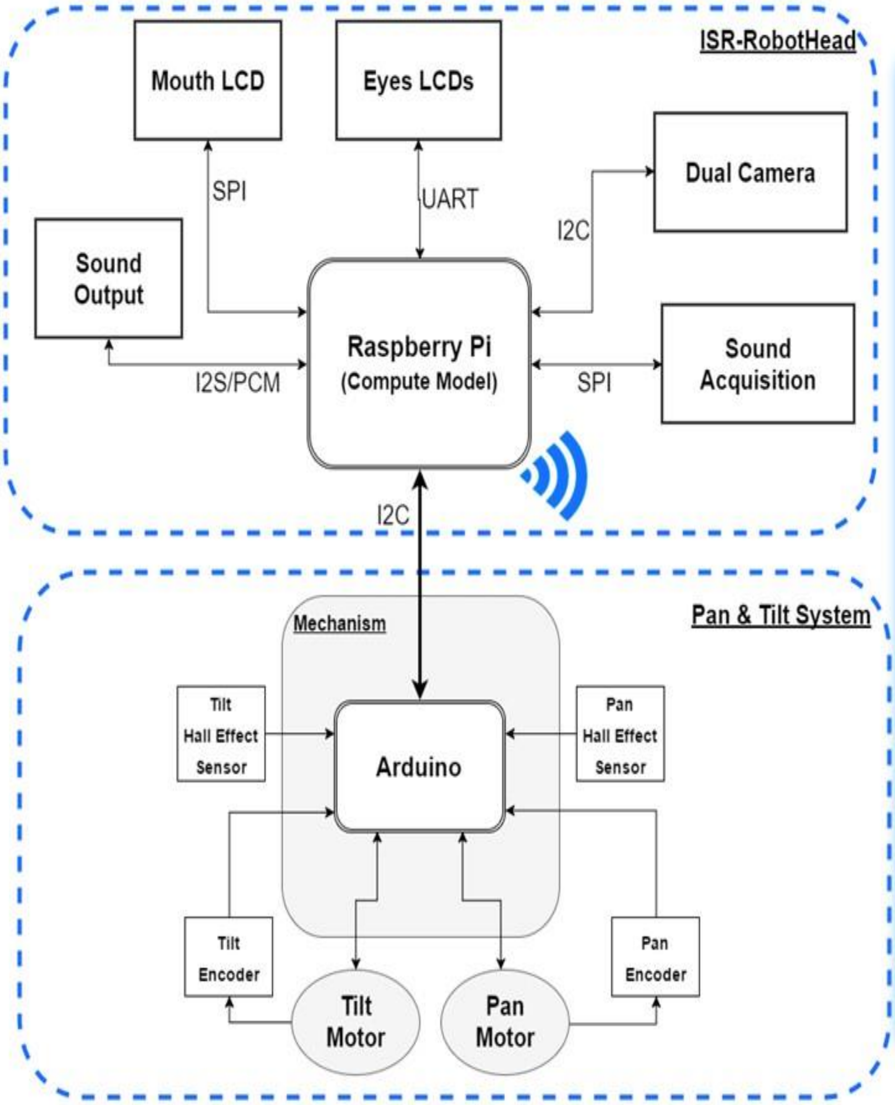


Ricardo Loureiro

Development of a Robotic Head to Express Human Emotions

Dissertação de Mestrado em Engenharia Eletrotécnica e de Computadores
02/2017

· U C ·

Universidade de Coimbra

FCTUC

University of Coimbra

Faculty of Sciences and Technology

Department of Electrical and Computer Engineering

Development of a Robotic Head to Express Emotions and Associated Motions

Ricardo Gomes Loureiro

Supervisor:

Prof. Doctor Urbano Jose Carreira Nunes

Jury:

Prof. Doctor Urbano Jose Carreira Nunes

Prof. Doctor Gabriel Falcão Paiva Fernandes

Prof. Doctor Gabriel Pereira Pires

Dissertation submitted to the Electrical and Computer Engineering Department of the Faculty of Science and Technology of the University of Coimbra in partial fulfilment of the requirements for the Degree of Master of Science.

Coimbra, February of 2017

Acknowledgements

I would first like to thank my dissertation advisor Prof. Urbano Nunes of DEEC at University of Coimbra, he consistently allowed this dissertation to be my own work, but steered me in the right direction whenever he thought I needed it.

I would like to thank all the people involved in the ISR-RobotHead project. To Daniel Almeida for all the hardware support and aid in the assembly, André Lopes who made and design and printed the ISR-RobotHead structure, and a especial thanks to my friend, Leila Gonçalves for the aid in designing the expressions for the final product demonstration.

I would also like to thank Doctor Carlos Carona that organized the experiment with Cerebral Palsy Association of Coimbra and at a private clinical centre for children and youths (Psikontacto, Lda.) in order to validate this project.

I would also like to thank ISR for hosting me, providing the necessary resources, conditions and personnel that allowed me to accomplish all the goals I worked for. This work has been supported by Fundação para a Ciência e Tecnologia (FCT), COMPETE under the project "AMS-HMI2012- Assisted Mobility Supported - RECI/EEIAUT/0181/2012 and UID/EEA/00048/2013, with FEDER funding, programs QREN and COMPETE.

Finally, I must express my very profound gratitude to my family for providing me with unfailing support and continuous encouragement throughout my years of study and through the process of researching and writing this thesis. This accomplishment would not have been possible without them.

Thank you, Ricardo Loureiro

List of acronyms

API Application Programming Interface.

RPiCM Raspberry-Pi Compute Module.

CMDK Compute Module Development Kit.

CSI Camera Serial Interfaces.

DSI Display Serial Interfaces.

DAC Digital-to-Analogue Converter.

GPIO General Purpose Input/Output.

HDMI High Definition Multimedia Interface.

I2C Inter-Integrated Circuit.

LCD Liquid crystal display.

PCB Printed Circuit Board.

PI Proportional Integrative.

RAM Random Access Memory.

RH Robot Head.

RPiCM Raspberry-Pi Compute Module.

SPI Serial Peripheral Interface.

SSH Secure Shell.

TFT Thin Film Transistor.

USB Universal Serial Bus.

VGA Video Graphics Array.

Abstract

Worldwide, several research works have been devoted to the development of humanoid robots for human-robot interaction (HRI). Particular attention has been paid to the concept, design and applications of robotic heads and the face-expressions associated to the robot head. Together, the head and its facial expressions make up a fundamental, and "natural", interface to the end-users; which is particularly true in child-robot interaction (CRI). This dissertation contributes with a functional prototype of a robot headand its LCD-based facial elements: two eyes and eyebrows, and its mouth. This work falls within the field of “social robotics”, or social-interactive robots, a multidisciplinary research area involving sensory perception, psychology, bio-inspired systems, bioengineering, mechatronics, artificial intelligent, and so on. In terms of applications, we can highlight the use of social-robots as guiders, teachers or therapeutic caregivers.

In such cases, the robot receives inputs (such as gesture and speech recognition) and present feedback in a transparent interface that regular people can interpret and capture the attention from the observers: this is possible by employing emotional/facial expressions that a human (child, adult or elder) can easily interpret and then establishing a clear communication and interaction between the human and the robot. Regarding the potential solutions to be used in a robotic head to transmit an emotional feedback to the end-user, the most commons are: LED interactive, kinematic, animatronic and LCD-based robotic head. LCD interactive robotic heads, like the one presented in this dissertation, represent faces mostly in tablet like screens and can achieve a big array of human expressions. Our solution uses LCDs for eyes-eyebrows and mouth animations.

Thoughout this work it is presented the technical solutions employed to achieve a prototype of a working robotic head with the ability com convey emotional expressions.This solution was implemented in a Raspberry-Pi Compute Model micro-computer, which functions as the main controller of this robot. In order to validate this robotic head, experiments were performed with children to verify if this solution is capable of transmitting emotional expressions. Allong with the experimental results, it is also presented the computational and memory encumbrances uppon the micro-computer which enlights the vialibilty of this robotic head solution.

In addition, in order to validate the solution's ability to convey emotions, it was conducted an experiment with children where each subject classified the facial expressions displayed in the robotic head.

Keywords: **Child-Robot Interaction, LCD-based Robotic Head, Facial Expressions, Raspberry-Pi Compute Module**

Resumo

No mundo, vários trabalhos de investigação têm-se focado no desenvolvimento de robots humanoides para interação homem-robot (HRI). Uma especial atenção tem sido dada ao conceito, design e aplicações de cabeças robóticas e expressões faciais associadas à cabeça do robot. Juntos, a cabeça e as suas expressões faciais constituem a interface *natural* para o utilizador; o que é particularmente verdade em interação criança-robot (CRI). Esta dissertação contribui com um protótipo funcional de uma cabeça robótica e os seus elementos faciais baseados em LCDs: dois olhos, sobrancelhas e boca. Este trabalho enquadra-se no campo de "robots sociais", ou robots social-interativos, uma área de investigação multidisciplinar que envolve, percepção sensorial, psicologia, sistemas bio-inspirados, bioengenharia, mecatrônica, inteligencia artificial, entre outros. Em termos de aplicações, podemos realçar o uso de robots sociais como guias, educadores ou auxiliares terapêuticos.

Nestes casos, o robot recebe sinais (como gestos ou reconhecimento vocal) e reage numa interface transparente que pessoas regulares consigam interpretar e capturar a atenção dos observadores: isto é possível empregando expressões emocionais/faciais que uma pessoa (criança, adulto ou idoso) consiga interpretar facilmente e de seguida estabelecer uma clara comunicação e interação entre humano e robot. A respeito de possíveis soluções para uma cabeça robótica ser capaz de transmitir uma resposta emocional para o utilizador final, as mais comuns são: interativas com LEDs. cinemáticas, animatrônicas e baseadas em LCDS. Cabeças robóticas baseadas em LCDs, como a solução presente nesta dissertação, representam caras maioritariamente em ecrãs semelhantes a *tablets* e conseguem obter várias representações para diversas expressões humanas. A nossa solução, usa LCDs para fazer animações dos olhos, sobrancelhas e boca.

Ao longo deste trabalho é apresentado as soluções técnicas empregadas para alcançar um protótipo de uma cabeça robótica funcional com a capacidade de transmitir expressões emocionais. Esta solução foi implementada num micro-computador Raspberry-Pi Compute Module (RPiCM), o qual funciona como o principal controlador do robot. De modo a validar esta cabeça robótica, foram efetuadas experiências com crianças para verificar se esta solução é capaz de transmitir expressões emocionais. Acrescentando aos resultados experimentais para validação do protótipo é também apresentado os pesos computacional e de memoria do micro-computador com o objectivo de clarificar a viabilidade desta cabeça robótica.

Palavras-chave: **Interação criança-robot, Cabeça robótica baseada em LCD, Expressões faciais, Raspberry-Pi Compute Module**

Contents

List of Figures

List of Tables

Chapter 1

Introduction

Nowadays, human-robot interaction (HRI) is a grown discipline with applications in industrial and domestic setting. Social robots have been built on an appealing way, with uncommon face design but graceful at the same time to capture the interest of the person with whom is interacting [8]. When a person engages on a social interaction with another the emotional response is a key factor. It can represent acceptance, rejection, happiness, aggression, sadness and other kinds of emotions. As a result, the design and interaction of a social robot should be able to emulate emotions which humans are familiar with. With this in mind, worldwide several research projects focus on the development of robots (eg. humanoid robots) for human-robot interaction (HRI). Especially for the head design, there is still a discussion if it should look like a human head or be more toy-like. Therefore, the development of socially anthropomorphic robots has taken many solutions in the scientific community in order to tackle a set of key factors, such as:

- How the robot would actually communicate [8]?
- Is there a target demographic group (children, adults) [9]?
- Should the design be more human-like or toy-like [17]?

These become important factors to take into account in the design and development of social robots that can interact with humans. However, for this interaction to occur the robot should be capable of transmitting messages involving verbal and non-verbal communications, where the first is made through voice and the second made by the expressions directly associated with emotions for the purpose of emulating human interaction.

1.1 Motivation

The great majority of robotic heads express emotions using series of mechanic actuators for the eyes and mouth (*e.g.* ROMAN [17]), others using a liquid crystal displays (LCD) to emulate all the face features (*e.g.* BAXTER [23], Buddy robot [24]) and there is still those that use LEDs for the same goal (*e.g.* NAO [25], Pepper [26]). Each solution has its vantages and disadvantages normally as regarding expressiveness and complexity. With this dissertation, it was intended to develop the expressive features of a robotic head to be used in research applications for children-robot interaction (CRI). A new robotic head with distinct LCD-based eyes and mouth to convey emotional expressions for CRI is described.

1.2 Objectives

The main objective of this dissertation is to present a proof of concept for a new robotic head solution to operate in a CRI environment. In order to achieve this goal, the ISR-RobotHead recreates expressions that children can associate to emotions and it is equipped with sensors and actuators for interaction purposes. To that end, the ISR-RobotHead is composed by different modules in order to distribute the computation requirements through the different hardware. Thus the main objectives encompass the following developments:

- Software required to implement the LCD-based eyes and mouths visual primitives.
- Orientation ISR-RobotHead control system.
- LCD-based visual expressions.
- Evaluation of the system regarding CPU and RAM occupancy.

1.3 Implementations and Main Contributions

In this dissertation, we report the achieved results concerning the design and implementation of expressions on a robotic head for CRI studies. As it was previously mentioned, a robotic head has different kind of solutions with degrees of complexity. With this work we present a new proof of concept where the employed techniques simplify the expression response on a development point of view and, at the same time, a person can identify the expressions displayed. For such purpose an LCD-based solution of adopted, it was used LCDs to display eyes and mouth's animations for the emotional expressions. It was developed a library in *C* to interact with each display module and to control the ISR-RobotHead's orientation. The dissertation is organized as follows:

RobotHeads and Human Emotional Expressions (Chapter 2)

- Presentation of different robotic heads' solutions
- Human perspective on expression representation

Robot Head Modules - Hardware Overview (see Fig. 1.1) (Chapter 3)

- Use of a low-cost Micro controller (Raspberry Pi Compute Module) with the capability to control multiple peripherals.
- Development of a robotic head with the ability to express emotions.
- Modular architecture, where each module can operate independently.
- General view of every module with special attention to the expressive.

Results and Benchmark (Chapter 4)

- Visual results of the ISR-RobotHead expressions.
- Experimental validation with children.
- Computational performance on the Raspberry-Pi micro computer to operate the different modules.

In Chapter 5 conclusions are drawn and guidelines for future work are provided.

Figure 1.1 represents the overall system's modules.

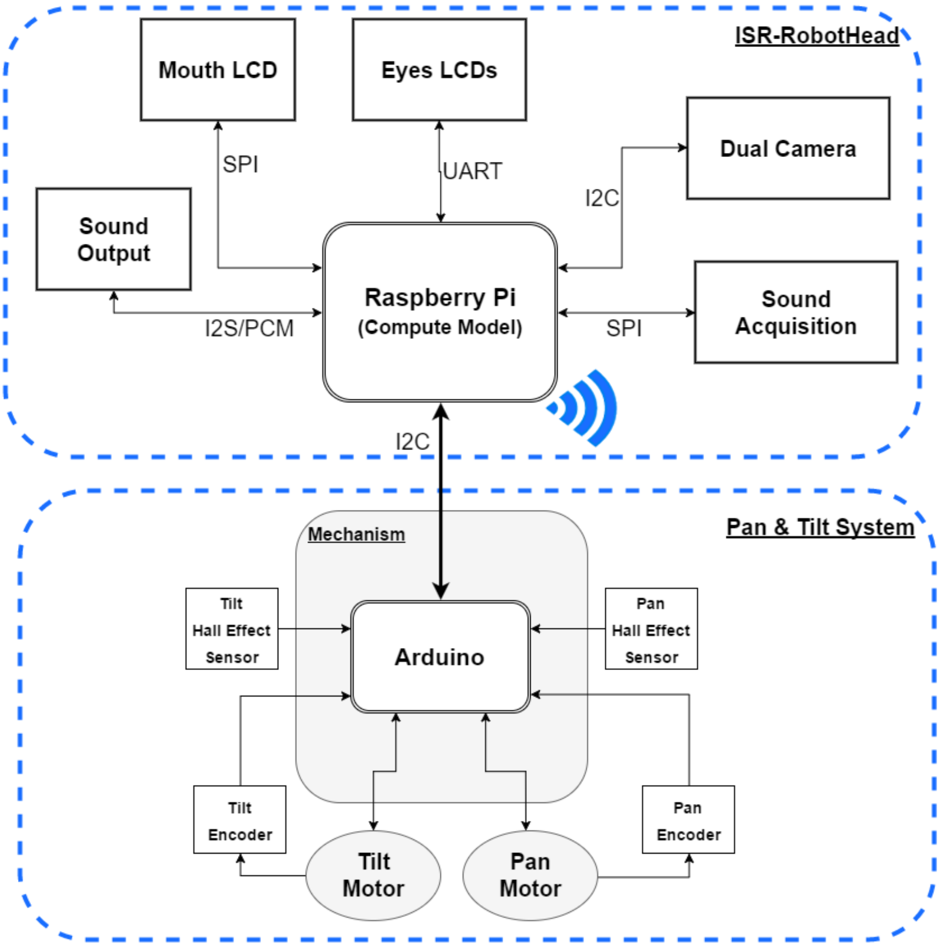


Figure 1.1 ISR-RobotHead modules overview.

Chapter 2

Robot Heads and Emotional-like Expressions

This chapter addresses existing robotic heads with the ability to emulate emotional expressions, its relevance and applications at HRI context. Moreover a review of studies that classify human emotions through facial features, as well as, some robotic heads with the ability to be display emotion-like expressions.

2.1 Robotic Heads that are able to Express Emotions

Social robots can be used for a variety of purposes: as research platforms, as toys, educational tools, or as therapeutic caregivers.

Most researchers design robots as assistants, companions, or pets in addition to the more traditional role of servants with interactive features to augment its functionally and interest in the eyes of the user. To that end, in order for a person to interact with a robot in a believable manner, the robot's face functions as a "natural" interface by employing human-like social cues and communication modalities, being this given by sounds or facial expressions from the robot.

However a person may not perceive or be interested on a robot depending on its design. M. Mori *et al.* [19] present a relation between anthropomorphic robots and human acceptability that is commonly denoted as the uncanny valley (see Fig. 2.1).

