ROYAL INSTITUTE
OF TECHNOLOGY

The Alan Project

JOEL SCHRÖDER
ELIF TOY
JACOB JOHANSSON
VICTOR KARLSSON
FREDDI HAATAJA
ANDERS ÅSTRÖM
DANIEL MOFORS
ANQING DUAN
SANDRA AIDANPÄÄ

Mechatronics Advanced Course
Supervisors: Jad EL-Khoury, Sagar Moreshwar Behere

Abstract

This report covers the initial work of designing a robotic art piece as part of the Mechatronics Advanced course at KTH. The art piece consists of two humanoid robots having a conversation with each other while making gestures and having the ability to make eye contact with people in their proximity. The critical functions of this product have been identified as the ability to track faces, the ability to move the arms in a human manner and speech between robots. Different ways to realize this has been examined through a state of the art study as well as a pre-study, resulting in solutions and functional prototypes that lead the way into the next step of the project. This autumn the robots are to be realized into a final, functioning product.

Abbreviations

W	Watt
dB	Decibel
Hz	Hertz
DAC	Digital-to-analog converter
PWM	Pulse-width modulation
PCM	Pulse-coded modulation
HOG	Histogram of Oriented Gradients
HOD	Histogram of Oriented Depths
SOTA	State of the art
FPS	Frames per second
PIR	Passive infrared sensors
rpm	Revolutions per minute
DOF	Degree of Freedom
PID	Proportional-integral-derivative
GPIO	General-purpose input/output
I2C	Inter-Integrated Circuit
EMI	Electro magnetic interference
Arduino	Family of microcontrollers
Raspberry Pi	One card computer
OpenCV	Open Source Computer Vision Library

Contents

Contents

Chapter 1

Introduction

The authors of this report are the team of students that have taken on the task of realizing the art piece as part of the project course MF2058 & MF2059 Mechatronics, Advanced Course. This report covers the work done on the project by the team during the spring semester of 2015, and sets the basis on which the work will proceed the following semester.

1.1 Background

In the spring of 2016, an art exhibition is going to be held at the Manchester Art Gallery in the UK in memory of computer scientist Alan Turing. The exhibition is organized in the frame of a science program arranged in Manchester, which will be the European City of Science in 2016. Named "The Imitation Game" after a phrase used in one of Turing's most famous and influential works, the exhibition will feature contributions made by ten artists. These ten installations all express in different ways what happens in the meeting between humanity and technology. The Swedish artist Tove Kjellmark, whose art pieces based on research of another nature refusing the difference between human and animal nature and difference between mechanics and organics, would like to create two human like robots, which contain and merge both human and robotic features and attributes for the exhibition in Manchester. Alan project team containing ten skilled master students helps Tove Kjellmark to make her idea real as part of their advanced mechatronics course.

1.2 Project description

The main purposes and goals for the project product are identified as:

- Realizing the artist's vision and express her ideas in a successful way

- Possessing human/humanoid robot qualities and behavior in both physics and movement
- Interacting with the visitors, i.e. both sensing and addressing them in a satisfactory manner
- Being a well-designed, well-produced and well-documented complete product that will function in a real environment with all its challenges

The installation will consist of two humanoid robots seated in armchairs, having a conversation that can be interrupted and resumed. The robots will show some human-like behavior, gesticulating in a human way as they talk, with as varied and expressive movements as possible. If a spectator approaches the robots and comes close enough, one of the robots will sense this and turn head and eyes to and make eye contact with the spectator and ask him or her not to disturb their conversation. If the exhibition area gets too noisy, at a certain sound level the same robot will sense this and be able to ask the visitors to keep the volume down while making a hushing motion. As the robots should be on and functioning in a public area continuously for many hours a day, for several months, they must be built in a very robust way. As it will be transported, it should be easy to assemble and dismantle as well as repair. The robots will also have integrated esthetic features made by the artist.

1.3 Requirements

The requirements for the robots have been derived and elaborated during conversations with Tove Kjellmark, the projects number one stakeholder. For the more ambiguous requirements, new requirements have been constructed that are easier to verify, construct tests for and reduces the uncertainty in the interpretation of them. The second part of requirements below consists of these.

