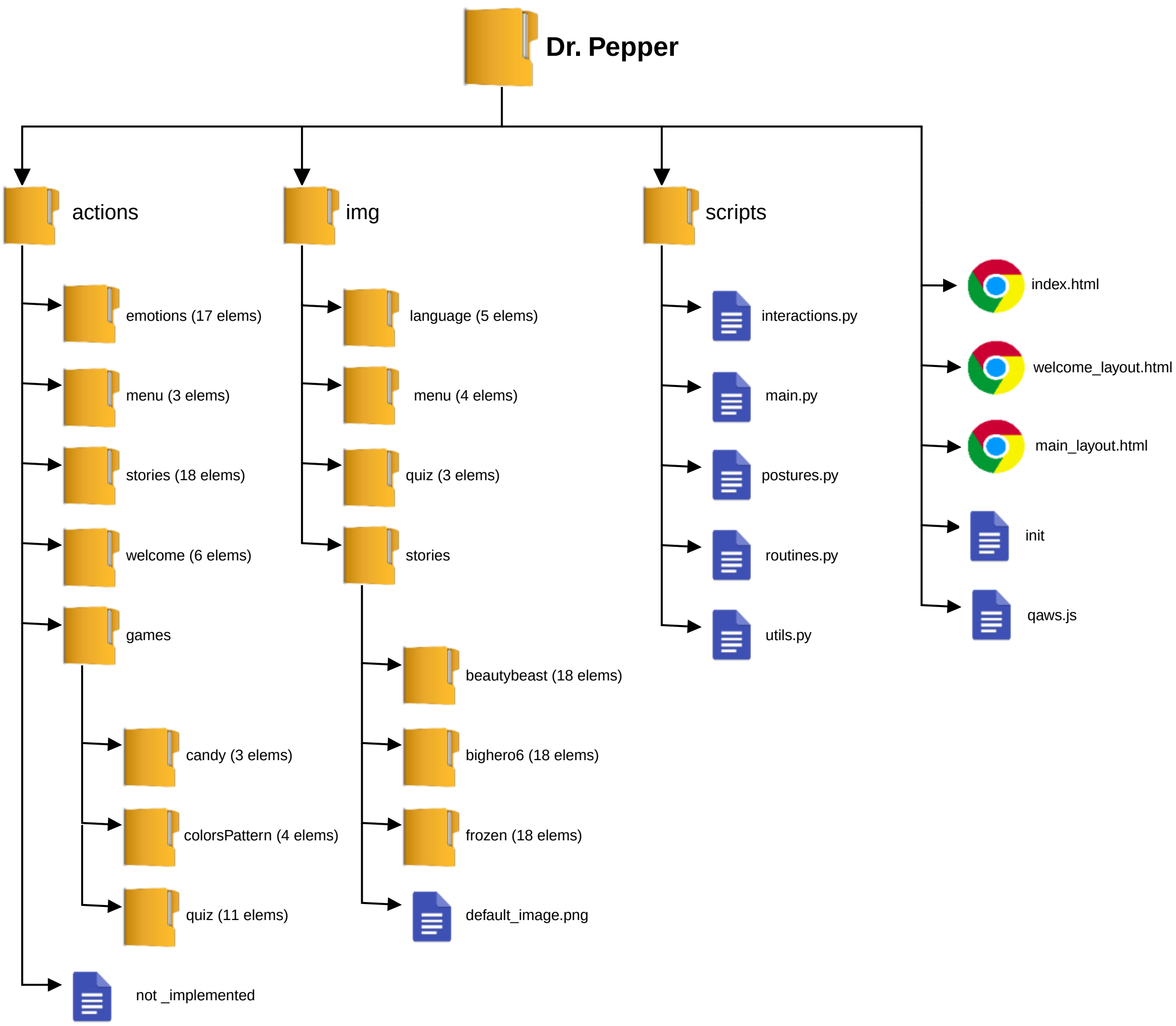


Figure 4.1. Structure of our work. The innermost files are not represented here, but an indication on the number of elements could be found next to their corresponding folder.