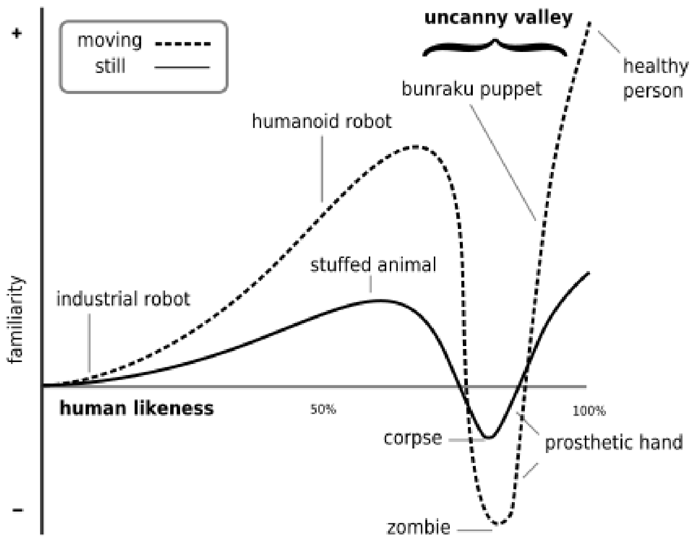


Figure 2.1 Hypothesised by M. Mori *et al.* [19] between robot anthropomorphism and human acceptability.

The aspect of interest is that anthropomorphic robots, which resemble a person, create a certain uneasiness in humans, especially in children, since the resemblance to a human creates confusion on a person viewing the robot. Whereas children seem to prefer toy-like robots with human-like characteristics. This idea is reinforced by S. Woods *et al.* [28], where the researchers worked with children to understand what is their view on robot design, from which the research suggest that a robot for children should be cartoon like, with a mixture of human-machine features.

2.1.1 Interactive Robotic Heads Based on LEDs

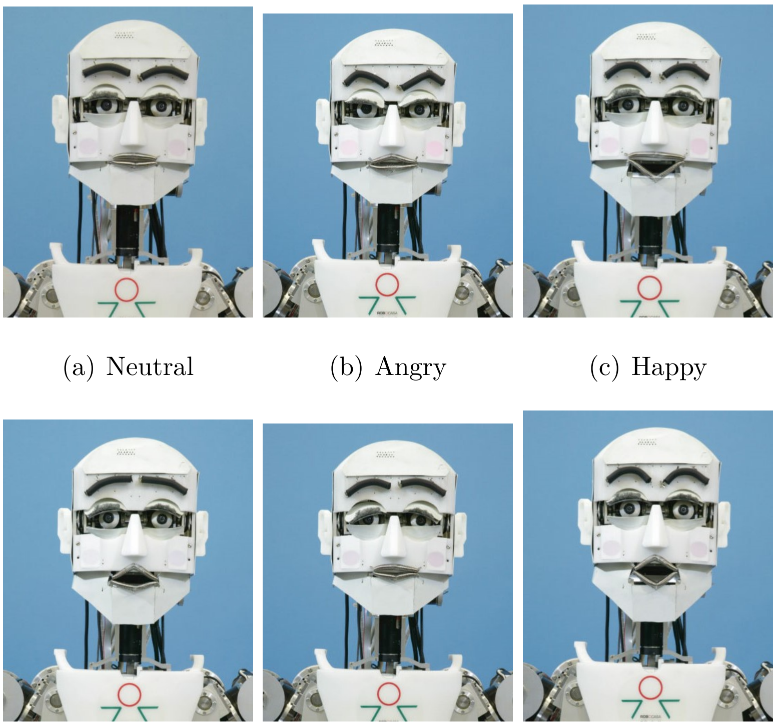
Despite its simplicity, the LEDs are capable of displaying a wide range of colours and are very cost-effective. In addition, the robot heads have embedded cameras, microphones and speakers to read environmental information and to transmit sounds. LED-based robotic heads (see Fig. 2.2) have advantages and disadvantages regarding expressiveness [6], but the provided enough features to keep the human attention. One good example is the NAO robot. It mainly expresses emotions by gestures and coloured LED eyes, but due to its white flat and inanimate face it cannot express facial expressions.

Figure 2.2 LED based robotic heads. Taken from [?]

2.1.2 Kinematic Robotic Heads

Kinematic heads are animatronic heads developed to provide a communication with humans. These solutions are equipped with mechanical actuators to create eye and mouth movements. However, solutions in these category have a high degree of technical complexity since this heads are equipped with a variety of moving parts to replicate face movements and expressions.

Robot Kobian (see Fig. 2.3) is a good example of a kinematic robotic head, developed by Kazuko Itoh *et al* [16].



(a) Neutral (b) Angry (c) Happy

(d) Frighten (e) Sad (f) Surprised

Figure 2.3 Robot Kobian facial expressions. Taken from [**?**]

The robot can six common emotions through facial expressions (shown in Fig. 2.3), such as neutral, anger, happiness, fear, sadness and surprise.

2.1.3 Animatronic Robotic Heads with Flexible Skin

The robot heads with the most human-lie features are animatronic robot heads with artificial and flexible skin. Similarly to the kinematic robot heads, these robots recreate human expressions through the use of actuators that recreate muscles and tendons, which give them a close resemble to humans. One example of these heads is the robot ROMAN) as it is presented in Fig. 2.4.

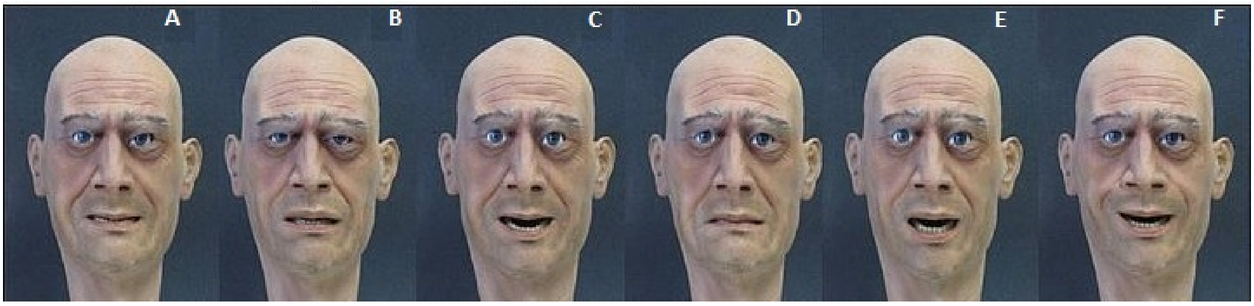


Figure 2.4 Robot ROMAN facial expressions [17]. The expressions' correspondence are as follows: A = anger, B = disgust, C = fear, D = sadness, E = surprise and F = happiness. Taken from [1]

However, according to the theory of the uncanny valley, these faces may be uncomfortable for some persons since they may look similar to a human but at the same time they clearly are not, and this gap of almost human may create a certain discomfort to the person interacting with it. Another disadvantage is the development process since these heads require complex systems and special materials to come to fruition.

2.1.4 LCD-based Robotic Heads to Display Expressions

Another technical solution for robots to convey emotions are LCD-based robotic heads. These robotic heads use liquid-crystal display (LCD) screen in each eye and mouth to reproduce facial expressions. One example of such robots its the Buddy (see Fig. 2.5) [24], it can reproduce expressions that enrich interaction with users and capture their attention.

Figure 2.5 Buddy Robot. Developed by *BlueFrog Robotics*

This robot of a screen looking like a tablet, with cartoon-like eyes and eyebrows to change expressions for intuitive messages to nearby users.

2.2 Emotional Expressions

Most research involving human-robot interaction employs robots to be used as assistants, companions or pets to interact with a target group. For example, children with autism tend to exhibit difficulty in interacting with other humans. With this in mind, researchers have been exposing children, with this impediment, to interactive humanoid robots and results suggest that these children develop some social skills with the increase exposure to such robots [22].

Another application worthy of mention was an experiment where the researchers have used a robot to give a science class to children [12]. The research suggested that the use of robots can engage students in active learning, keep attention to subject and attract interest in the lesson.

Overall, research works in child-robot interaction (CRI) still require further development and experimentation to validate robots effectiveness in children development.

For a robotic head, it is not enough to just create emotional expressions like a human being, but also to provide interaction skills with the observer in an appealing way, so people can feel empathy with the robotic head. In an experiment performed by Bruce *et al.* [5], a robot tries to engage persons to answer a set of questions by tracking them and then responding with emotional expressions regarding the persons' willingness or unwillingness during the interaction with the robot. The results suggest that, between the facial expressions and the movement tracking, persons' interest towards the robot is the same. However, by combining both features the persons were more willing to stop and interact with the robot, which enforces the idea that emotional representation in a robot is important to enhance the experience with HRI.

Still in the field of emotional expressions, Paul Ekman [7] tried to identify if persons from different cultures and backgrounds were able to identify a set of human emotional expressions and, additionally if each expression was characterized by the same emotion regarding each person's interpretation. Pictures of persons performing emotions such as happiness, sadness, surprise, fear, disgust and anger were shown to each subject that participated in this experiment, and results confirm the hypothesis that each of these six basic expressions can be identified by different persons from different cultures and backgrounds. These expressions were also implemented in some robotic faces to convey emotion for HRI [17, 16]. These emotions were also implemented in the ISR-RobotHead for experimental evaluation.

Chapter 3

Robot Head Modules - Hardware Overview

In this chapter we introduce, in detail, each module implemented for the development of the ISR-RobotHead (see Fig. 3.1) and their purposes along with the software developed and restrictions founded throughout this project.

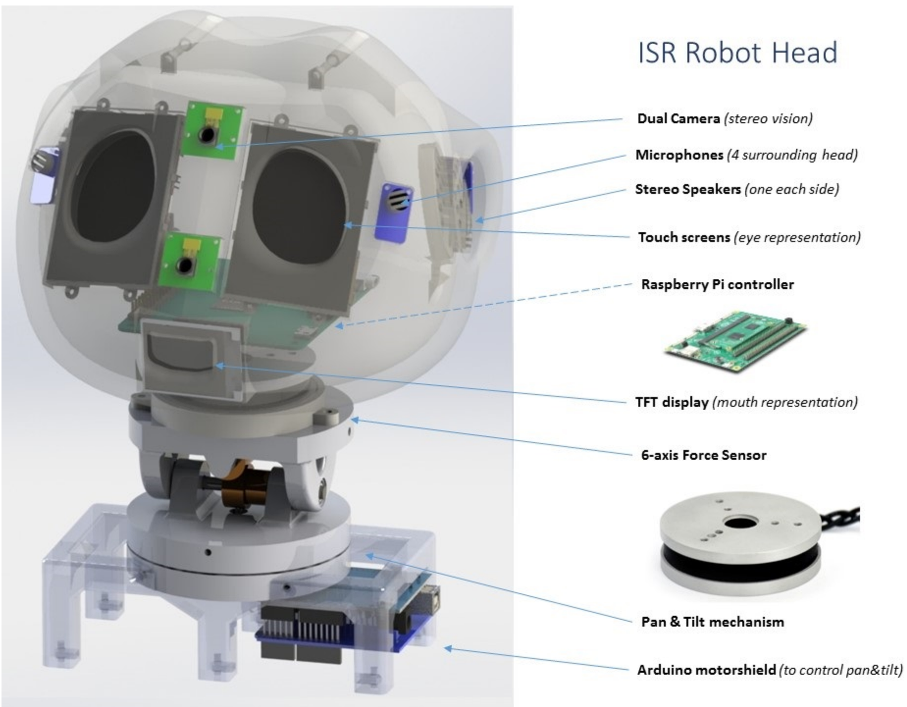


Figure 3.1 Design and Hardware concept of the ISR-Robot Head [3].

The main focus of this dissertation was to develop the interface for the expressions representation (eyes 3.3.1 and mouth 3.4 LCDs) and the mechatronic system for the head orientation 3.5. The sensory and sound modules will also be briefly addressed to validate the ISR-RobotHead. In this way, it was developed libraries in C for the Raspberry-Pi Compute

Module (RPiCM) in order to interact with eyes and mouth LCDs as well as communicate with the Pan and Tilt system, being its controller also projected and implemented. The ISR-RobotHead falls into the LCD based robotic heads since it conveys expressions through the use of LCD screens. This robotic head has also multiple peripherals to enhance the CRI experience, which are: an output sound system, two cameras for stereo vision, Pan-Tilt orientation system and three LCDs to represent facial expressions, two for the eyes and one for the mouth. All this peripherals are controlled by a Raspberry Pi with the exception of the Pan-Tilt, which is controlled by an Arduino [3].

3.1 Raspberry Pi-Compute Module Development Kit

The Raspberry Pi is a small and affordable computer that operates with a Debian based Linux distribution operative system and has a number of GPIO (General Purpose Input/Output) available for interaction with any device or electronic system that is capable of doing so.

3.1.1 Compute Module

3.1.1.1 Hardware Characteristics

For this project, we are going to use a Raspberry-Pi Compute Module (RPiCM) Raspberry instead of a conventional Raspberry. The RPiCM has a BCM2835 processor which operates at a minimum speed of 700 MHz and has a 4 GBytes flash memory opposed to a conventional Raspberry that uses as physical storage an external Micro SD card. In order to interact and program the RPiCM we needed to use the Development Kit Board (see Fig. 3.2).

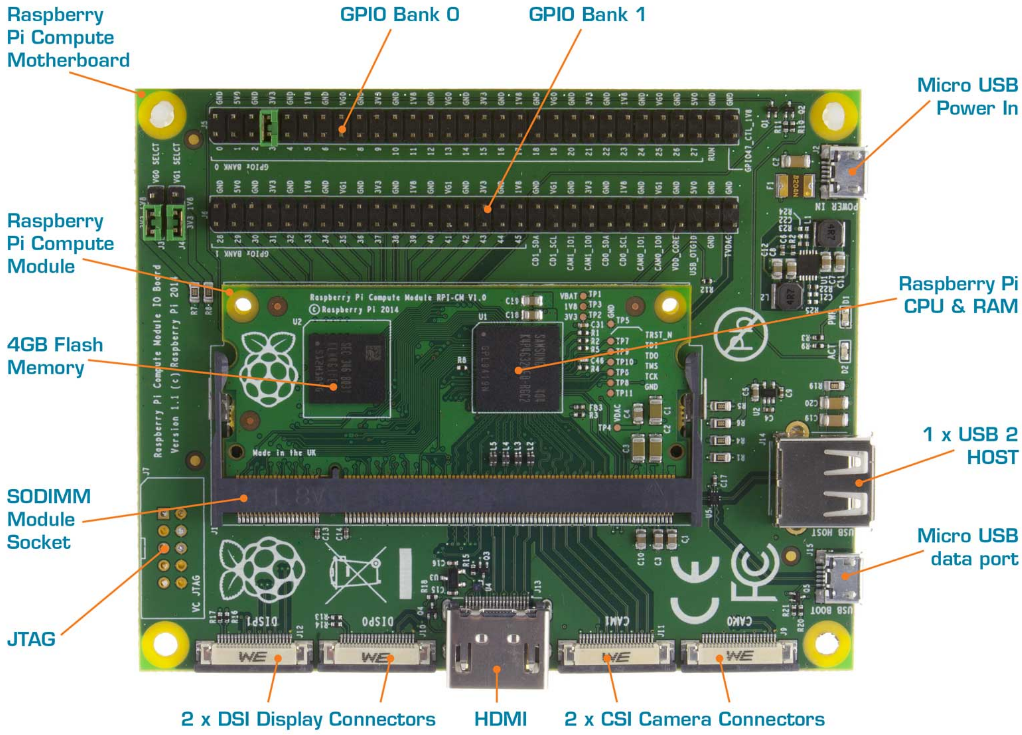


Figure 3.2 Raspberry Pi Compute Module [10] attached to the Development Kit [11].

The Development Kit has as socket in the centre where the RPiCM is placed, two GPIO banks to access all the pins available, two Micro USB ports one for power and other to upload data for the RPiCM, two Camera Serial Interfaces (CSI) and two Display Serial Interfaces (DSI). The GPIO pins are where all the peripherals will be connected, apart from the cameras. Each bank is divided in two stripes, where one is for the GPIO that operates the peripherals and the other for sources (5V, 3.3V, 1.8V and Ground). In addition, this control module composed by a wifi adaptor in order to communicate RPiCM. Furthermore, the communication with the RPiCM was established through secure shell (SSH) protocol. In table 3.1 is a brief overview of the RPiCM and Raspberry Pi 1 Model B+ for comparison purposes, since both have the most similarities in memory and CPU performances.

Table 3.1 Comparison between RPiCM and Raspberry Pi 1 Model B+.

	Compute Module	Raspberry Pi 1 Model B+
CPU	700MHz	700MHz
RAM	512 MBytes	512 Mbytes
USB Ports	1	4
HDD	4Gbytes eMMC Flash	Micro SD Card
Cameras	2	1
GPIO	120	40
Ethernet Port	No	Has

By comparing the two devices it becomes clear that the RPiCM has disadvantages when taking into account the physical memory and the USB and Ethernets ports available. However, taking into account that in the current project we intend to have stereo vision for future work and manipulate multiple peripherals the RPiCM becomes the most appropriate choice. Another feature in favour of the RPiCM is that it can be upgraded by replacing it with a more recent version. Attending to the previous arguments, the features for choosing the RPiCM over the Raspberry Pi are: access to all GPIO available in the Raspberry, which give a great degree of flexibility for development, able to operate a stereo camera system and easy to upgrade the hardware by changing the RPiCM to a more recent version. Attending to the peripheral interfaces the Raspberry Pi has two I^2C, three SPI, one PCM and two UART, which are needed for this project.

Operative System

For this project we use the Raspbian Wheezy operative system. It is a system based on Debian GNU/Linux optimised for the Raspberry Pi hardware. It comes with pre-compiled software and tools for use. Still the current operative system build posed restrictions regarding the peripherals. As was stated previously, each communication protocol has multiple interfaces. However, the default Raspbian build only has software implemented for the UART0, SPI0, I^2C1 peripherals, these restrictions were taken into account during the development of the software for the ISR-RobotHead. In order to use the peripherals and operate the Raspberry's GPIOs it is used the *WiringPi* library [13]. Which is an open source API that was developed with the aim to easily program the Raspberry. It already comes with a set of tools and functions to operate the Raspberry's pins and serial protocols, which were used throughout these project.