1. **Stakeholder Requirements**

   1.1. The robots shall be able to moving its

      Arms

      Head

      Eyes

   1.2. The movement of the robot shall be stable

      no vibrations

      no shaking

      human-like

   1.3. The robot shall be able to produce high quality sound from pre-recorded audio files

1.4. The recorded conversation shall be able to be paused if the robots are interrupted by

audience who get closer than 1,5 meters

audience who get to loud

1.5. The robot shall be able to interact with the the one who get to close by

looking the interuptor in the eyes

say something from a pre-recorded file

1.6. If the robot is interrupted by a loud audience, it shall turn its head toward the crowd and say something from a pre-recorded audio file

1.7. The robots shall be able to continue the recorded conversation from the beginning of the last said sentence

1.8. The sound produced from the speakers shall come or appear to come from robots mouth

1.9. The conversation shall be easily distinguishable within a radius of 5 meters

1.10. Documentation shall be provided to every specific feature in respect to installation, start-up and to ease trouble shooting

1.11. The robots shall be easy to assemble and dissemble with clear guidelines provided on how to do so

1.12. Extra parts shall be provided for critical parts that has a risk of breaking or risk of being worn out

1.13. The robots shall be durable and robust

safety measures for the robots more fragile part shall be taken into account

2. **System requirements**

This section contain the more ambiguous requirements from last section, rephrased to make them easier to verify and construct tests for.

2.1. **distanceToNearestPersonAccordingToUltraSound**[cm] shall not deviate more than +- 10 cm from **ActualDistance**[cm]

The distance from where people shall be detected has an upper limit of 1,5 meters

2.2. Spectator shall be detectable within the **designatedInteractionArea**[degrees] of +- 25 with allowed error of 5 **measurementError**[degrees]

2.3. If the microphone, located in the robot, measure 65 dB or more, the robots shall be interrupted

65 dB corresponds to the sound level of a normal conversation

2.4. The head should be able to rotate 180 degrees

2.5. The movements of the robots should not contain any overshoot

In order to construct the requirements, the following process has been used.

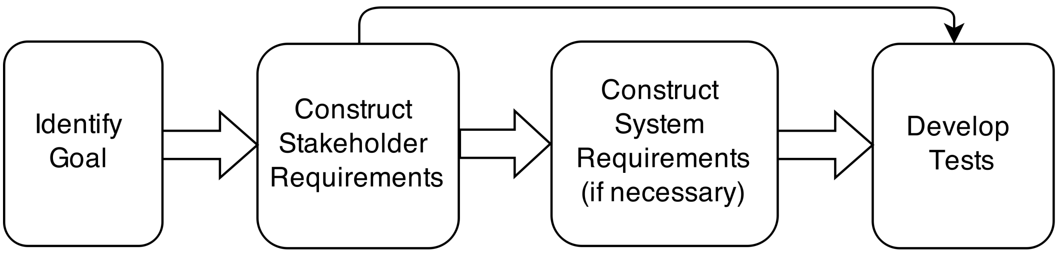

Figure 1.1. Process for working with requirements continously through the project

Due to recent meetings with the Curator of the museum where the exhibition is going to be held, as-well as a representative from the human brain project, Michael Hopkins, these requirements could come change during the start of the fall-semester.

Chapter 2

State of the art

This chapter gives brief information about current high level technologies in face detection and mechanical part of humanoids robots, which will be developed for the project.

2.1 Face tracking

One of the most important requirements of the robots are their ability to interact with people attending the exhibition. One way to achieve this is by seeking eye contact with people who get within a certain range of the robots. In order to achieve eye contact, the eyes or faces of visitors in the exhibition hall needs to be detected, the changes in the location of the eyes should be tracked in real time and coordinates of the closest eyes/face sent to the motors of the eye mechanism, an example of such a mechanism can be seen in the Figure below. Therefore, a state of the art research concerning computer vision with focus on face detection was conducted in order to identify plausible solutions.