Chapter 5

Interaction Results

One of the key features in this work is the level of customization of the interaction. As explained in Chapter 4, we have use MODIM to perform multimodal interaction through actions dependent on a particular *user profile*. The choice of the user profile was fundamental since we wanted to have an high level of customization of the interaction while preventing the code to explode. For this reason we have defined the user profile as a triple <*category, gender, language*> that initially starts with its most general form, that is $<*,*,*>$. The first element of this triple is *category* that represents the range of age of the user. We have identified three groups: s that stands for users of 4, 5 and 6 years old, m for users in the range 7-9 y.o. and finally l for users from 10 to 12 years old. We have selected an overall range of age from 4 to 12 years old since it is the one usually mapped to the pediatric area. The second parameter is *gender* in which the possible values are m for male and f for female. Finally, the last item of the tuple is *language* that represents the language chosen for the interaction among $\{en, it, es, de, fr\}$. The setting of these elements happens during the welcoming, for the language, and information gathering, for the age and gender, stages as described in Chapter 3.

Based on the updated profile, the interaction within different stages of the solution takes different forms. The parameter that mainly changes the course of the interaction is the language, in fact, based on the value of this parameter, the language that Pepper speaks and that accepts for answers varies. For languages such as Italian that requests for gender agreement, also the gender parameter plays a role in changing the form of the sentences told by Pepper. Aside from the pure conversational aspects, the user profile affected mostly the entertainment stage, in particular regarding the stories and the games that Pepper may offer. In order to build the most representative catalogue possible for the stories, we have interviewed several children within the chosen age range to discover their favourite stories. This step leaded to the choice of two different sets for each gender and we have taken the most proposed story for each age category. Therefore our project contains the storytelling ability for *Frozen* for $< s, f, * >$, *The Little Mermaid* for $< m, f, * >$ and *The Beauty and the Beast* for $< l, f, * >$. In addition, Pepper may tell about *Cars* for $< s, m, * >$, *The Incredibles* for $< m, m, * >$ and finally *Big Hero 6* for $< l, m, * >$. At the time of the release of this project we have automatically forced the redirection of the user to the story best compliant with the current user profile, however the structure of the catalogue should contain all the titles but using the information of the user profile to order those stories putting the most suitable ones first, therefore without removing the possibility of choosing of the user. The structure that we have implemented for the storytelling ability of Pepper, allows also to easily insert additional stories besides to the ones already implemented. Each story is composed by 18 different modim actions in order to allow the user to interrupt at any time the story without have to wait until the end, through voice or by pressing the `Stop` button on the tablet. Another reason for this subdivision relies on the synchronization among the Pepper's voice, the text appearing on the screen and the

corresponding illustration.
Other sources of customization of the interaction rely on the implemented games. The first game, i.e. candy game, simply asks the user to choose one of the two hands to guess in which one a virtual candy was hidden. Since the game is usually driven by chance, the user may encounter a series of defeats that could make him/her upset or angry. Since our project is aimed at inducing positive feelings rather than generating negative ones, we augmented the chance of winning accordingly to the age of the partner. For the sake of precision, for each game, its state was determined though the random generation of a value up to one and diverse fixed thresholds for each category of users: the younger the higher[1]. If the generated value was lower than the corresponding threshold then the game is a win for the child. As regards the quiz game instead, another type of customization took place: instead of modifying the chance of winning we have calibrated the 21 total questions according to each age category, e.g. the ones for the youngest players are made with the idea of dealing with their inability to read, therefore using images, Pepper TTS module and numbers in place of textual information. Finally, the customization of the guess the pattern game, tackled different parameters of the game. The speed with which the colors of the sequence show up in Pepper's eyes is regulated by the category of the user: the youngest the slowest. Another parameter that varies is the length of the sequence to be remembered: three colors for $< s, *, * >$, four for $< m, *, * >$ and five for $< l, *, * >$.
Other features of this work, however not directly dependent on the user profile, are the eyes color and the voice characteristics. Their importance in the formation of social bonds was well covered in Chapter 2. In our work we have mapped hot colors to intense emotions and cold colors for more balanced ones, that Pepper exhibits when the context suggests a particular mood, e.g. *happy* after the child has won a game, or because influenced by the child behaviour to simulate empathy, e.g. *sad* when the child discloses a negative feeling. In particular, we have mapped red to anger, yellow to happiness, blue to calm emotions and green as default eye color. We have applied the same reasoning also on voice characteristics, by setting an higher volume and higher speed to intense emotions and lower volume and speed for the opposite case. However we have also mitigated the effects of these changes in contrast to what was done in [2], whose high fluctuations in voice, as example, strongly affected its understanding. This was the right trade-off between using [2] ideas to make the robot showing empathy, while using the highlighted negative findings to prevent our project from suffering of the same drawbacks.
The way in which the functionalities were implemented was designed in such a way to not only allow future developments, such as the addition of others stories and games, but also to insert additional functionalities and services with ease.
In the next chapter we will draw the conclusion of this work.