3.2 Stereo Cameras

The RPiCM has the capacity to support stereo vision since it has two CSI to plug two Raspberry-Pi cameras. The Raspberry Pi camera module can be used to take high-definition video, as well as stills photographs. In order to use the cameras in this project, we use the software *Raspivid* [21]. It offers a variety of configurations that go from colour saturation, configurable image dimensions and stereo capture. Each camera attaches via a 15cm ribbon cable to the CSI port on the Raspberry Pi Development Kit board.
Besides the ribbon cables the cameras require that their respective pins in the J6 header be connected to specific GPIO pins for the RPiCM to be able to receive data from them.

3.3 Eye Expressions Module

3.3.1 Eyes LCDs

The uLCD-32PTU (see Fig. 3.3) is a compact and cost effective Touch screen Intelligent Display Module controlled by a PICASO processor and capable of being an interface controller for a number of applications [2].

Figure 3.3 LCD used to display the Eyes the RobotHead's eyes [2].

The LCDs have the following characteristics: it has a 240 x 320 pixel resolution with a 16 bit (65k) colour depth, a Serial Interface to operate the module and a Micro-SD card socket for data storage, where the images of the eye expression will be stored. The PICASO processor can be programmed and used in four different environments categorised in Standalone or Slave configurations. Each has two environment to operate the LCD, which are: Designer, ViSi Environments for Standalone Configurations and ViSi Genie, Serial Environments for Slave Configuration. In this project it is going to be used the Serial Environment for the communication between the Raspberry Pi RPiCM's UART0 peripheral and the uLCD-32PTU module. The LCDs for the right and left eyes are connected in parallel to the same UART peripheral since the operative system only has software to operate the UART0, which implies that, with this solution the Raspberry only needs to send one command to operate both LCDs. However, this poses a constraint since the LCDs will not be able to operate individually because the Raspberry can't send different data to a single LCD. The eyes' module is drafted in the Fig. 3.4.

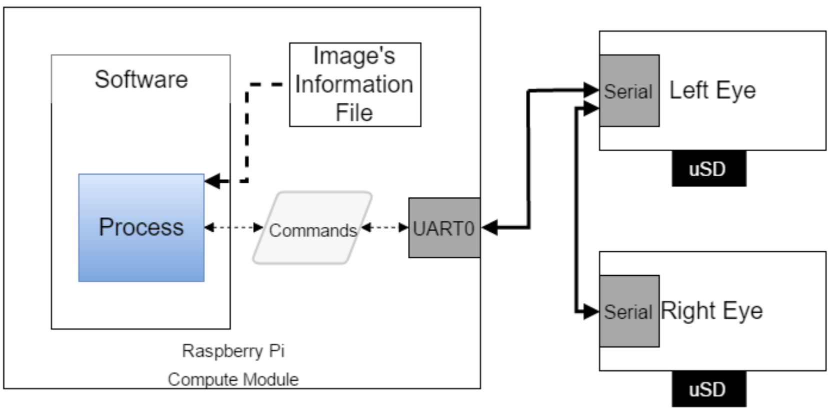


Figure 3.4 General view of the RobotHead's Eyes process.

3.3.2 Serial Protocol

As previously stated in section 3.3.1, the Serial Protocol Environment is considered for the interaction between the RPiCM and the Eyes' LCDs. The Raspberry Pi sends a set of commands with associated parameters through the UART peripheral. Thus, the Eyes' LCD can display computer graphics primitives, text as well as images and video from micro-SD card. It can also receive touch events on the 4D Systems Display Module. The serial environment has 135 commands and operates with a baud rate 9600. Each data transmission, which can be a command or a parameter, sent to the LCD is composed by a 16 bit or 2 byte word. For sake of simplicity, it is illustrated the way how the background colour command in transmitted to LCD.

- Background colour Command: 0xFF 0xA5
- Background colour Parameters: 0xFF 0xFF (colour white)
- Background colour Response: 0x06 (Acknowledge)
- Background colour Response: 2 Byte Word (Previous Background colour)

The number of parameters and responses may change depending on the issued command. The displays are connected in parallel to the Raspberry's UART peripheral, which entails that when a command is sent to both LCDs, it will process and transmit the response to the Raspberry, being it only an acknowledge or any return that the command function may have. Unfortunately, the current configuration of the Raspberry does not allow to send data to a single LCD neither know which LCD responded to a command. This is an important aspect to take into account in the following sections. For the purpose of displaying eye expression, the Raspberry Pi sends the address of an image, already present in the micro-SD, for the module to display the intended picture. However, the images of the right and left eyes must be loaded first into the respective micro-SD, in an appropriate order, and then manually placed in the corresponding LCD to be displayed successfully.

3.3.3 Eyes' Images Upload

To load the images we use the program *4D Systems Graphics Composer*, which gives us the ability to select the images (frames) to be loaded into the LCD and allows to select the order in which the images are uploaded into the micro-SD card, in other words we can easily create a database of pictures that represent eye expressions with multiple frames for each representation. The images are uploaded in two different sessions of the *Graphics Composer* to the respective LCD (illustrated in Fig. 3.5).

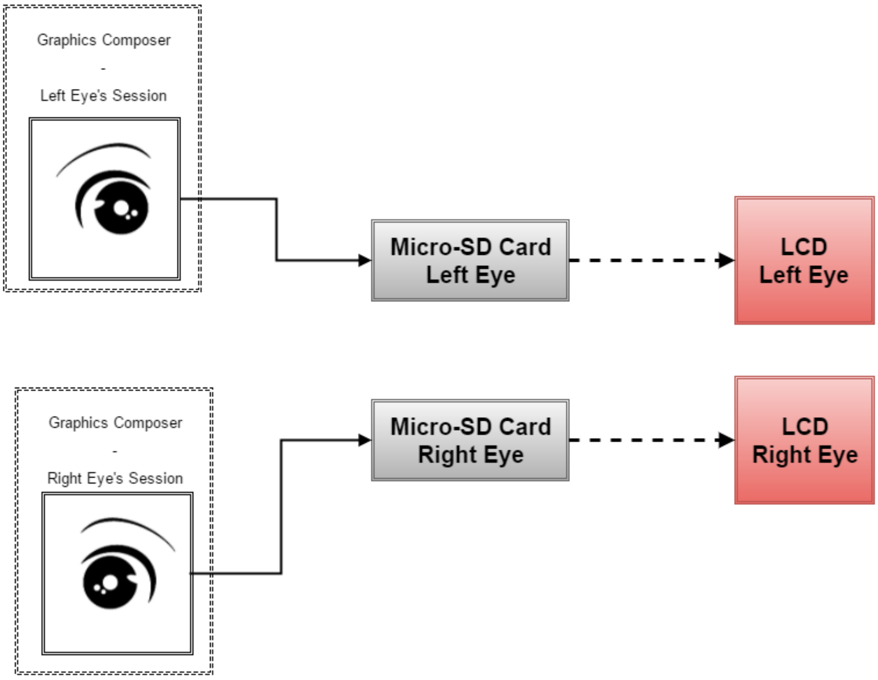


Figure 3.5 Micro SD image storage schematics. The left and right eye respective images are loaded into the respective micro-SD and then the cards are placed in the corresponding display.

This poses few limitations when designing the eyes of the ISR-RobotHead, because the uploaded images can have any valid image format. Which implies that the design can be made on an appropriate software and the images can be elaborated only taking into account the resolution of the LCD (320 x 240 pixels), however the images do not need to have the same size of the LCD's resolution. Despite the design advantages, there are two key technical aspects to display the images correctly.

- Order in which the images are uploaded (see Fig. 3.6)
- Dimensions of a pair of images (height and width)

Both displays are connected to the same peripheral and the images are stored as raw data in blocks of 512 Bytes in the SD's raw partition. In that regard, if the left and right frame of a pair picture are previously upload in different order, the memory block addresses won't coincide and the images displayed will be incorrect. In a like manner, pictures with different dimensions may require a distinct number of blocks to be stored in the micro-SD card which disarranges the images' addresses.

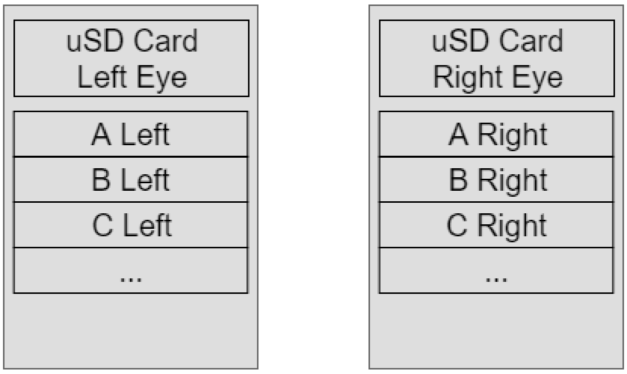


Figure 3.6 Micro SD image storage schematics.

After uploading the images correctly in the micro-SD cards there is still need for knowing the address. This is given by *4D Systems Graphics Composer* in a file as is shown in Fig. 3.7.

```
// Image - "image1.png" Size 188x209
// Width  = 188
// Height = 209
// Usage: image1.png(x, y);
#constant image1.png    $media_SetSector(0x0000, 0x0000);

// Image - "image2.png" Size 188x209
// Width  = 188
// Height = 209
// Usage: image2.png(x, y);
#constant image2.png    $media_SetSector(0x0000, 0x009A);
```

Figure 3.7 Images' addresses

The file has the image name, width, height and the image address in **$media_SetSector**. With this information we can display any pair of expressions that are in the micro-SD cards. However, the information return from *4D Systems Graphics Composer* is not organized in a fashion to have access all the images characteristics and addresses. To solve this we wrote a program to filter all the unnecessary characters and organize the information in lines to be easily read by a process.

3.3.4 Eyes' Library

In the Raspberry Pi Compute Module we developed a library in C, `RHeyesLib.c`, to configure the UART0 peripherals with all the GPIO configurations required, load the image addresses in a data structure in order to display and alternate easily between expression through a logic frame sequence.

3.3.4.1 RHeyesLib Image's Data Structure

Each frame information, loaded from the filtered file, is saved in a global pointer wherein each index is composed by the data structure presented in Fig. 3.8 .

```
struct LCD_rawimg{
  char * iname;
  int Hsec;
  int Lsec;
  int width;
  int height;
};
```

Figure 3.8 Structure to keep image information

The image name is saved in the character pointer **iname**, the most significant address value in the **Hsec**, the lesser significant address value in the **Lsec** and the image width and height in the **width** and **height** integers to centre the image in the intended pixel coordinates. The routine for the RPiCM to load this information is depicted in Algorithm 1.

Algorithm 1 RHeyes_LoadSectors function. Load image information into data structure buffer.

function RHEYES_LOADSECTORS(const char * sectors)
  Open $cleanfile$
  $N \leftarrow$ Count number of entries
  Allocate global structure by sizeof(struct LCD_rawimg) $\times N \leftarrow global\_pointer$
  for $i \leftarrow 0$; i to N **do**
    $global\_pointer[i] \leftarrow$ Get data line from file
  end for
end function

After a process calls this function it puts the RPiCM ready to change the eye expressions in the ISR-RobotHead.

3.3.4.2 RHeyesLib Display Eyes

The following functions were developed for displaying the eye expression's frames. **media_SetSector**: This function receives the high and low words of the micro-SD memory sector and sends the *"Set Sector Address"* command to the LCDs. This command sets the media memory internal address pointer for the specified sector. **media_Image**: This function receives the x and y coordinates. Displays an image from the media storage at the specified coordinates. The image address is previously specified with the *"Set Sector Address"* command. If the image is shown partially off screen, it may not be displayed correctly. This commands are used in the function `RHeyes_ShowImg`. How this functions are employed is illustrated in Fig. 2.

Algorithm 2 RHeyes_ShowImg function. Sends commands to LCDs to display image.

```
function INT RHEYES_SHOWIMG(const char * eye, int x , int y)
    for i ← 0 ; i to N do
        if global_pointer[i].iname == eye then
            media_SetSector(Global_Pointer[i].Hsec , Global_Pointer[i].Lsec)
            Ximg_center = X_arg + X0 − img_Width / 2
            Yimg_center = Y_arg + Y0 − img_Height / 2
            media_Image(Ximg_center , Yimg_center)
            return i
        end if
    end for
    return −1
end function
```

The function receives as arguments the image name that it is intended to display, **const char * eye** and the pixel's x and y coordinates for the image to be displayed,**int x** and **int y**. The algorithm cycles through the global pointer indexes until it finds an image with the same name as the **eye** argument. Then sends the *media_SetSector* command with the arguments **Hsec** and **Lsec**. After the sector address pointer in the LCD is set, the command *media_Image* is sent with the arguments $Ximg_{center}$ and $Yimg_{center}$ which are the LCDs' pixel coordinates where the images will be placed. After all the commands are sent the function returns the image index in the global pointer. In case no image is found, returns -1. Regarding the *media_Image* command's $Ximg_{center}$ and $Yimg_{center}$ arguments the mathematical operation 3.1,3.2.

$$Ximg_{center} = X_{arg} + X0 - \frac{img_{Width}}{2} \tag{3.1}$$

$$Yimg_{center} = Y_{arg} + Y0 - \frac{img_{Height}}{2} \tag{3.2}$$

This operation places the images in a more intuitive way. The $X0$ and $Y0$ represent half of the LCD's pixel resolution 120 and 160, respectively. Which translate the default top left corner LCD's referential to the centre of the display. X_{arg} and Y_{arg} variables represent the **x** and **y** arguments of the `RHeyes_ShowImg` and consequently display the image starting in the pixel with those coordinates from the image's top left corner to its bottom right. Because of that, half of the image's dimensions, $\frac{img\_Width}{2}$ and $\frac{img\_Height}{2}$, is subtracted to its respective coordinate, in order to centre the image in the pixel with $x = X_{arg}$ and $y = Y_{arg}$.

3.4 Mouth's LCD

For the mouth display it is used a ITDB02-1.8SP (see Fig. 3.9), which is a 1.8" TFT LCD module with 65K colour depth (16 bit) and 128 x 160 resolutions. The display uses the Serial Peripheral Interface (SPI) to receive data or commands and it has 4-line serial interface (SCL,SDA,CS,RS channels), plus a reset line (RST), to receive commands. The controller of this LCD module is ST7735 [15].

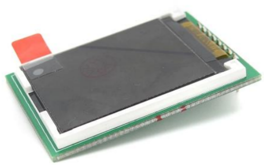


Figure 3.9 LCD used to display the RobotHead's mouth [15].

Each input pin has a specific function for the LCD to operate correctly. The CS,SCL,SDA are part of the SPI communication protocol from which the data is sent to the LCD. The CS pin is the chip select that when is low activates the peripheral, SCL is the protocol's clock and SDA the data channel. The manner in which the mouth's LCD works similarly to the eye module (illustrated in Fig. 3.10). It receives a set of commands and its arguments and acts accordingly.

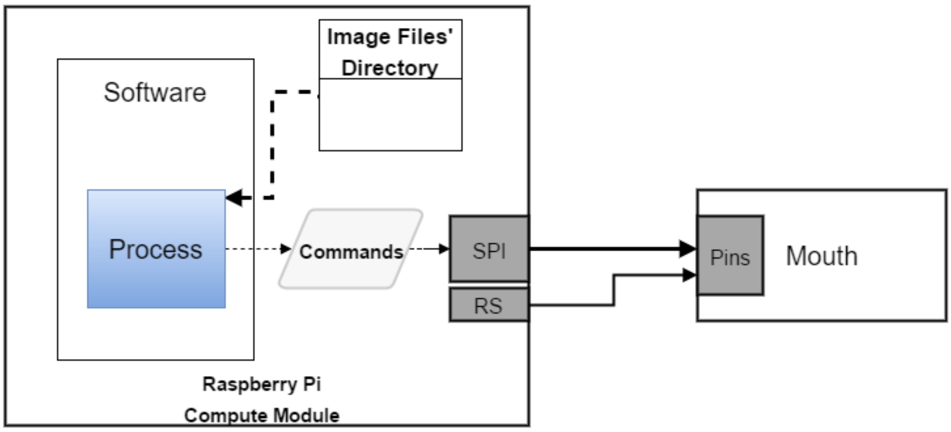


Figure 3.10 The image files a open by a process and the information is sent through the SPI peripheral.

However the mouth's module does not have the capability of responding to such commands neither has micro-SD memory card slot to store images of mouths, therefore the image's files must be stored in the RPiCM and be transmitted to the mouth's module, unlike the eyes' LCDs. Each image sent is stored in the module's Random Access Memory (RAM), where each pixel has a predefined colour format, in our case raw 16 bits, and subsequently displayed in an pre-configured order to the LCD screen.

3.4.1 Mouth LCD's Commands/Data input

In 4-line serial interface (SCL,SDA,CS,RS channels), the data packets just contains transmission byte and control bit Command/Data is transferred by the RS pin. If RS is "low", the transmission byte is interpreted as a command byte. If RS is "high", the transmission byte is stored in the display data RAM (memory write command), or command register as parameter. Any instruction can be sent in any order to the driver. The most significant byte (MSB) is transmitted first. The serial interface is initialized when CS is high. In this state, SCL clock pulse or SDA data have no effect. A falling edge on CS enables the serial interface and indicates the start of data transmission [27]. In figure 3.11 there is a representation of the mouth's module communication protocol.