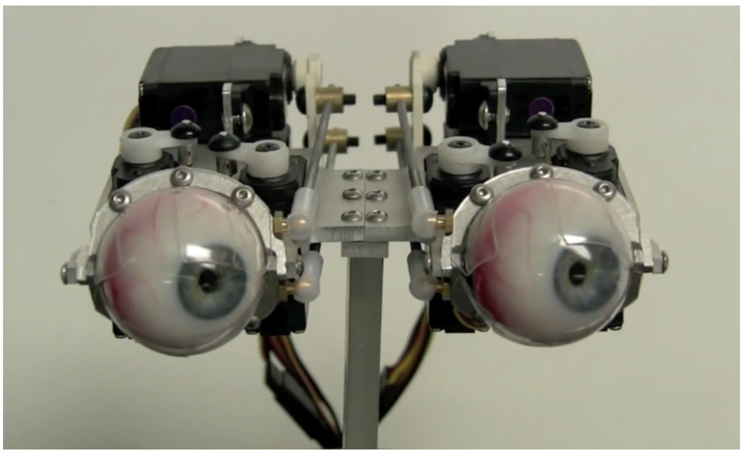

Figure 2.1. A mechanism for enabling actuation of two eyes.

The face detection techniques investigated are face detection in images with controlled background and face detection in unconstrained scenes.

2.1.1 Face Detection in Images with Controlled Background

This technique is based on detecting the face by eliminating the static background. The most common techniques in this frame are face detection by color, by motion or by combination of the two. In face detection by a color technique, the face can be detected by accessing the color pattern in images and compare them with typical face patterns. Although this technique is very simple to implement, it is color dependent and it does not work very well with all skin colors or under all lightning conditions. Face detection by using a motion technique has the assumption that the human face will move continuously. This method cannot differentiate other moving objects than humans. Since the aim of the product is to be able to detect faces with any kind of attribute (such as skin color) in any kind of environment, these methods are not investigated further.

2.1.2 Face Detection in Images with Unconstrained Background

There are two common techniques used to detect images in unconstrained background: weak classifier cascades and model based face detection.

Weak classifier cascades:

This method is based on machine learning which process images quickly and with good update rates. In this method Haar-like features are used which require less computation than working only with RGB pixel values. These features deal with in-class variability problems and reduces the rate of false detection in unconstrained backgrounds.[1] A Haar-like feature is defined by a adjacent rectangular region in an image, see Figure below. The pixel intensities in each rectangular region are summed up and the features are determined according to the difference between the sums of the adjacent regions. This difference categorizes subsections of an image. The algorithm chooses a number of critical features from the image and classifies them. The algorithm is based on AdaBoost, which is a boosting algorithm for classifiers. To separate the background (the other objects) from the promising object in order to focus on interesting objects, this method cascades the classifiers[2]. The main advantage of this method is that the processing of every image happens almost simultaneously and does not differentiate minor discrepancies in the face like skin color.

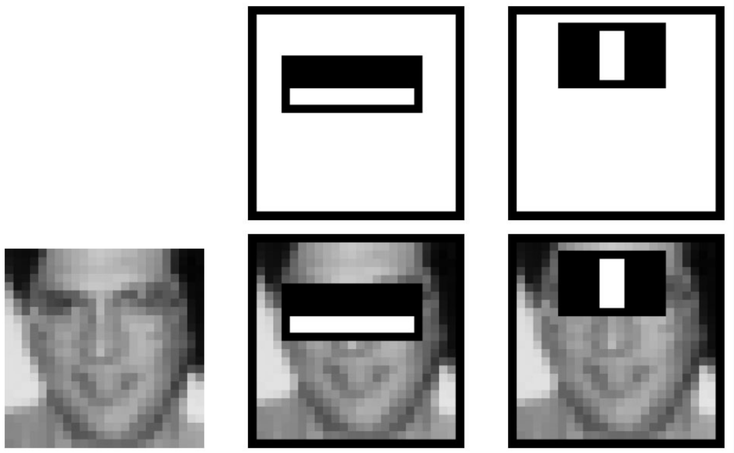

Figure 2.2. Haar-like Features on human face

Model based Face Detection:

The most efficient model-based face detection is based on 3D models. An example of how the software looks while running can be seen in the Figure below. Multi-view face recognition systems are solving the dependency of two dimensional images to pose facial expressions and lighting, and to deal with low resolution in videos. Although this is a really efficient method to detect faces it is computationally expensive.[3]