[1] Specifically, the s category was the most advantaged having a 0.75 as value for the threshold, the m one 0.65 and finally the l one 0.55.

Chapter 6

Conclusions

In this work we have addressed the task of developing a social robot to entertain and assist children in pediatric emergency rooms, using Pepper robot by Softbanks. In particular, we have built an interactive system able to respect the distances implied by proxemics studies to set up a correct base for starting an interaction. A conversational stage allowed the formation of a social relationship between the child and the robot as to become a channel for conveying positive feelings to contrast the negative environment usually presents in this context. Before entertain the child we have also performed a reassuring step through the robot to answer to any doubt or fear the child may had that s/he could not reveal to any other member of the staff of the hospital. In addition, the child sees Pepper as an active member of such staff, therefore accepting the motivations or facts that Pepper tell him/her with more confidence, relying on its authority. We offered an entertaining stage in which Pepper could tell a story and play some games. We have considered the frequent change-of-idea that children exhibit, integrating an escape command to exit the current interaction, to change it or to end it definitively. In this work we have considered a short-time interaction to mirror which is the real situation in which this project may be applied, in contrast to a long-time one that is usually required with already hospitalized children that have to face long-inpatient cures. We have developed a system able to work in an international hospital, therefore allowing the child to choose a different language among a group of five. Another characteristic of our work is the customization of the interactions: through an user profile containing information about the child's age and gender, we have adapted the games and storytelling ability to taking into account the physical and intellectual limits that the age implies. To overcome some of these limitations, while providing a more effective interaction, we have built this project upon the concept of multi-modality, hence to express information in a manifold of natures. This project was particular interesting since it has taught us the difficulties behind the creation of a simple autonomous system that has to cope with the diversity of the human kind. Moreover, try to encode all the information and behaviours that we normally express in a non-verbal way, such as postures, involuntary face expressions and sounds was very difficult. Different studies tried to encapsulate this set of features, however, even with the most accurate information we would have to deal anyway with the limitation of the mechanical system of Pepper. Even though this project provides a quite stable starting point, it might be further extended by implementing a chat bot system able to offer a grater flexibility in the dialogues through all the interaction. Other improvements could be the addition of more entertainment functionalities, e.g. more games or stories or even new services such as a video streaming platform. Finally, it will be worth to add also a conversational part aimed at informing the child about the reasons behind his/her hospitalization and the medical procedures s/he will face in order to reduce negative expectations. All these improvements should strongly contribute in creating an important resource to enhance the quality of the stay of the child in the hospital and even to augment the efficacy of the cures.

References

[1] M. Alemi, A. Meghdari, A. Ghanbarzadeh, L. J. Moghadam and A. Ghanbarzadeh. *Effect of utilizing a humanoid robot as a therapy-assistant in reducing anger, anxiety, and depression.* 2014 Second RSI/ISM International Conference on Robotics and Mechatronics (ICRoM), Tehran, 2014, pp. 748-753.

[2] M. Tielman, M. Neerincx, J. Meyer and R. Looije. *Adaptive Emotional Expression in Robot-Child Interaction.* 2014 9th ACM/IEEE International Conference on Human-Robot Interaction (HRI), Bielefeld, 2014, pp. 407-414.

[3] A. Vinciarelli, M. Pantic, H. Bourlard, A. Pentland. *Social signal processing: state-of-the-art and future perspectives of an emerging domain.* In Proceedings of the 16th ACM international conference on Multimedia, 2008.

[4] L. Iocchi, M.T. Lázaro, L. Jeanpierre, AI. Mouaddib. *Personalized Short-Term Multi-modal Interaction for Social Robots Assisting Users in Shopping Malls.* In: Tapus A., André E., Martin JC., Ferland F., Ammi M. (eds) Social Robotics. ICSR 2015. Lecture Notes in Computer Science, vol 9388. Springer, Cham.