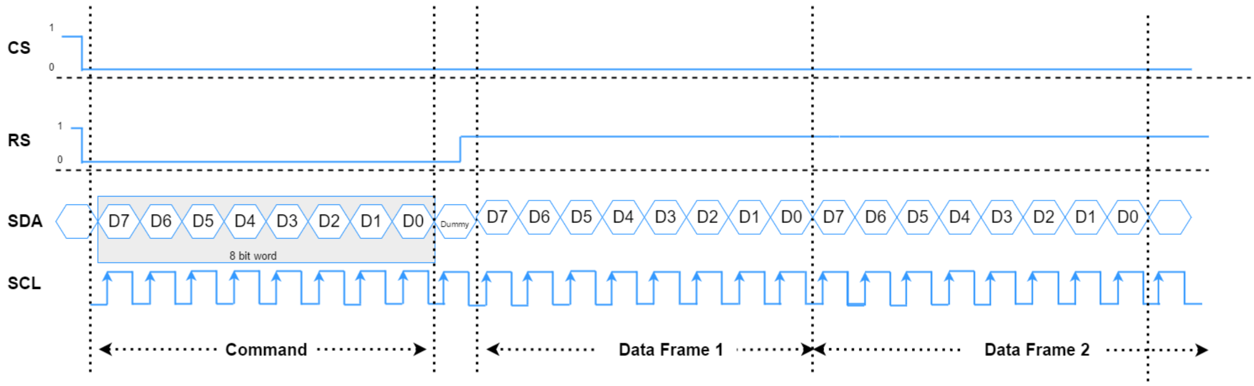


Figure 3.11 Mouth Module communication protocol

3.4.2 Mouth image's pre-processing

Although, there is already an understanding on how the mouth's module operates, there is still need to have the mouths' images in a format the is readable for the LCD. With this in mind the images for the mouth must have the same resolution has the mouth's LCD which is 160 x 128 pixels. There is also the matter of the image encoding, more precisely the image's colour format and the order from which the pixels' information is processed, because the image's pixel must be stored in the module's RAM with 16 bit raw format, where 5 bits are for the colour red, 6 for green and 5 for blue. This requires that each image be converted into a format that eases the process of writing the image's information to the mouth's LCD. For simplicity, a script in Matlab was made to search all the folders, in a target directory, which contain the frames that constitute one mouth's animation/expression. Then each image is converted from a 24 bit format to 16 bit. The resulting image's raw data is saved in a binary file to be displayed later in the ISR-RobotHead's mouth. Figure 3.12 represents the flowchart for the mouth's images conversion.

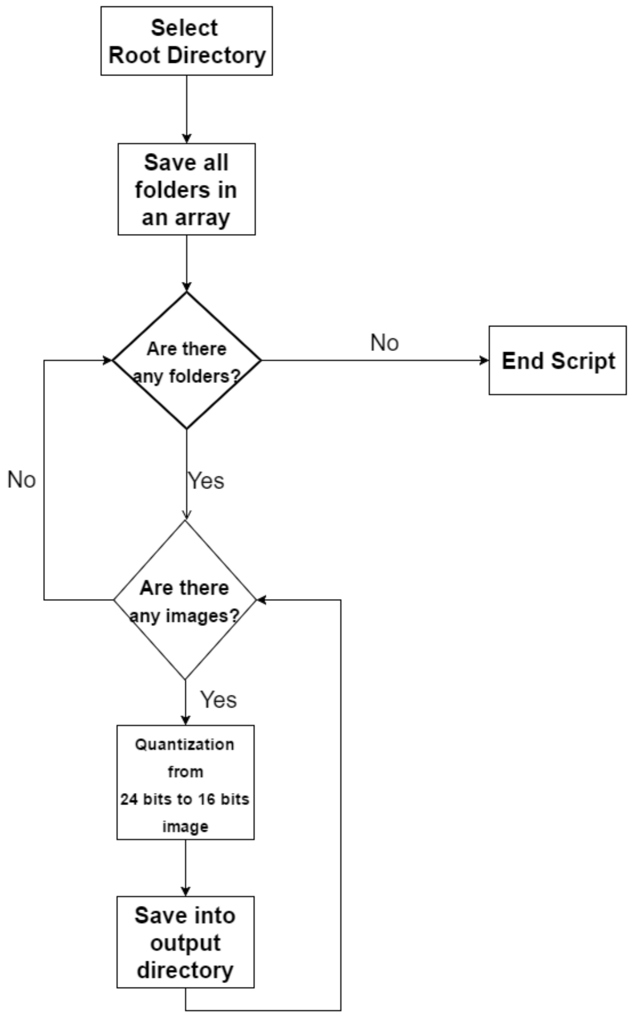


Figure 3.12 Mouth images encoding flowchart

In the *"Convert from 24 bit to 16 bit"* block the script runs through the images' pixels and reads its 24 bit colour value, 8 bit for red, green and blue components, which entails that each component has 256 colour levels. Each image is then quantized from a 24 bit format to 16 bit, which the colour encoding is configured as 5 bits for red, 6 for green and 5 for blue. This means that each component has 32, 64 and 32 colour levels, respectively. The quantization is a technique used in image processing that compresses a range of values to a single value (illustrated in Fig. 3.13). In this case, the quantization is made for each of the image's components individually, which entails that the colours' ranges in the 24 bit image that corresponds to a single value in 16 bit, for each component are 8, 4 and 8, for the colours red, green and blue, respectively.

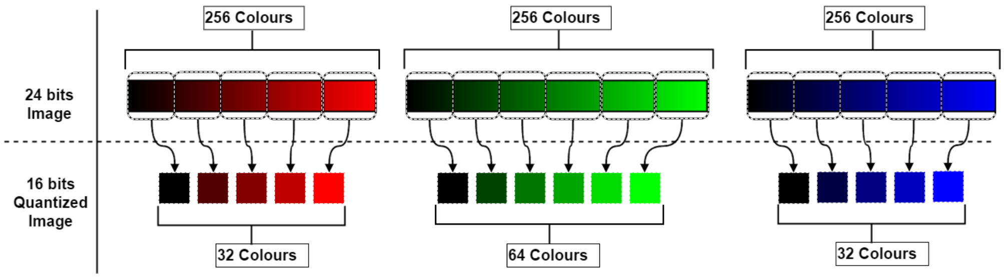


Figure 3.13 Illustration of the quantization of 24 bit image to 16 bit. Range of values of a 24 bit format correspond to a single 16 bit value

After the conversion, each pixel's value is written to a binary file in words composed of two bytes, this data is in a raw format, which means that the encoding of the pixel's information is written in two bytes with the associated colour values. The image's pixels are encoded in the order represented in Fig. 3.14, because this is the same order from which the mouth's LCD module displays the information from the RAM memory to the screen.

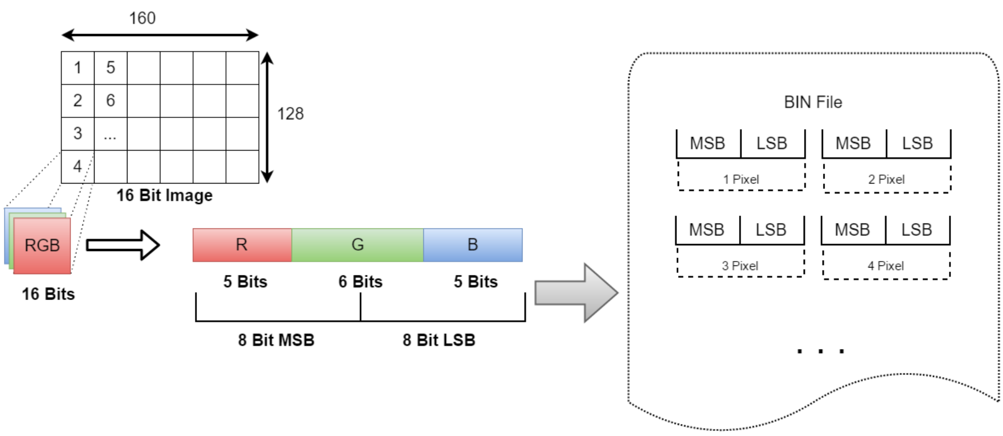


Figure 3.14 Image's pixel order conversion

With the images pre-processed in this manner, the animation of mouth expressions become quite linear (as it will be shown in the following section), since its encoding and order from which the pixel's are read, correspond to the processing of said images in the mouth's module. The Raspberry Pi only needs to read and send the information as it is structure, in the processed *bin* file, to the LCD.

3.4.3 Mouth's Library

In this section we are going to aboard the library `RHmouthLib.c` that was developed for the mouth module. Its main functions configure the SPI peripheral, load and send images to the "mouth's" LCD, in a similar methods as for the "eye's" LCD however with a different manner since the images are stored in the Raspberry Pi.

3.4.3.1 RHMouthLib Data Structure

The mouth's data structure (see Fig. 3.15) stores the image's file name char *img_name* and its file descriptor in order to search and send the image to the mouth's LCD.

```
struct RHmouth_imgs{
  int n_mouth;
  char img_name[40];
};
```

Figure 3.15 Mouth's data structure keep image information

The mouth's image name is saved in array **img_name** and the **n_mouth** is used with the purpose of saving the file's descriptor. Algorithm 3 illustrates the routine for the images' loading.

Algorithm 3 RHmouth_loadimages function. Load image information into data structure buffer.

```
function RHmouth_loadimages(char * dirname)
    Open File Directory
    N ← Count number of entries
    Allocate global structure by sizeof(struct RHmouth_imgs) ×N ← mouths_pointer
    for i ← 0 ; i to N do
        Open image file
        Save Image's Descriptor in RHmouth_imgs.n_mouth
        Save Image's Name in RHmouth_imgs.img_name
    end for
end function
```

The function works similarly to algorithm 1 for the eye modules. Although, the difference is that the images' files are opened in the Raspberry, and then their respective descriptors and names are saved in the global data structure.

3.4.3.2 RHMouthLib Display Mouth

The routine for the "mouth's" image to be displayed has the same code structure as the "eye's" LCDs 2. This is shown in algorithm 4.

Algorithm 4 RHmouth_display_image function. Send image file to the mouth's LCD routine.

```
function INT RHMOUTH_DISPLAY_IMAGE(int spi_fd, const char * name)
    for i ← 0 ; i to Number of images ; i++ do
        if global_pointer[i].img_name == name then
            Write image pixels through the SPI channel
            return global_pointer[i].n_img
        end if
    end for
    return −1
end function
```

However the main difference is how the image are transmitted to the display module. Where the "eye's" functions by sending commands that are processed by the LCDs and the image is displayed from its micro-SD card, the "mouth's" module require the full image information to be sent.

In figure 3.16, the flowchart represents the routine that transmits the image's pixels to the mouth's module.

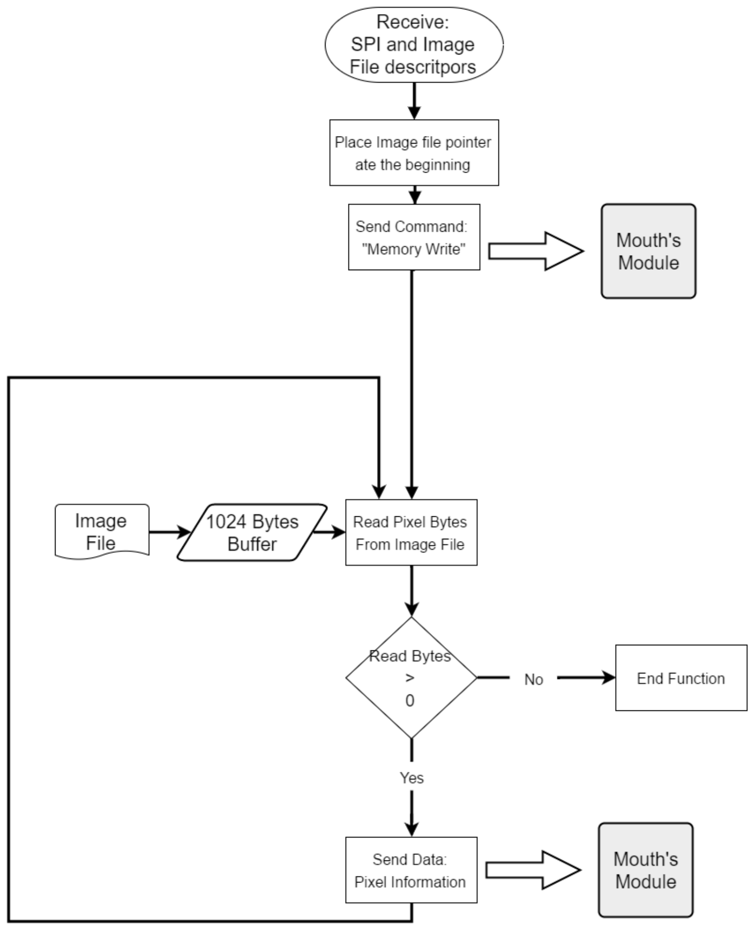


Figure 3.16 Image's pixel data being transmitted to the "mouth's" module.

- Receive the SPI and the image's file descriptors.
- Place the file pointer at the beginning.
- Send "Memory Write" command: puts the RS pin "high" and send through the SPI peripheral "Memory Write" command
- Read the image's file in 1024 Byte buffer, if the number of bytes read are more than 0 send the buffer to the SPI peripheral. This process is repeated until the image fully displayed in the LCD.

3.5 Pan and Tilt Mechanism and Actuators

3.5.1 Mechanism

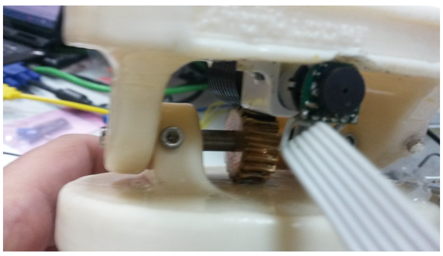
The Pan & Tilt mechanism [3] uses two 6 V brushed DC motor with a 297.92:1 metal gearbox are coupled to quadrature magnetic encoders to acquire the motor's shaft position, where each encoder adds a reduction ratio of 12:1 to the motor rotation.

The Pan and Tilt was created to change the RobotHead's orientation in the vertically and horizontally. The mechanical system is shown in the following figure:

Figure 3.17 Pan and Tilt mechanism

The Pan motor is coupled at the base of the mechanism and rotates the upper plate, where the Tilt mechanism is assembled. The Tilt mechanism has a very particular setup. It is composed by two different worm teeth wheels , however the only characteristic that is of interest for this project was the number of teeth, as it shall be explained in the following text.

(a) Tilt front view.



(b) Tilt motor.

Figure 3.18 Tilt's mechanism

In the front view image (Fig. 3.18), it can be seen both wheels that composed the Tilt system, one black other bronze. The black wheel is where the Tilt's motor is coupled and this wheel is also attached to the bronze wheel. This last is fixed to the platform, because the bronze wheel functions as a foothold to the wheel coupled to the motor. This setup makes that, when the motor rotates clockwise, the Tilt's platform moves forward, when the motor rotates counter-clockwise, the platform moves backward. The last aspect important to note, is that when the black wheel makes a full rotation, it shifts to other ledge of the bronze wheel, which means that in order to do a full rotation along the bronze wheel the motor's shaft must rotate 20 times, in other words, the bronze wheel adds a 20:1 reduction ratio.

3.5.2 Arduino and Motorshield

To control the motors coupled to the Pan & Tilt we use an *Arduino Duemilanove* micro controller with a Motor Shield that will drive the two motors (see Figure 3.19). The Motor Shield is a dual full-bridge driver designed to drive inductive loads such as relays, solenoids, DC and stepping motors. It drives two DC motors with *Arduino* board, controlling the speed and direction of each one independently. It can also measure the motor current absorption of each motor.

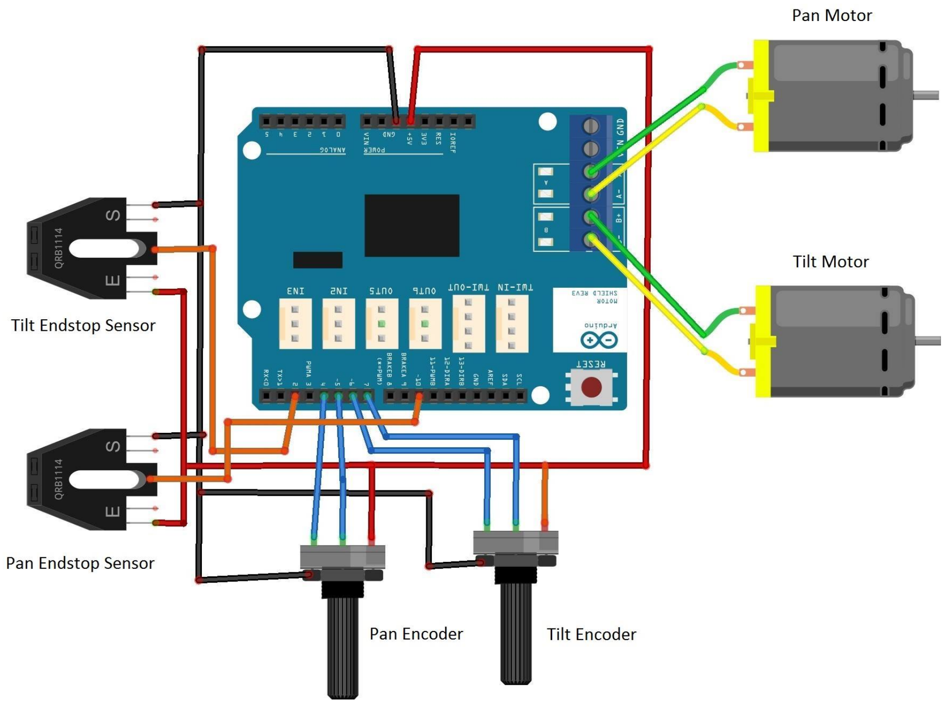


Figure 3.19 Arduino connections schematics.

In this section, it is presented the steps taken to implement the ISR-RobotHead's Pan and Tilt system controlled by an Arduino. First, it was necessary to estimate the transfer function for both DC motors coupled with Pan and Tilt system. Thereafter, it was projected and implemented an adaptive controller for each motor to control the Pan and Tilt orientations in velocity and position. To conclude, it was implemented the communication between the Raspberry and the Pan and Tilt to control the ISR-RoboHead's orientation. It is important to note that the head's structure was not coupled to the mechanism during the estimation and implementation of the Pan and Tilt controller.

3.5.3 System Estimation

Since the Pan and Tilt is composed by independent motors with their respective load, the same estimation process was used to estimate each orientation. Thus, the Matlab's *System Estimation Toolbox* [18] was used in order to model the Pan and Tilt motors' transfer functions. The angular velocity and input voltage for the motor estimation was acquired by using the *Arduino*. This values were sampled with a period of 0.02 s. In Fig. 3.20, it is presented the dataset used to estimate the pan system's transfer function.