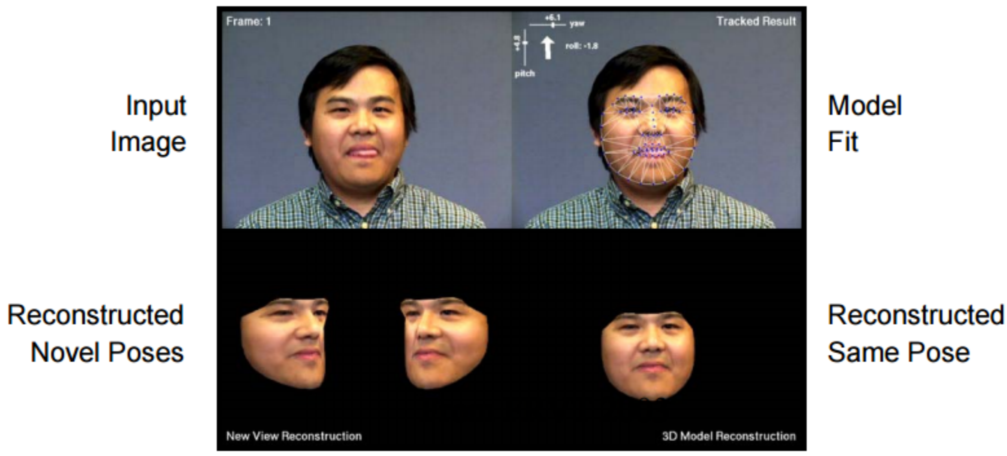

Figure 2.3. Model Based Face Detection

2.2 Human detection and monitoring providing range

Human detection providing range is a key issue for robots sharing space with humans as well for other applications such as autonomous breaking system in cars. A variety of techniques are available. Instead of using range finders such as ultrasound sensors to determine range, cameras can be used to determine distance, known as range imaging [6]. According to Spinello (2011), identification of humans is done using image recognition, such as HOG (Histogram of Oriented Gradients). For determining range, a variety of approaches are available. A novel approach is Histogram of Oriented Depths (HOD). HOD is an image processing method for finding 3D shapes. By calculating oriented depth gradients of objects and by using arrays of local depth changes, a 3D shape can be calculated [6]. Combining HOD and image recognition for people detection, the distance to a person in vicinity to the camera can be determined. This combination, known as Combo-HOD, is particularly suitable for crowded environments [6]. When no depth data is available by using HOD, range finding can function using triangulation assuming an average height of humans detected by image recognition. Triangulation assuming an average height is today used in monitoring system for cars [11]. A Figure describing this technique can be seen in the Figure below.

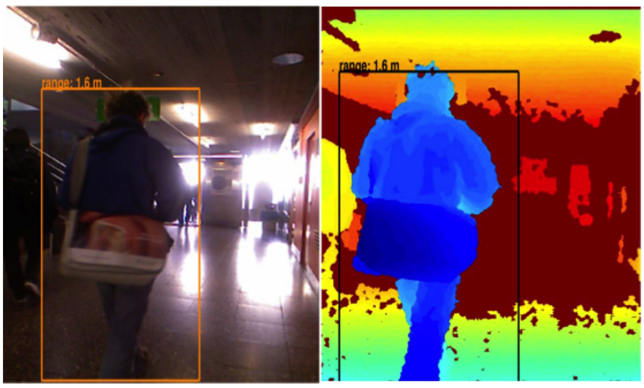

Figure 2.4. People recognition provning range. Triangulation is used assuming an average height using HOG (left) and Combo-HOD (right).

Another common method for providing range imaging, used in for example Kinect sensors, see Figure below, is to project a grid of infrared dots on the objects of interest. Using triangulation the distance to objects is determined [6].

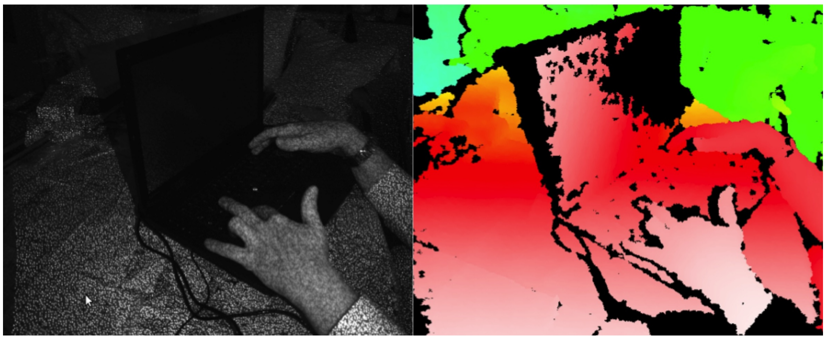

Figure 2.5. Infrared grid provided by by a Kinect camera (left) and the distance visualized using colors gradients.

Chapter 3

Pre-study

This chapter describes the pre-studies of the different sub systems. Its aim is to validate the decisions made during this process.

3.1 Face-tracking

To determine what kind of software and hardware to use in order to perform the face-tracking needed for the eye mechanism to look a person in the eyes, open source software and projects concerning cameras tracking a face are studied.

3.1.1 Software

The software used to perform the face-tracking needs to be easy to use and the code should be easy to acquaint oneself with because the robots won't always be operated by the development team. The programming language Python is a user-friendly programming language and thus will be good to include in the search for plausible solutions. The most frequent and widely used computer vision library in open source projects found on the web, with and without Python, is OpenCV (Open Source Computer Vision Library) and seems like the best and logical solution to choose.

OpenCV

OpenCV is a popular open source computer vision library widely used when constructing similar applications as the one needed for this project. It is free and thus there exists an abundance of examples and forums with help to reach a solution for face-tracking.

OpenCV is released under the Berkeley Software Distribution (BSD) license and hence it is free for both commercial and academic use. It supports most popular operating systems and programming languages. This provides a broad platform of information and pre-made solutions that could benefit the project.

OpenCV along with a trained Haar cascade classifier, explained further in 2.1, was decided to be examined closer in order to determine if it was a good enough solution for performing face-tracking.

From a mechatronics point of view, the different features of the robot should be made in a modular fashion, working independent of each other. From an artistic point of view, one could argue that the heavy calculations being performed in the robots system ought to be made inside of the head. Because of these factors, the possibility to perform the tracking of the face with the help of a Raspberry Pi 2 was researched and tried out.

3.1.2 Hardware

To fit the software, and to perform the relatively heavy calculations needed for computer vision, a proper computer is needed. If the computer is to fit inside of the head, size matters as well. A device that fulfills all of these three requirements is the Raspberry Pi 2.

Raspberry Pi 2

The Raspberry Pi is a computer developed with the purpose to promote teaching of basic computer science in schools. It has a It has a 900 MHz quad-core and 1 Gb of ram. With these impressive specifications for its price, just below 400 SEK, along with having the size of a credit card, it is a great candidate. The machine's operating system is a version of Linux, and can be downloaded for free from the Raspberry Pi homepage.

Raspberry Pi Camera

Along with the libraries in OpenCV and the computer it runs on, a camera is needed in order to produce the stream of images. A camera custom-made for the raspberry pi was used. In the Figure bellow, an image of the camera connected to the Raspberry pi is shown.

Figure 3.1. Raspbeery pi 2 with a connected Raspberry pi camera

It is capable of a frequency of 120 frames per second, up tp 1080p resolution.

This setup, along with OpenCV running a face-tracking python script is able to produce an adequate refresh-rate of 4-5 x and y face-coordinates per second. These Coordinates are then transferred to an Arduino, initially by using I2C. This setup was later changed to the more robust serial communication.

3.2 Motors and motion control

With the stakeholder requirements as a starting point, a study to search out appropriate motor solutions for actuating the different parts of the robot has been conducted. Initially, typical motor types utilized in similar robotic projects and their characteristics have been investigated.

The most common types of motors for this application is the continuous DC-motor and the servo motor. The different important characteristics of the two are listed in table 3.1.
A servo motor runs with a built in position sensing system to rotate a specific number of degrees, usually limited to 180 degrees unlike the DC-motor which rotates freely. This makes the servo motor suitable for robotic applications where this type of motion is desirable. The InMoov [17] robot is a full sized humanoid robot using servo motors as its main source of actuation achieving human-like motional behaviour.