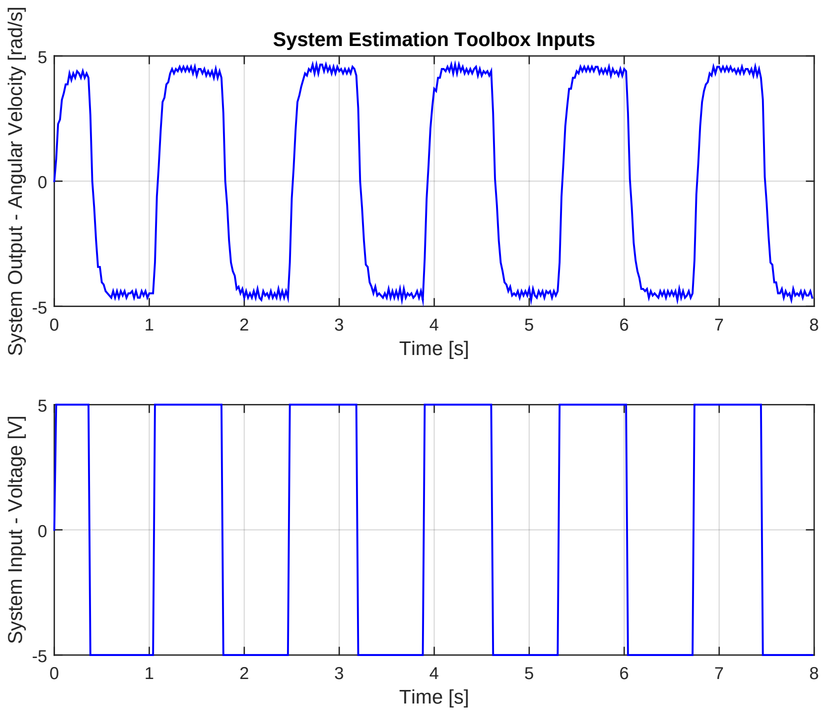


Figure 3.20 Data acquired from the Arduino.

The transfer function was computed with *System Estimation Toolbox* by are approximating the DC motors to first order systems. The resulting estimated system yielded a 91.21% fitting with real systems. Equation 3.3 is the estimated transfer function for the pan's motor.

$$G_{Pan}(s) = \frac{19.58}{s + 21.66} \tag{3.3}$$

The resulted fitting of this estimation was 91.21%.

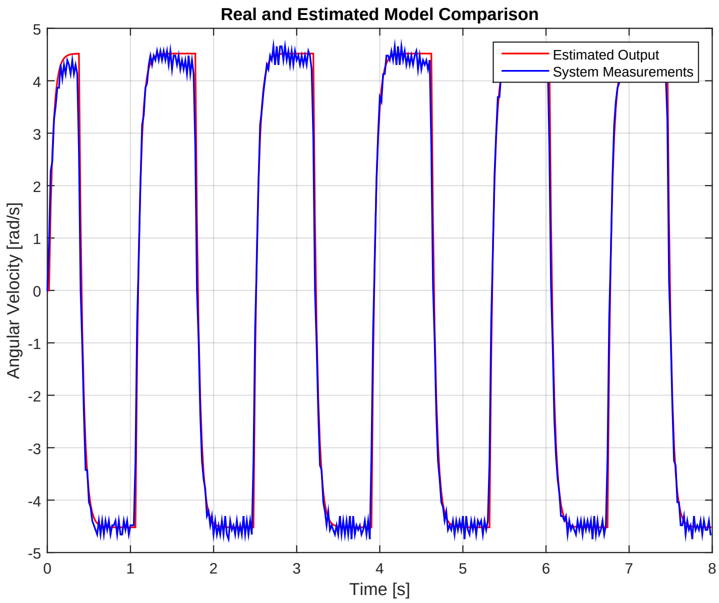


Figure 3.21 Estimation and real data comparison with a 91,21% fitting.

For the Tilt orientation, the same process was adopted. Which resulted in the Tilt system being estimated with the following transfer function.

$$G_{Tilt}(s) = \frac{1.407}{s + 21.11} \tag{3.4}$$

3.5.4 Control Model - Position and Velocity Controller

In order to change the ISR-RobotHead's orientation without compromising its internal components and make a smooth transitions between the motor's position, it was projected and implemented a controller to control the motor in velocity and position (illustrated in 3.22). The system has a sample period of *20 ms*, the same for modelling the system. The same controller was implemented for both pan and tilt's DC motors.

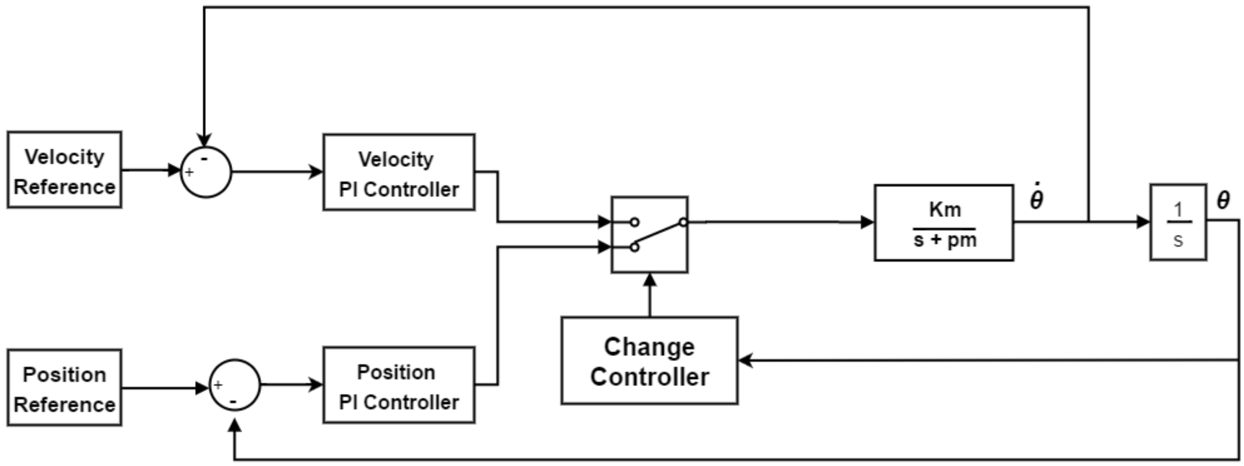


Figure 3.22 Control Model for one of the motors of the Pan and Tilt system

This model has a position and velocity controllers. The system start by controlling the motor's in velocity with a ramp input, then, when the motor is within a given position threshold, the controller switches to control the motor's position. This system was implemented with proportional integrative (PI) controllers for each motor to control velocity and position. This system was implemented in *Simulink* to validate its implementation in the *Arduino* micro-controller.

3.5.4.1 Position and Velocity - PI Tuning

With the estimation of both Pan and Tilt transfer functions, it was projected a PI velocity and position systems. The PI was used to control the DC motors since it minimizes the steady-state error which allows for the system to maintain a steady velocity, in case of velocity control, or place the motor DC at position given by the reference value. For the velocity and positions systems' angular frequency it was used 12 rad/s and 10 rad/s for both Pan and Tilt motors. All the systems were projected as critically damped. The obtained gains are presented in the following table:

Table 3.2 Gain values for the Velocity and Position controllers for each orientation

	PI Gains			
	Velocity Controller		Position Controller	
	Pan	Tilt	Pan	Tilt
Kp	2.054	0.3238	86.85	6.803
Ki	102.345	10.01	78.8913	8.478

The previous controller was implemented on the *Arduino*, with a sample rate of 2 ms. The Figs. 3.23 and 3.24 show the response of both simulations and real system. In the graphics are represented the systems' angular position, angular velocity and command signals: For the Pan's controller (Fig. 3.23) the reference position was set at *1 rad* and the threshold for the transition between controllers was set to *0.2 rad*, in other words, at *0.8 rad* the controller transits from velocity to position.

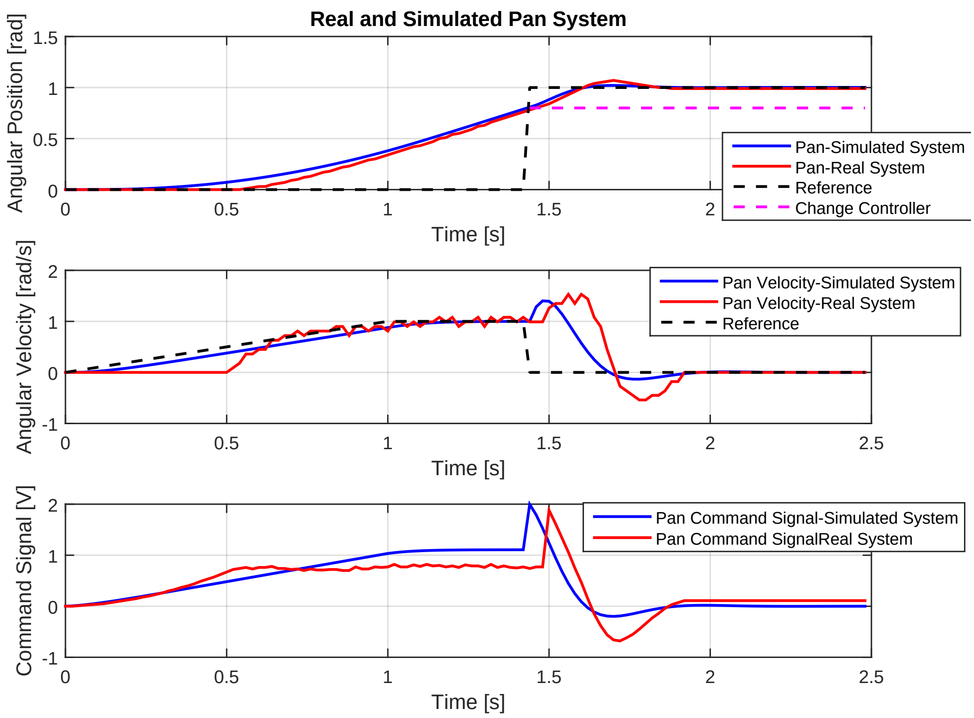


Figure 3.23 Response Graphics: Transition between velocity and position controllers

In the angular position figure, both simulation and real system present a similar curve during the transition from its origin to the reference as expected. Furthermore, the transition between the velocity and position controller can be seen on the velocity and command signals graphics, in which both exhibit a sudden variation when the motor is within the given threshold.

In the Fig. 3.24 it is represented the Tilt's controller response, its reference was set to *0.3 rad* and the threshold to *0.1 rad*.

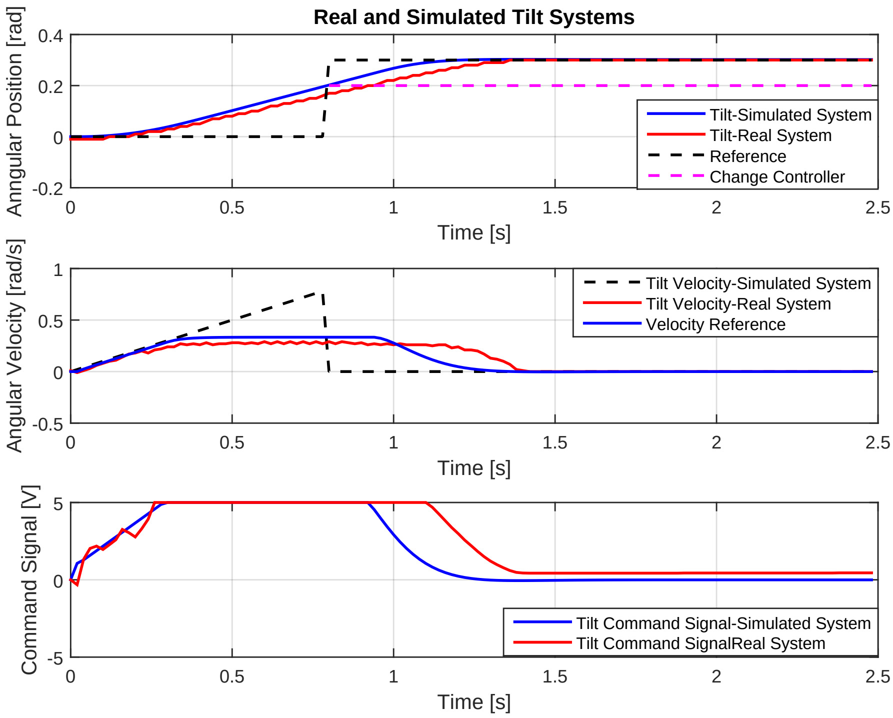


Figure 3.24 Response Graphics: Transition between velocity and position controllers

Similar to the Pan system, both simulation and real systems arrive at the given reference, however both saturate very soon since it is imposed by 20:1 reduction ratio which poses a difficulty when coupling the robotic head to the mechanism since the current system may require more electric power to move the head regarding the Tilt's orientation.

3.5.5 Raspberry Pi and Arduino Communication-I^{2}C

Having the controller implemented in *Arduino* ,it was still necessary to implement a method that allows change the pan and tilt motors' references. With this purpose, it was used the I^{2}C protocol to exchange information between the *Raspberry Pi* and *Arduino* board. The Raspberry functions as the master and the *Arduino* as slave. The *Raspberry* controls the I^{2}C, which means that besides the Pan and Tilt controller implemented it was necessary to write the interrupt routines for the I^{2}C on the *Arduino* to transfer data between each controllers. The data structure sent through the I^{2}C peripheral is as follows:

```
struct RHposition{
float Pan;    //Pan orientation in radians
float Tilt;   //Tilt orientation in radians
int arrive;   //flag for status verifications
};
```

Figure 3.25 Structure to keep image information

The variables, **Pan** and as the information of angle for the orientations Pan and Tilt, restively. The **arrive** variable functions as a status flag for the data transmitted in the I^{2}C channel. On the Arduino, the I^{2}C protocol as two associated interrupts for receiving other for sending information.

The flowcharts presented in Fig. 3.26 and 3.27 show the Arduino's interrupt routines for the receive and send interrupts, respectively.

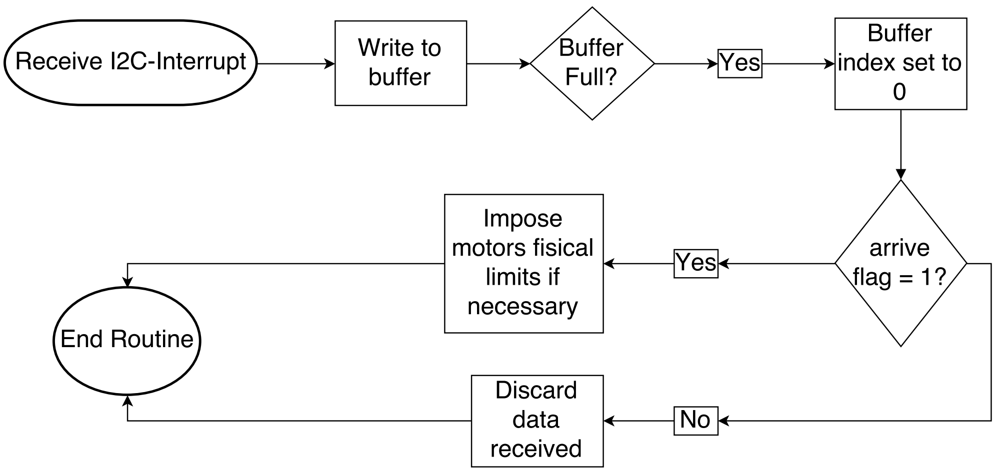


Figure 3.26 I^{2}C interrupts flow chart

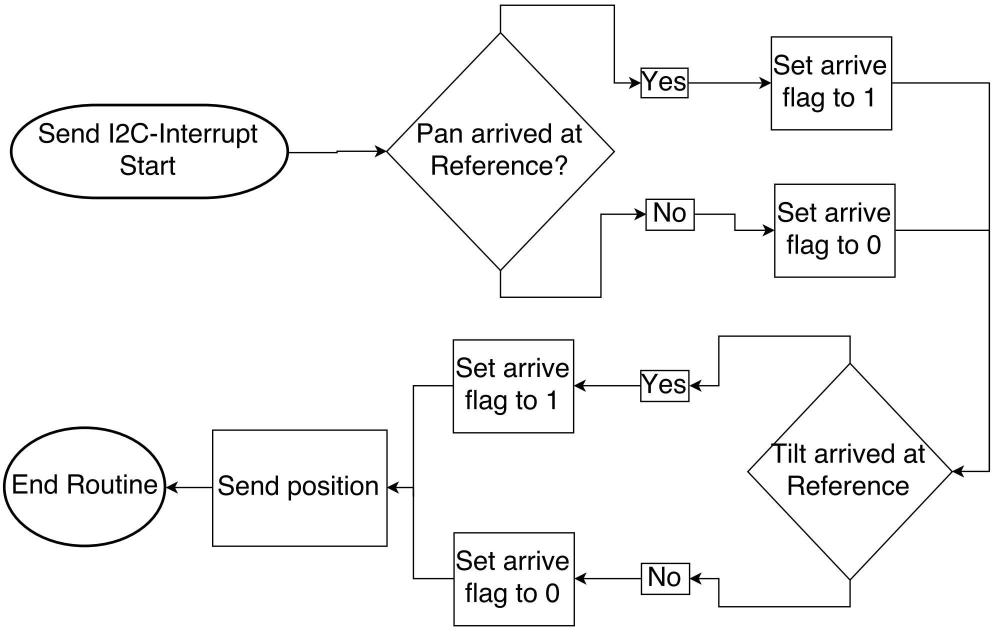
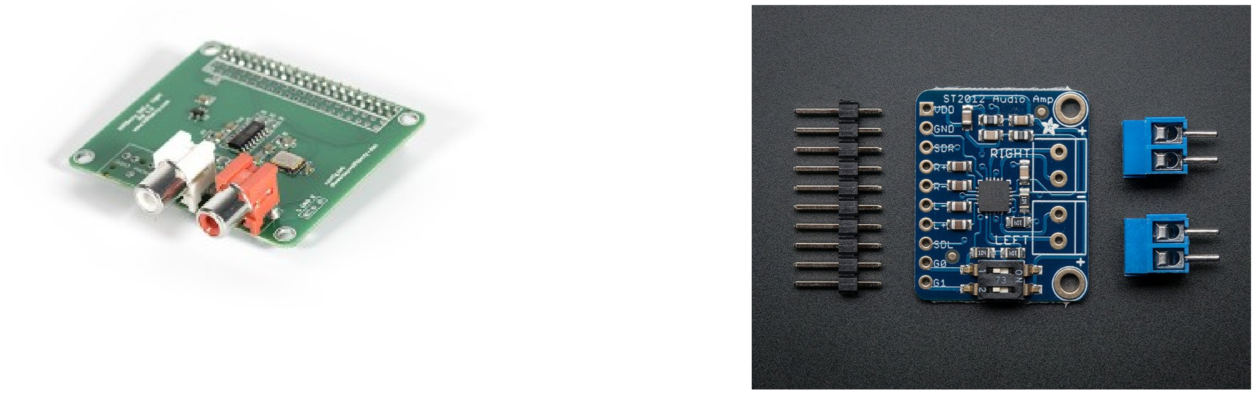


Figure 3.27 I^2C interrupts flow chart

- **Receive Information**
  - In the first callback it initializes an *int* variable that functions as the reception buffer's index
  - Reads a byte sent into the structure's buffer in each interrupt callback and increments the index for the following data to be received
  - When the buffer is full,verifies if the **arrive** flag is one, if not discards buffer
  - Then verifies if the motors' references received exceed the imposed limitation. If yes, limits the motor
- **Send Information**
  - Verifies if the motors are in the reference giving, then gives places the flag's bits at on or zero depending on the current status.
    * Pan in reference -> arrive=0001
    * Tilt in reference -> arrive=0010
    * Exceeded Pan's limitation in reference -> arrive=0100
    * Exceeded Tilt's limitation -> arrive=1000
  - After the **arrive** flag is set, send's buffer.

3.6 Sound Output

The sound output system is composed by two modules. An Hifiberry DAC+Lite which is a audio Digital-to-Analogue Converter (DAC) add-on board for Raspberry and a sound amplifier (seen in Fig. 3.28).



(a) Hifiberry Sound Card [14]. (b) Sound amplifier [4].

Figure 3.28 Sound output components.

The Hifiberry was developed to be easily plugged in a Raspberry Pi B+, because of that it was required to have the GPIO of the Compute Module Development Kit in the corresponding pins of the B+, although only the pulse code modulation (PCM) peripheral and the sources are needed for the DAC to be able to operate correctly.

The Hifiberry has three output pin outs, left and right sound channels and ground. This are wired to the audio amplifier to the corresponding inputs which then outputs to the left and right speakers (illustrated in Fig. 3.29).

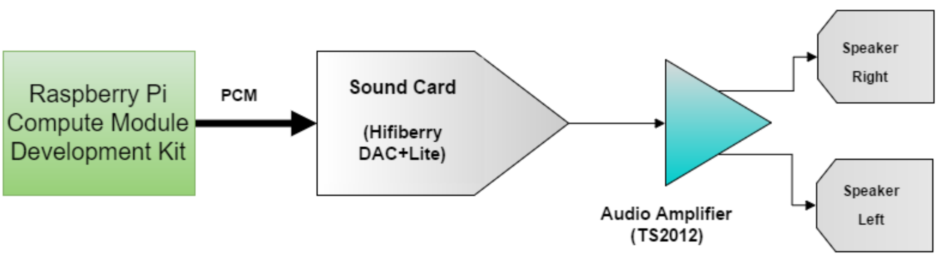


Figure 3.29 Sound system hardware schematic.

This module was tested by using *mplayer*, which is an open source software to play either audio or video. In this project we used this program to test the sound output module of the

ISR-RobotHead. This module is compatible with any software that employs the *Advanced Linux Sound Architecture* (ALSA) [1] drivers.

[1] http://www.alsa-project.org/main/index.php/Main_Page

Chapter 4

Results and Benchmark

In this chapter, the developed libraries for the robot's eyes and mouth are shown working together to achieve expressions' representations along with the results of an experiment to validate the solution.

On a more technical note, it is presented the time constraints to display images in the eye and mouth peripherals. In order to measure the time for each peripheral, the POSIX *time* [20] library was used.

Thereafter, it is presented each modules' computation and memory occupancy to better understand the strain in the Raspberry Pi controller to operate the peripherals.

4.1 ISR-RobotHead's Expression Representation

With the developed libraries described in 3.3.4 and 3.4.3, it is now possible, to employ routines for printing images to the LCD's mouth and eyes to create expressions in the ISR-RobotHead.

This setup, gives the possibility to create animations for each expression, instead of having an abrupt change between eyes or mouth images, each frame can be printed in a logic sequence, with a time delay between two images, to create a more fluid transition or emulate movement in the robot's eye or mouth. A result of an expression can be observed in the Fig. 4.1 along with the image files to create the facial expression.

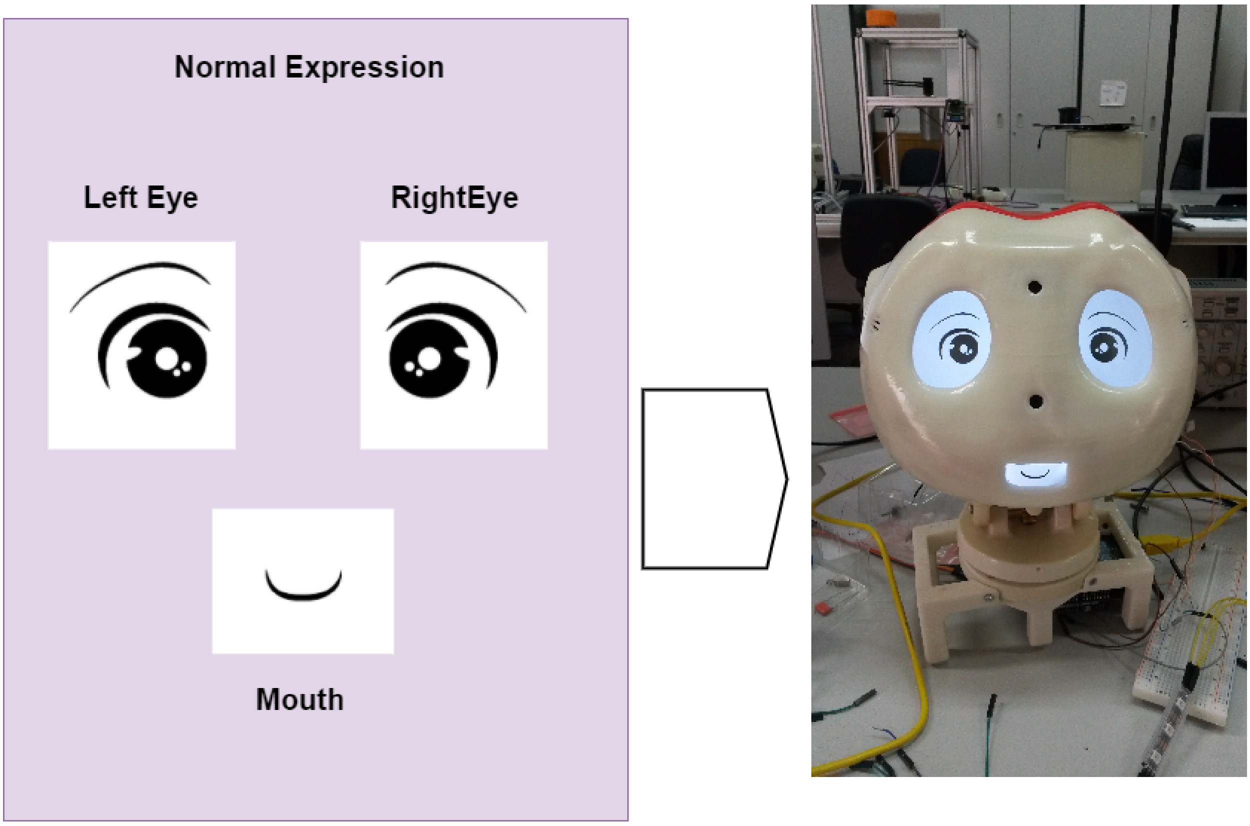


Figure 4.1 ISR-RobotHead Normal expression along with the image files for the respective expression.

The previous expressions was considered as the "**Normal**" or default expression for the eye's and mouth, since all images designed to create animations start from this set of frames for the respective peripherals. The eyes and mouth's images presented in the left frame, were upload to the respective device through the processes described in sections 3.3.3 and 3.4.2.

For illustration purposes, in Fig. 4.2 is the "**Blink Eye**" animation. Wherein each pair of "eye'" frames are displayed by employing the developed functions with a time delay between two commands. However, this time delay is set higher than 70 milliseconds, since by measuring the time between the call `Display Eye` function and receiving an acknowledge from one LCD the result was 70 milliseconds.

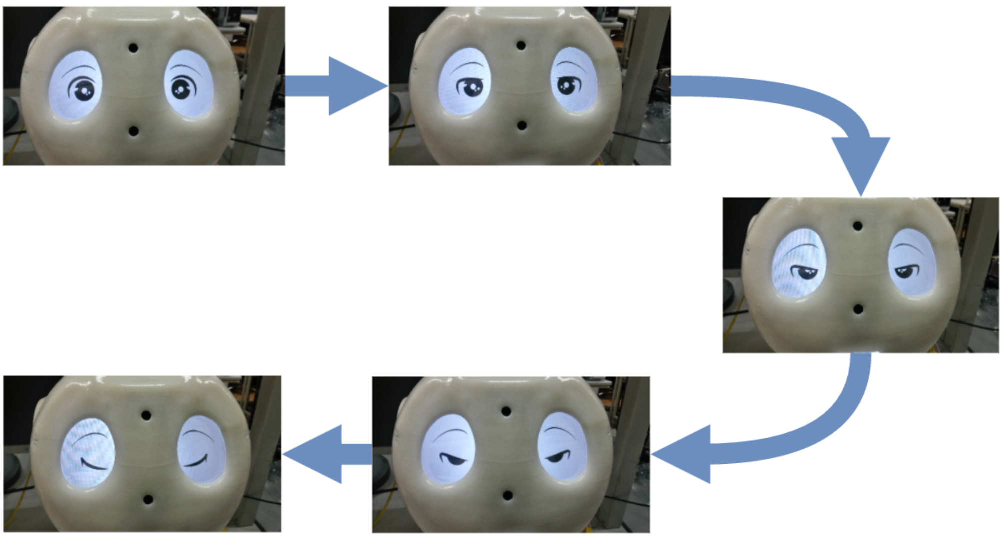


Figure 4.2 Normal Expression with the corresponding images

The time delay between two expressions, can vary depending on the animation. However, by measuring the time that takes for an issued command to respond, the time delay should not be inferior to 70 milliseconds, since the data sent can be misinterpreted by the eye's LCD. This time restriction, however, does not seem to pose a visual constraint since the change in image being displayed is not apparent to the viewer.

In a likely manner, the mouth's module takes 30 milliseconds to display an image. However, for this peripheral, the frames are loaded in the Raspberry Pi and transmitted to the LCD. After the data is sent, the peripheral does not require to set a delay to be operable again, unlike the eyes' modules. In the Fig. 4.3 it is presented the **"Sad"** mouth animation, which follows the same principle of the "eye" module animations, but with different data transmission method.

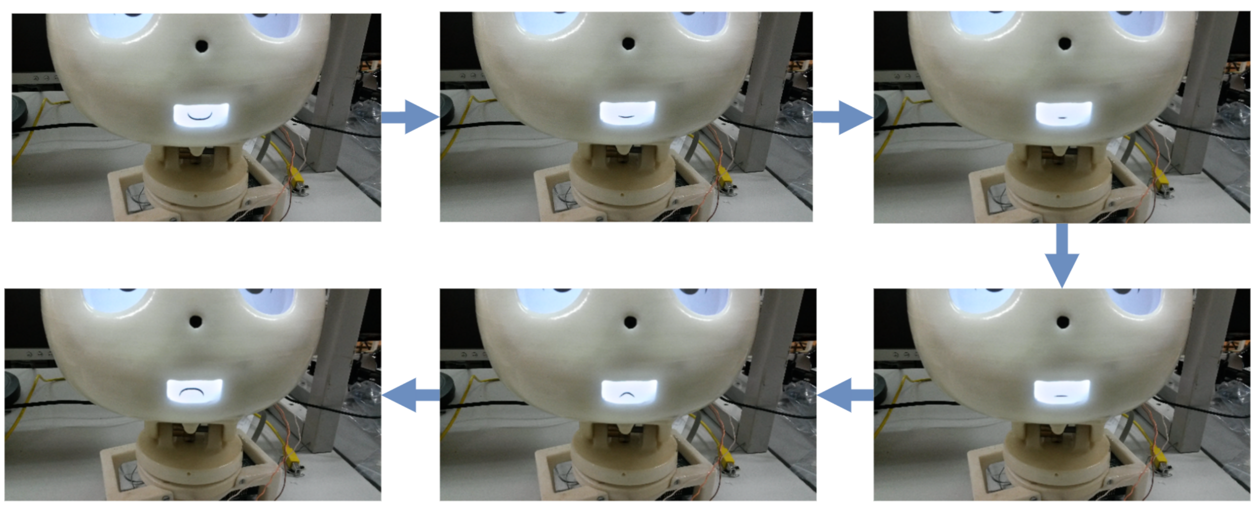


Figure 4.3 Sad mouth animation

In order to demonstrate the expressions, it was created a program in *C* with a set of animations that represent an emotional expression (see Fig. 4.4). This is achieved by alternating between images previously loaded into the RobotHead's controller employing the functions implemented in subsections 3.3.4.2 and 3.4.3.2.

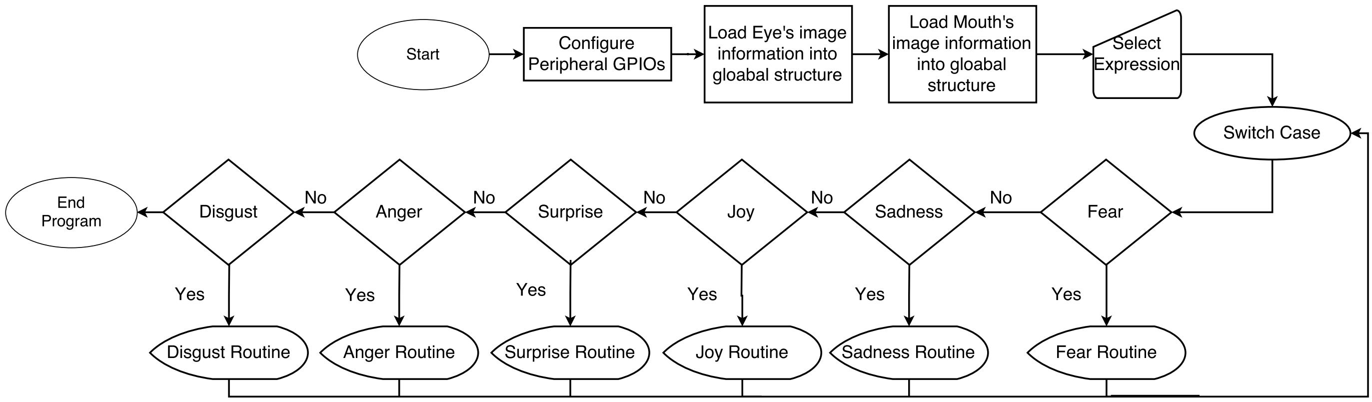


Figure 4.4 Flowchart of the robot head program to display expressions

The necessary information required to display any given image is loaded into the "eye" and "mouth" arrays, explained in sections 3.3.4.1 and 3.4.3.1. After this configurations, the expressions are implemented into routines by calling any given sequence of images that compose one single expression, where each has time delays between the called images to create a fluid animation.

The expressions created for demonstration purposes were: **Fear**, **Sadness**, **Joy**, **Sad**, **Surprise**, **Anger** and **Disgust** (illustrated in Fig. 4.5).

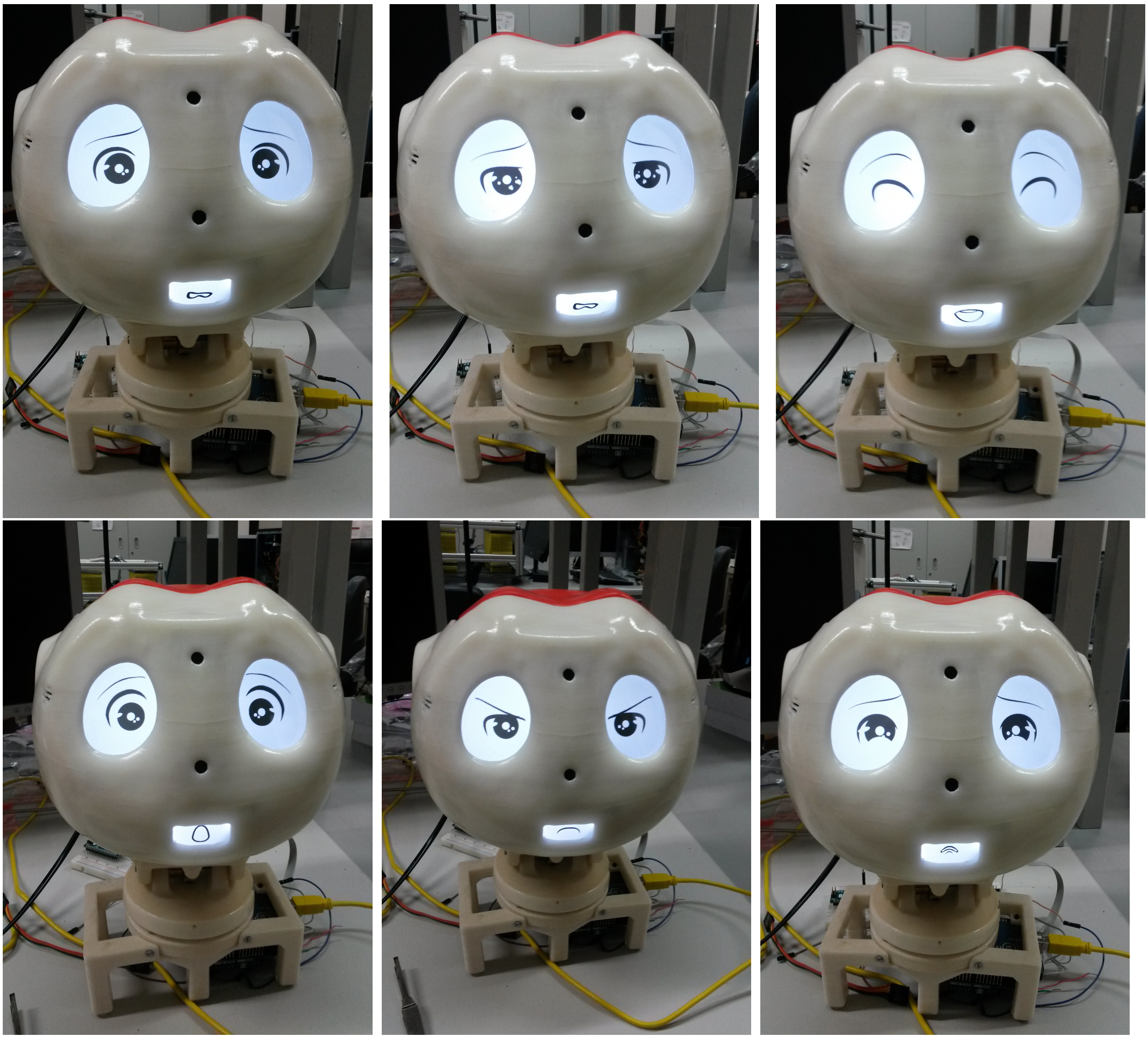


Figure 4.5 Pictures of the ISR-RobotHead representing a set of encoded emotional expressions. On top from left to right fear, sadness, joy. On bottom from left to right surprise, anger and disgust.