As the DC motor runs at higher speeds, it makes the motor suitable for vehicles and other applications that require high speeds. On the other hand, with an integrated

Characteristics	Continuos DC	Servo
Torque	Has a high rpm (usually around 5000 rpm) to torque ratio. High torque achievable with external gear.[13, 15]	Utilizes built in gear reduction to increase torque. Larger servos can achieve higher torque [14].
Control	Easy to control via computer with relays or electronic switches. Requires external encoder for angular sensing.	Precise angular motion using built in position sensing usually with a potentiometer. Requires special driving circuit.

Table 3.1. Comparison of continuous DC motors and servo motors

encoder and gearbox the DC motor becomes versatile and suitable for robotic applications. Since an integrated encoder with a high resolution provides precise angular position sensing, the design of accurate controllers is achievable. Combined with high gear ratio gearboxes the DC-motor is a viable option for precise and heavy robotic applications. The humanoid service robot REEM [16] focuses on having as human-like traits as possible, achieving this with brushless DC motors.

3.3 Speech

There are several ways to control the speech for the robots. The main functions are to control the speech with inputs and to read specific outputs. To get speech on two different speakers with the same amplifier and by only using one sound output, the playing sound files should be saved as a left/right channel corresponding to the specific robot. Two prototypes are constructed and programmed to evaluate the robustness of the function. To have a good playback the sound file should be at least 32 kHz sample rate to distinguish pronunciation as ”S” and ”L” sounds. Depending on the amplitude of the speaker, the number of bits for the DAC needs to be evaluated to ensure audio quality.

The media player needs to be able to get inputs if it should start playing the discussion, pause because of sensor interruption and play a specific interruption file. The outputs need to be given if a file is currently being played so that the next file can be played once it has stopped.

3.3.1 Hardware

One prototype is using an Arduino that play sound files via an SD-card in the WAVE format. The prototype can play 32 kHz, 8-bit sound files with a satisfactory playback and it is easy to control via inputs and outputs. It is communicating with I2C through another Arduino, but since this is very low level protocol the robustness will not be satisfactory if one message gets lost. A secondary prototype for speech playback is designed using a Raspberry Pi 2 via a Python program. With this solution it is possible to play mp3 files which would reduce the amount of storage needed for the sound files.

The amplifier and speakers needs to have enough power to produce the sound needed to meet the requirement. A 5 W amplifier is used with a 20 W speaker to produce sound. The sound being played is in good quality but the amplification may be a bit to low to meet the requirement of audible sound level at five meters. To ensure a robust design, a manufactured amplifier should be used to reduce influence of EMI to the speakers.

3.3.2 Software

The Python script for the audio playback uses a function *subprocess* defined in the Python language which will start a new process of the media player OMXPlayer. This makes the audio player process run in the background with the Python script still running. To make the speech function user friendly each sound file corresponding to the specific robot will be stored in subdirectories, and the playback order is listed in a text file. When the program starts it reads the text file and saves the name of the file in a linked list. When a sound file is done playing, it is removed from the list and the next one will be played. This way a sound file can be repeated if it was interrupted by sensors. Once the linked list is empty, the text file will be opened and re-read. This makes it easy to change or modify the discussion without having to re-program the Python program.

To make the function work with only one audio output, each sound file must be formatted to only send sound to the left or right channel corresponding to the speaking robot. This will give the end-user more work to get the playback correctly but will make the function more robust.

3.4 Sensors

The robot is to detect if spectators are standing to close to the robot or if the audience is talking to loudly. Here, possible solutions are presented.

3.4.1 Distance sensors

For detection of people in vicinity of the robot, the following alternative concepts have been considered

- Active infrared sensor
- Passive infrared sensor
- Ultrasonic range finder

Active infrared sensors determine distances by transmitting a beam of infrared light and determining the distance using triangulation [12]. Due to the narrow beam [12], a great number of beams, i.e. sensors, would be required to monitor a wide area.

Passive infrared sensors (PIR) function by measuring emitted infrared light from their surroundings [7]. Due to their way of operating, the PIR sensors are not very accurate in terms of determining range and can be disturbed by objects in its surroundings emitting heat [7].