This specific expressions were chosen for the experiment since other works have used them for the same analysis [17], [16]. There are a possible set of emotions that the robot can express, since this solution allows to include other and more elaborate images to represent emotions in order to engage in human robot interaction.

In order to validate this expressions, a experiment was made where a number of children one-by-one, characterized the facial expressions as they saw, fit.

4.1.1 Experimental Validation

For validation, a survey was performed at the Cerebral Palsy Association of Coimbra in Portugal and in Psikontacto, Lda with a sample of 9 children, aged between 6 and 10 years old (experimental setup presented in Fig. 4.6). Most of the children were referred to the services because of some form of psychological disorder ($n = 7$); 3 of these children had some degree of intellectual disability; and 2 of them had a neurodevelopmental condition. Children were presented the robotic head and asked to identify six emotional states under examination 4.5 (fear, sadness, joy, surprise, anger, and disgust).

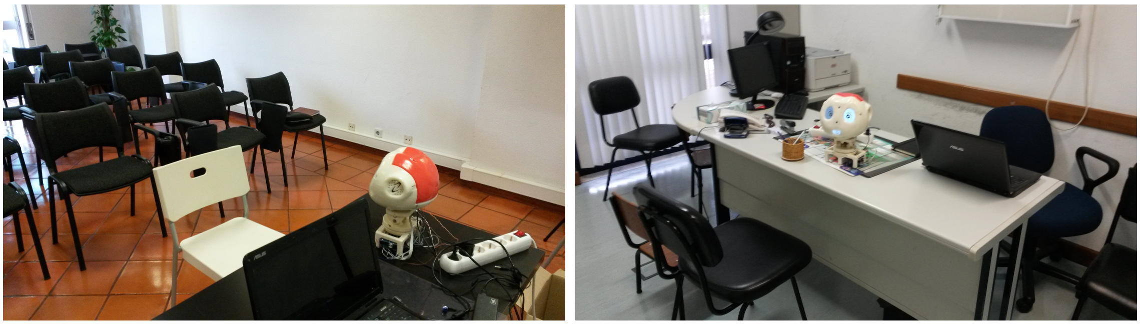


Figure 4.6 Experimental set up in the psychology clinic Psikontacto (left) and Cerebral Palsy Association of Coimbra (right)

4.1.2 Results

As depicted in Fig. 4.7, all participants correctly identified the emotions of Sadness, Joy, and Anger, thus attesting the ability of this LCD-based robotic head for non-verbal communication of these emotions. Likewise, most participants ($n = 6$) correctly detected the emotions of Fear and Surprise. However, only one of the participants was able to identify the emotion of Disgust, which indicates a clear need of improving the LCD animations created for this particular emotion. Despite the fact that younger children lack the verbal ability to accurately label more complex emotions, we believe that the LCD eyes-eyebrows and mouth animations can be recreated to properly convey this particular emotion of disgust, which is triggered in humans by the threat-detection affective regulation system and is often accompanied by vocalizations in its expression. The small sample size precluded the conduction of comparative analyses between age-groups or different levels of cognitive ability, but nonetheless, these are aspects to be fully taken into account in similar studies to be conducted in the future.

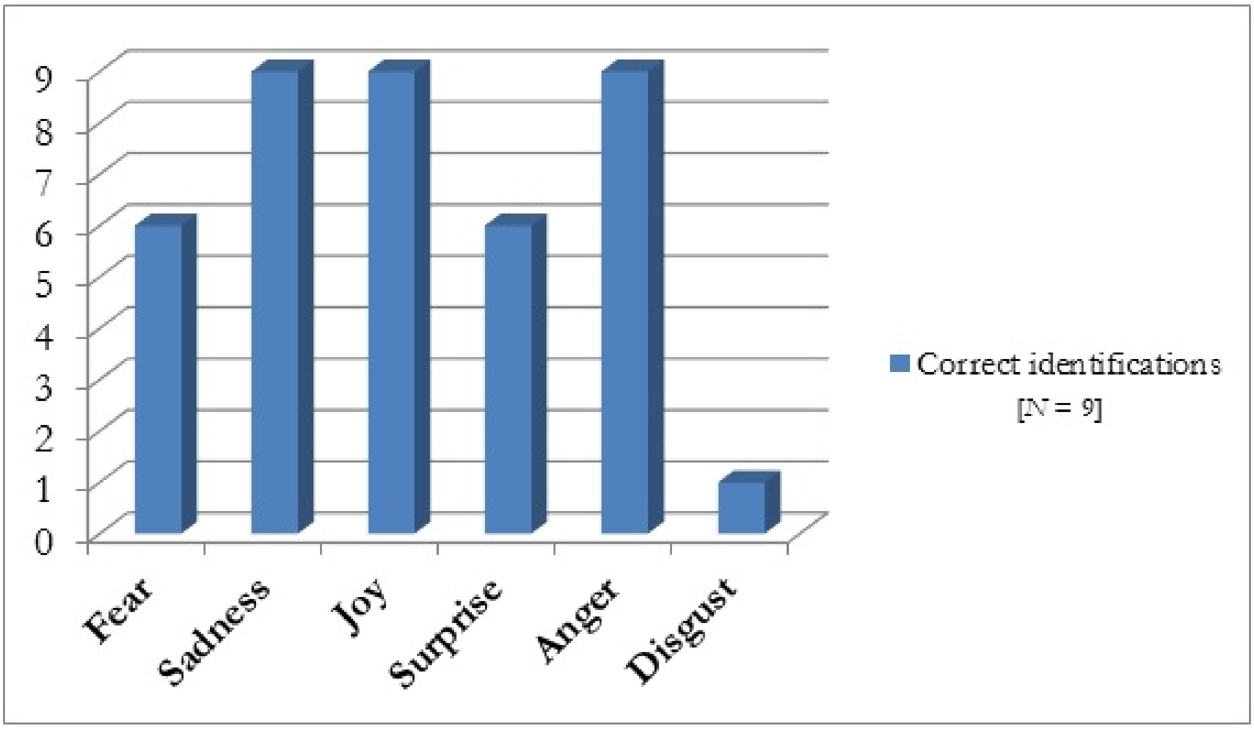


Figure 4.7 Frequencies of correct detection of robotic emotional expression by children.

4.2 CPU Occupancy and Memory

Besides the expression representation, the ISR-RobotHead has other modules that would be used in future iterations of this project. In this section, it is presented the technical benchmarks of each module controlled by the RPiCM, in order to access the computational (CPU) and memory (RAM) occupancy necessary for the micro computer to operate each module.

To measure the CPU and RAM usage using *htop* [1] while the target modules were executed in background. The RPiCM's RAM is 256 MBytes and the CPU has a speed of 700 MHz.

Recalling the modules that are part of the robotic head, the overall system are: dual camera system, sound output, eyes and mouth expression and Pan and Tilt system. For the dual camera system, two programs were used, one for video capture and other to stream (*nc*[2]) the captured frames to an external computer.

The following table 4.1 has the benchmark results for sound output, video capture, video stream, expression modules and Pan and Tilt system.

[1]**htop** is an interactive process viewer for Unix systems. It is a text-mode application (for console or X terminals).

[2]**nc** is a utility program to create arbitrary TCP and UDP connections and listens

Table 4.1 Benchmark results. Sound output, Video Capture, Video Stream, Expressions Routines CPU and Memory Occupancy

Peripheral	Software	CPU Occupancy %	Memory Used %
Sound Output	*mplayer*	15	1.3
Video Capture	*raspivid*	3	0.9
Video Stream	*nc*	4	0.7
Expression Routine	*RH_demo*	1	1.7

The results of the benchmark measurements show that the Raspberry Pi can be used to control the expression's peripherals. Since the developed software in this dissertation shifts most of the processing requirement to its respective module, the RPiCM only requires to send data/commands to interact with each actuator.

The video capture and stream also use a low percentage of the CPU and memory. This is interesting to note because of the possibility of the Raspberry Pi not being able to handle the processing of computer vision software, however this can be done on one external computer and the Raspberry can handle the interaction with the environment.

The sound output module, however is the one that requires most of the processing power. This was tested by employing *mplayer* and *aplay*, which yielded the same results.

Chapter 5

Conclusion and Future Work

In this work, it was developed a new solution for a robotic head with the ability to express emotions for CRI. The ISR-RobotHead is equipped with actuators to imitate the eyes and the mouths of humans. The results from the experiment with children served to validate the solution, since the surveyed subjects were able to identify most of the emotions that were intended to be reproduce in the robot. However, results also yield that there is still room for improvement in design of some of the emotions present in the robotic head. In addition, more expressions can be added into the ISR-RobotHead since one of the worries throughout the project was to allow a fairly simple technical solutions to be updated and upgraded in future versions of this work.

At the moment, the ISR-RobotHead has expressive capabilities, however, it still lacks the ability to speak which would make the solution more interesting since it gives the robot head the additional human modalities for communication (verbal and facial expression). In addition, most of the remaining peripherals work and are operational, although still lack some sort of function or still need some improvements, these are: the stereo camera system, the pan and tilt controller and the sound output.

Furthermore, there are still improvements to made tp the current ISR-RobotHead solution. The pan and tilt system needs to be improved since the tilt motor does not have enough torque to move the robot and the support structure has to be redesigned since it has structural flaws and does not allows for the head to be safely supported when controlling the tilt's motor to a new position. For the eye LCDs, it would be useful to have both LCDs with a dedicated channel in order to easily update the eye database in each LCD independently and also to use touch-screen features of these peripherals.

Regarding the sensor modules, the ISR-RobotHead can not react to any kind of action of effect since the sensors included in the project (cameras, microphones, touch-screens LCDs) don't send any type of feedback for the system to react. Which gives a great array of possibilities for this project on a multidisciplinary level. It could be interesting to implement artificial intelligence (AI), for example: identify a person or child emotional expression through the use of machine learning and react accordingly; face tracking by using the cameras and following with the pan and tilt system. To add further, it would be important to do more experiments with children and do studies in the field of psychology to see how the robotic head can be used for children with some mental or social disabilities.

Bibliography

[1] Humanoid robots. https://agrosy.informatik.uni-kl.de/en/research/humanoid-robots/ [Online; accessed October 27, 2016].

[2] 4DSystems. http://www.4dsystems.com.au/product/ulcd_32ptu/. [Online; accessed July 18, 2017].

[3] André Lopes and Daniel Almeida. ISR Robot Head: Mechanical, electronics and design. Technical report, Instituto de Sistemas e Robótica-Universidade de Coimbra, 2015.

[4] Arduino. https://www.arduino.cc/en/main/arduinomotorshieldr3. [Online; accessed July 18, 2016].

[5] Allison Bruce, Illah Nourbakhsh, and Reid Simmons. The role of expressiveness and attention in human-robot interaction. In *Robotics and Automation, 2002. Proceedings. ICRA'02. IEEE International Conference on*, volume 4, pages 4138–4142. IEEE, 2002.

[6] Albert De Beir, Hoang-Long Cao, Pablo Gómez Esteban, Greet Van de Perre, Dirk Lefeber, and Bram Vanderborght. Enhancing emotional facial expressiveness on NAO. *International Journal of Social Robotics*, 8(4):513–521, 2016.

[7] Paul Ekman. *Facial expression.* Nonverbal behavior and communication, 1978.

[8] Fernanda Faria. A study on robotic heads design within the scope of social and emotional expressiveness for intuitive human-robot interaction. Technical report, Instituto de Sistemas e Robótica-Universidade de Coimbra, 2015.

[9] T Fong, I Nourbakhsh, and K Dautenhahn. A survey of socially interactive robots: Concepts. *Design, and Applications*, 10:16–17, 2002.

[10] Raspberry Pi Foundation. https://www.raspberrypi.org/blog/raspberry-pi-compute-module-new-product/. [Online; accessed July 18, 2016].

[11] Raspberry Pi Foundation. https://www.raspberrypi.org/wp-content/uploads/2014/04/cm+io-small.jpeg. [Online; accessed August 18, 2016].

[12] Takuya Hashimoto, Hiroshi Kobayashi, Alex Polishuk, and Igor Verner. Elementary science lesson delivered by robot. In *Proceedings of the 8th ACM/IEEE international conference on Human-robot interaction*, pages 133–134. IEEE Press, 2013.

[13] Gordon Henderson. Gpio interface library for the raspberry pi. http://wiringpi.com/ [Online; accessed Aug 29, 2016].

[14] Hifiberry. https://www.hifiberry.com/wp-content/uploads/2015/09/dacplus-light-300x300.jpg. [Online; accessed July 18, 2016].

[15] ITEADStudio. https://www.itead.cc/display/tft-lcm/itdb02-1-8sp.html. [Online; accessed July 18, 2016].

[16] Kazuko Itoh, Hiroyasu Miwa, Munemichi Matsumoto, Massimiliano Zecca, Hideaki Takanobu, Stefano Roccella, Maria Chiara Carrozza, Paolo Dario, and Atsuo Takanishi. Various emotional expressions with emotion expression humanoid robot WE-4RII. In *Robotics and Automation, 2004. TExCRA'04. First IEEE Technical Exhibition Based Conference on*, pages 35–36. IEEE, 2004.

[17] Jochen Hirth Karsten Berns. Control of facial expressions of the humanoid robot head ROMAN. Technical report, Robotics Research Lab Department of Computer Science University of Kaiserslautern Germany, 2006.

[18] Lennart Ljung. *System identification toolbox: user's guide*. Mathworks Inc., 1995.

[19] Masahiro Mori, Karl F MacDorman, and Norri Kageki. The uncanny valley [from the field]. *IEEE Robotics & Automation Magazine*, 19(2):98–100, 2012.

[20] time.h library POSIX. http://pubs.opengroup.org/onlinepubs/7908799/xsh/time.h.html. [Online; accessed November 5, 2016].

[21] Raspivid. www.raspberrypi.org/documentation/raspbian/applications/camera.md. [Online; accessed October 18, 2016].

[22] Ben Robins, Kerstin Dautenhahn, R Te Boekhorst, and Aude Billard. Robotic assistants in therapy and education of children with autism: can a small humanoid robot help encourage social interaction skills? *Universal Access in the Information Society*, 4(2):105–120, 2005.

[23] Baxter robot webpage. http://www.rethinkrobotics.com/. [Online; accessed February 12, 2017].

[24] Buddy robot webpage. http://www.bluefrogrobotics.com/. [Online; accessed February 12, 2016].

[25] NAO robot webpage. https://www.aldebaran.com/en/search/node/NAO. [Online; accessed February 12, 2017].

[26] Pepper robot webpage. https://www.aldebaran.com/en/a-robots/who-is-pepper. [Online; accessed February 12, 2017].

[27] Sitronix. St7735 -262k color single-chip tft controller/driver.

[28] Sarah Woods, Kerstin Dautenhahn, and Joerg Schulz. The design space of robots: Investigating children's views. In *Robot and Human Interactive Communication, 2004. ROMAN. 13th IEEE International Workshop*.

Robotic Head with Emotional Expressions for Human-Robot Interaction

Ricardo Loureiro[1], André Lopes[1], Carlos Carona[2], Daniel Almeida[1], Fernanda Faria[1], Luís Garrote[1], Cristiano Premebida[1] and Urbano J. Nunes[1]

***Abstract*—Worldwide, several research works have been devoted to the development of humanoid robots for human-robot interaction (HRI). Particular attention has been paid to the concept, design and applications of robotic heads and facial-expressions. Together, head and its facial expressions, they compose a fundamental, and "familiar", interface to end-users; which is particularly true in child-robot interaction (CRI). This paper contributes with a functional prototype of a robotic head using LCDs for eyes-eyebrows and mouth animations. The prototype also incorporates stereo cameras, microphones, speakers and pan and tilt control on the vertical and horizontal axis. Experiments and validation of the robotic head were carried out at the Cerebral Palsy Association of Coimbra with children (6 – 10 years old). Preliminary observations show that the proposed robotic head, motivated mainly as an interface for children, can accurately convey some of the most basic emotions, whose were recognised by children. The study carried out helped to identify issues for future developments, such as specific or complex emotional states that need to be refined.**

I. Introduction

HRI has grown at a very high rate, with applications in industrial and social environments across the globe. On a social level, the robots are built on an appealing way, being it visually uncommon but capable of capturing a person's interest [1].

Most of research on HRI is devoted in designing robots as assistants, companions, or pets in addition to the more traditional role of servants, with interactive features to augment its functionality and interest in the end-user's perspective. Accordingly, in order for a person (adult, elder, child) to interact with a robot in a believable manner, the robots face functions as a "familiar" interface by employing human-like social cues and communication modalities, by means of facial expressions, sounds, movements and artificial intelligence [2].

In this paper, a new solution for a robot head [3] is introduced (Fig. 1) towards research on child-robot interaction.

Given the fact the effectiveness of multimodal CRI significantly depends on the robots ability to perform meaningful social interactions (which in turn requires accurate emotional expression and detection) [4], this exploratory study was aimed at validating the ability of a LCD-based robotic head to convey six basic emotional states to children with special healthcare needs.

[1] Authors are with Institute of Systems and Robotics, University of Coimbra, DEEC-Polo II, Coimbra, Portugal.

[2] Author is with Cerebral Palsy Association of Coimbra; Cognitive and Behavioral Center for Research and Intervention-University of Coimbra, Coimbra, Portugal.

This work has been supported by FCT under projects AMS-HMI2012-RECI/EEIAUT/0181/2012 and UID/EEA/00048/2013, with FEDER funding, programs QREN and COMPETE. Luís Garrote is also supported by FCT Grant SFRH/BD/88459/2012.

Email: rgloureiro1121@gmail.com, ccarona@fpce.uc.pt, {cpremebida, garrote,fernanda, urbano}@isr.uc.pt

A. Related work

This work falls within the field of social robotics, or social-interactive robots, a multidisciplinary research area involving sensory perception, psychology, bio-inspired systems, bio-engineering, mechatronics, artificial intelligent, and so on. In terms of applications, we can highlight the use of social-robots as guiders, teachers or therapeutic facilitators. In such cases, the robot receives inputs (such as gesture and speech recognition) and present feedback in a transparent interface that regular people can interpret and capture the attention from the observers: this is possible by employing emotional/facial expressions that a human can easily interpret and then a clear communication and interaction between the human and the robot can be established. Regarding the potential solutions to be used in a robotic head to transmit an emotional feedback to the end-user, the most commons are: LED interactive, kinematic, animatronic and LCD-based robotic head. LCD interactive robotic heads, like the one presented in this paper, represent faces mostly in tablet like screens and can achieve a big array of human expressions [1].

1) Robotic Heads: All the robotic head solutions have the common objective of capturing a person's attention by giving the machine the social complexities required that are characteristic of human social behaviour. This aspects serve to enhance human interaction with the robot, when the last employs tasks such as a house servant or a healthcare giver [2].

Despite its simplicity, the LED-based robotic heads are capable of displaying a wide range of colours and are very cost-effective. In addition, the robot heads have embedded cameras, microphones and speakers to read environmental information and to transmit sounds. This category has advantages and disadvantages regarding expressiveness [5], but the provided features are enough to keep the human attention.

Kinematic heads are animatronic heads developed to provide a communication more appealing to humans. These solutions are equipped with mechanical actuators to create eye and mouth movements in a more natural way than LEDs, but, conversely, they have more technical complexity. The robot heads with the most humane characteristic are animatronic robot heads with artificial and flexible skin. Similarly to the kinematic robot heads, these faces recreate human expressions through the use of mechanical actuators that recreate muscles and tendons which give them a close resemble to humans. However, these solutions may be uncomfortable to interact with because the attempt to look too human for an artificial construct [6].

2) Applications: Most research involving human-robot interaction have robots that focus on assistants, companions or pets to interact with a target group. For example, children with autism tend to exhibit difficulty in interacting with other humans. With this in mind, researchers have been exposing children, with this impediment, to interactive humanoid robots and results suggest that these children develop some social skills with the increase exposure to such robots [7].

Another application worthy of mention was an experiment where the researchers have used a robot to give a science class to children [8]. The research suggested that the use of robots can engage students in active learning, keep attention to subject and attract interest in the lesson.

Overall, these research works in child-robot interaction (CRI) still require more further development and experimentation to validate robots effectiveness in children development.

B. Emotion Expressions

For a robotic head, it is not enough to just create emotional expressions like a human being, but also to provide the interaction skills with the observer in an appealing way, so people can feel empathy with the robotic head. In an experiment performed by Bruce *et al.* [9], a robot tries to engage people to answer a set of questions by tracking them and then responding with emotional expressions regarding the persons willingness or unwillingness during the interaction with the robot. The results suggest that, between the facial expressions and the movement tracking, persons' interest towards the robot is the same. However, by combining both features the persons were more willing to stop and interact with the robot, which enforces the idea that emotional representation in a robot is important to enhance the experience in HRI.

Still in the field of emotional expressions, Paul Ekman [10] tried to identify if persons from different cultures and backgrounds were able to identify a set of human emotional expressions and if each expression was characterized by the same emotion regarding each person's interpretation. Pictures of persons performing emotions such as happiness, sadness, surprise, fear, disgust and anger were shown to each subject that participated in this experiment, and results confirm the hypothesis that each of these six basic expressions can be identified by different persons from different cultures and backgrounds. These expressions were also implemented in some robotic faces to convey emotion for HRI [11], [12]. This emotions were also implemented in the robotic presented in this paper for experimental evaluation.

II. The Robotic Head System

The ISR-RobotHead falls into the LCD based category of robotic heads since it represents expressions through the use of

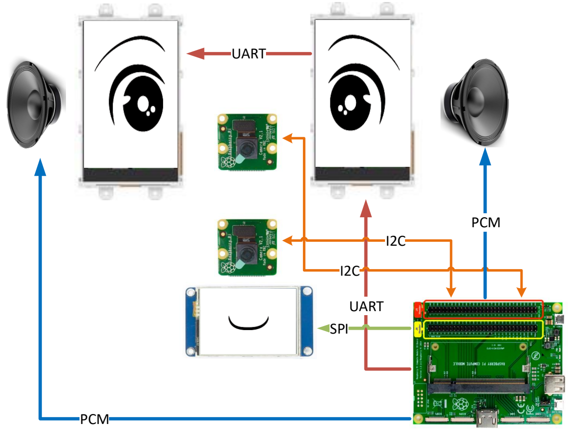


Fig. 1. ISR-RobotHead hardware overview.

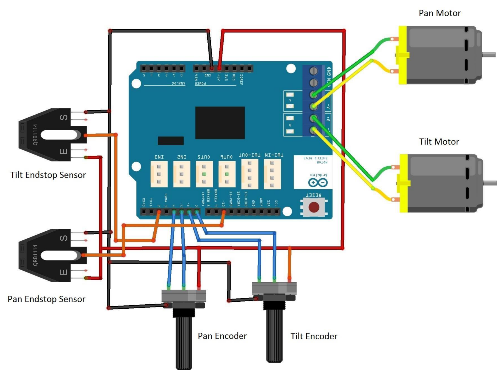


Fig. 2. ISR-RobotHead pan and tilt orientation controller.

LCD screens. This robotic head has also multiple peripherals to enhance the HRI experience, which are: an output sound system, two cameras for stereo vision, a pan and tilt orientation system and three LCDs to represent facial expressions, two for the eyes and one for the mouth. All the aforementioned peripherals (sensors and actuators) are connected to a central control module composed by a Raspberry-Pi with serial and wifi communication.

A. Hardware

The hardware architecture consists in a central acquisition and control module (as illustrated in Fig.1), based on a Raspberry-Pi, connected to each peripheral. The additional hardware components can be divided in four modules: stereo cameras; sound system; pan and tilt; eyes and mouth LCDs.

1) Raspberry Pi control module: All tasks related to communication, control and sensor interfacing are carried out by a Raspberry Pi Compute Module (RPiCM). The RPiCM is a small and cost effective computer that operates with a Debian based Linux operating system and has multiple General Purpose Input/Output (GPIO) ports available for interaction with external devices and sensors (by default it supports UART, SPI and I2C). The RPiCM operates at a minimum speed of 700 MHz and has 4 GB of flash storage, two Micro USB ports (for power and communication), two Camera Serial Interfaces (CSI) and two Display Serial Interfaces (DSI).

2) Stereo cameras: The RPiCM has the capacity to support stereo vision since it has two CSI ports. We adopted 2 standard Raspberry Pi camera modules that can be used to take high-definition frames. Each camera is connected to a CSI port on the RPiCM.

3) Sound system: The sound output system is composed by two modules. A HiFIberry DAC+Lite which is an audio Digital-to-Analogue Converter (DAC) add-on board for the RPiCM and a sound amplifier. The HiFIberry has three output pin outs, left and right sound channels and ground. They are wired to the audio amplifier to the corresponding inputs which then outputs to the left and right speakers.

4) Pan and tilt: The pan and tilt mechanism was created to change the RobotHeads orientation vertically and horizontally. The mechanism uses two 6 V brushed DC motors with a 297.92:1 metal gearbox which are coupled to quadrature magnetic encoders to acquire the motors shaft position, where each encoder adds a reduction ratio of 12:1 to the motor rotation. The pan motor is coupled at the base of the mechanism and rotates the upper plate, where the tilt mechanism is assembled. The tilt mechanism is composed by two different worm teeth wheels so that when the motor rotates clockwise, the tilt's platform moves forward and when the motor rotates counter-clockwise, the platform moves backward. To control the motors coupled to the pan and tilt, an Arduino Duemilanove micro-controller (connected to the RPiCM) with a Motor Shield was employed to drive the two motors (see Fig. 2). The Motor Shield is a dual full-bridge driver designed to drive inductive loads (e.g., relays, solenoids, DC and stepping motors). It drives the two DC motors for pan and tilt with the Arduino board, controlling the speed and position of each one independently by means of PID controllers. To compute the PID gains, the transfer functions for both DC motors coupled with the Pan and Tilt system were approximated to first order systems. Given the motor models, the PID controllers were projected and implemented in the Arduino Duemilanove for each motor to control velocity and position. The references in velocity and position for the PID controllers are given by the RPiCM.

5) Eyes and mouth LCDs: The eyes are composed by two uLCD-32PTU LCDs. The uLCD-32PTU is a compact and cost effective Intelligent Display Module controlled by a PICASO processor. The LCDs have a 240 x 320 pixel resolution with a 16 bit (65k) colour depth, a Serial Interface to operate the module and a Micro-SD card socket for data storage, where the images of the eye expressions are stored. The LCDs for the right and left eyes are connected in parallel to the same peripheral (UART) due to limitations on the base RPiCM operating system. With this solution the RPiCM only needs to send one command to operate both LCDs. However, this poses a constraint since the LCDs will not be able to operate individually because the Raspberry can not send different data

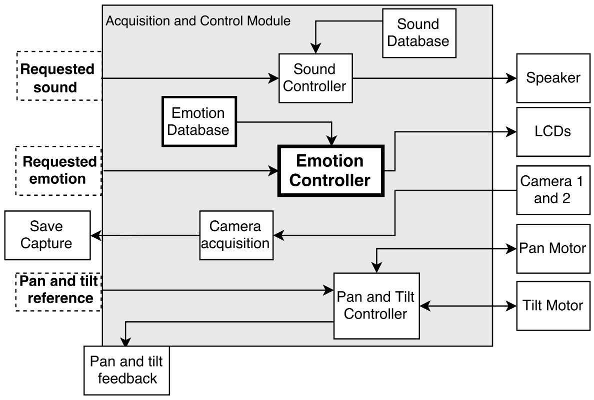


Fig. 3. High level software overview.

to a single LCD. To display eye expressions, each pair of left and right eye images must be uploaded into the respective micro-SD cards in an orderly sequence and the RPiCM only needs to load the uploaded images characteristics (memory address, images size and name) and send commands for the LCDs to display the target pair of images. For the mouth display a ITDB02-1.8SP was used, which is a 1.8" TFT LCD module with 65K colours and 128 x 160 pixel resolution. The mouths module uses the Serial Peripheral Interface protocol. The manner in which the mouths LCD follows a similar procedure/routine as for to that of the eye module. It receives a set of commands and its arguments and acts accordingly.

B. Software

The software implemented in this project was mainly developed for the purpose of generating emotional expressions by the robotic head. The implemented software packages controls the LCD peripherals to display a set of previously loaded images. Each image, stored in a data base, represents a given facial expression (see Fig. 3). The images, however, must be encoded beforehand in a raw 16 bit format and organized in a database to represent expression. Each expression is scripted into the RPiCM with a sequence of images to create animations for each expressions.

The other peripherals (stereo cameras, sound output) were configured and are operational however, at the current stage of the project, the results using stereo and sound are very preliminary and will be not addressed in this paper.

III. Experimental Evaluation

A. Method

The convenience sample for this study was collected in February 2017, at the Cerebral Palsy Association of Coimbra in Portugal and at a private clinical centre for children and youths (Psikontacto, Lda.), and included 9 children aged between 6 and 10 years old: most of them were referred to the services because of some form of psychological disorder (n = 7); 3 of these children had some degree of intellectual disability; and 2 of them had a neurodevelopmental condition. Children were presented the robotic head (see Fig.4) and

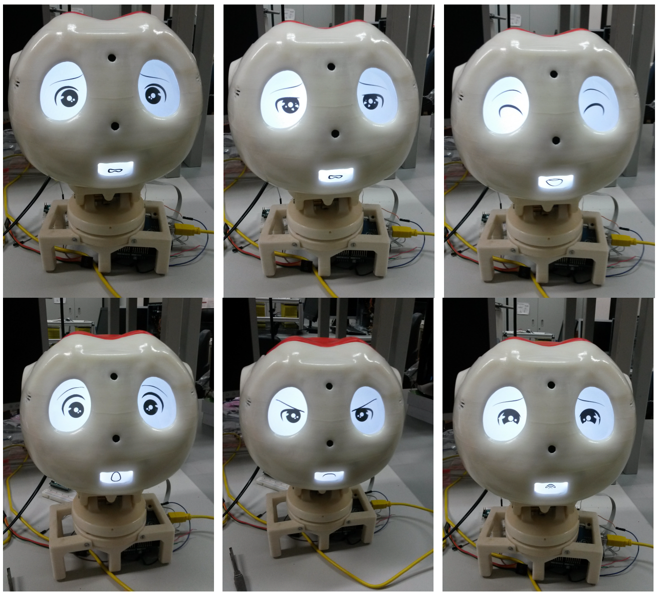


Fig. 4. Pictures of the ISR-RobotHead representing a set of encoded emotional expressions. On top from left to right fear, sadness, joy. On bottom from left to right surprise, anger and disgust.

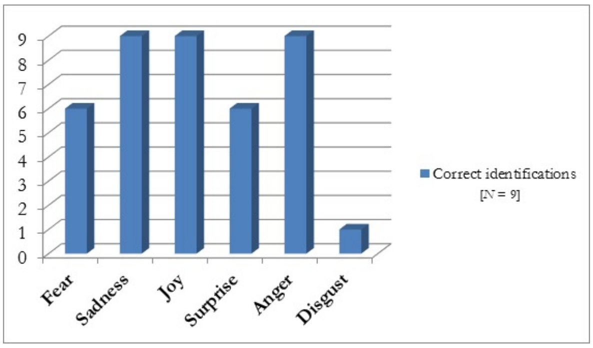


Fig. 5. Frequencies of correct detection of robotic emotional expression by children.

asked to identify six emotional states under examination (fear, sadness, joy, surprise, anger, and disgust).

B. Results and discussion

As depicted in Fig. 5, all participants correctly identified the emotions of Sadness, Joy, and Anger, thus attesting the ability of this LCD-based robotic head for non-verbal communication of these emotions. Likewise, most participants ($n = 6$) correctly detected the emotions of Fear and Surprise. However, only one of the participants was able to identify the emotion of Disgust, which indicates a clear need of improving the LCD animations created for this particular emotion. Despite the fact that younger children lack the verbal ability to accurately label more complex emotions, we believe that the LCD eyes-eyebrows and mouth animations can be recreated to properly convey this particular emotion of disgust, which is triggered in humans by the threat-detection affective regulation system and is often accompanied by vocalizations in its expression. The small sample size precluded the conduction of comparative analyses between age-groups or different levels of cognitive ability, but nonetheless, these are aspects to be fully taken into account in similar studies to be conducted in the future.

IV. Conclusion

In this paper we presented a new robotic head along with a brief overview of its current hardware modules and functionalities. The results yielded from the experiment were positive since the surveyed children were able to identify positively most of the emotions represented by the ISR-RobotHead. These results give us motivation to continue improving this project in order to improve some of the emotions representations and also to endow the robotic head with automatic and intelligent control.

Acknowledgment

The authors would like to thank Dr. Maria do Céu Salvador, for her kind availability in hosting part of the exploratory study at Psikontacto, Lda.

References

[1] F. Faria, "A study on robotic heads design within the scope of social and emotional expressiveness for intuitive human-robot interaction," Instituto de Sistemas e Robótica-Universidade de Coimbra, Tech. Rep., 2015.

[2] J. Broekens, M. Heerink, and H. Rosendal, "Assistive social robots in elderly care: a review," *Gerontechnology*, vol. 8, no. 2, pp. 94–103, 2009.

[3] A. Lopes, D. Almeida, and U. Nunes, "ISR Robot Head: Mechanical, electronics and design," Instituto de Sistemas e Robótica-Universidade de Coimbra, Tech. Rep., 2015.

[4] T. Belpaeme, P. E. Baxter, R. Read, R. Wood, H. Cuayáhuitl, B. Kiefer, S. Racioppa, I. Kruijff-Korbayová, G. Athanasopoulos, V. Enescu *et al.*, "Multimodal child-robot interaction: Building social bonds," *Journal of Human-Robot Interaction*, vol. 1, no. 2, pp. 33–53, 2012.

[5] A. De Beir, H.-L. Cao, P. G. Esteban, G. Van de Perre, D. Lefeber, and B. Vanderborght, "Enhancing emotional facial expressiveness on NAO," *International Journal of Social Robotics*, vol. 8, no. 4, pp. 513–521, 2016.

[6] M. Mori, K. F. MacDorman, and N. Kageki, "The uncanny valley [from the field]," *IEEE Robotics & Automation Magazine*, vol. 19, no. 2, pp. 98–100, 2012.

[7] B. Robins, K. Dautenhahn, R. Te Boekhorst, and A. Billard, "Robotic assistants in therapy and education of children with autism: can a small humanoid robot help encourage social interaction skills?" *Universal Access in the Information Society*, vol. 4, no. 2, pp. 105–120, 2005.

[8] T. Hashimoto, H. Kobayashi, A. Polishuk, and I. Verner, "Elementary science lesson delivered by robot," in *Proceedings of the 8th ACM/IEEE international conference on Human-robot interaction*. IEEE Press, 2013, pp. 133–134.

[9] A. Bruce, I. Nourbakhsh, and R. Simmons, "The role of expressiveness and attention in human-robot interaction," in *Robotics and Automation, 2002. Proceedings. ICRA'02. IEEE International Conference on*, vol. 4. IEEE, 2002, pp. 4138–4142.

[10] P. Ekman, *Facial expression*. Nonverbal behavior and communication, 1978.

[11] J. H. Karsten Berns, "Control of facial expressions of the humanoid robot head roman," Robotics Research Lab Department of Computer Science University of Kaiserslautern Germany, Tech. Rep., 2006.

[12] K. Itoh, H. Miwa, M. Matsumoto, M. Zecca, H. Takanobu, S. Roccella, M. C. Carrozza, P. Dario, and A. Takanishi, "Various emotional expressions with emotion expression humanoid robot WE-4RII," in *Robotics and Automation, 2004. TExCRA'04. First IEEE Technical Exhibition Based Conference on*. IEEE, 2004, pp. 35–36.