Ultrasonic rangefinders work by transmitting a burst of ultrasound. By measuring the time for the emitted sound to be reflected back, the distance to the nearest object is determined [8]. Since emitting a cone with a spreading angle up to 40 degrees, an ultrasonic range finder can monitor a greater area than an active infrared sensor for example. Humans have a relatively small effective reflection area for ultrasonic sound waves. This is since soft objects such as humans and clothing absorb the sound waves. A human typically reflects the same amount of ultrasonic energy as a 6 cm^2 dowel [8].

3.4.2 Sound level measurement

With regards to sound measuring, different microphone types exist. A common and simple type of microphone is the so called Pressure microphone [10], that measure the pressure caused by acoustic waves. The voltage produced is proportional to the acoustic pressure [10]. Since the voltage from the membrane is low, an amplifier is usually implemented to amplify the signal before the sound level is measured, i.e. the voltage [10].

3.5 Mechanical features

In order to actuate the moving parts of the robot, various types of actuators are studied, such as motors in the joints, timing belts, gears, hydraulics (or pneumatics), and drop arms. Implementing motors directly in the joints can simplify the work a lot because no torque or force transferring system will be needed. One problem for this alternative can be that the motor per se will increase the weight of the moving parts and thus become extra load for the motor at the higher level. The motor with built-in gear box should be considered in this case in order to get enough torque output. Timing belts, V-belts or chains are typically used in the condition where the two shafts are far apart. This mechanical structure is simple, inexpensive and has high efficiency. But there will be some movement error when using V-belts. Gears can also be used to transfer torque or movement. The precision of the motors and gears should be high to avoid friction losses and maintain control of the robot. When using hydraulics (or pneumatics), there will be risk of leakage. In addition, it is used more widely in heavy mechanical equipment instead of small robot without heavy load. For drop arms, it avoids that motors are implemented inside the arms and the rod shall be actuated by the servo or DC motor.

Chapter 4

Critical functions and prototypes

In order to verify the conclusions made during the pre-study, simple prototypes were made. This clarifies what seems plausible to achieve during the fall and what kind of modifications that needs to be performed in order to reach the stakeholders requirements.

4.1 Face-tracking

For face-tracking a code in python using OpenCV libraries is developed based on Haar-cascade classifiers. The coordinates are produced from the code and are sent to Arduino via serial communication. In order to verify that the coordinates are transferred correctly at a sufficient speed, a pan and tilt device using micro servos was 3D-printed and assembled, see Figure below.

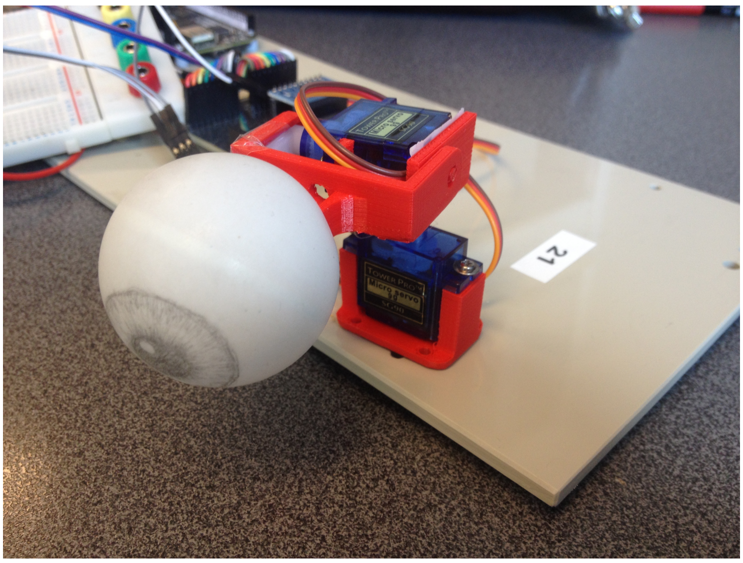

Figure 4.1. Haar-like Features on human face

The devise uses two **SG90** micro servos to pan and tilt and **PWM**-signals for the mechanism are provided from an Arduino Mega.

At first **a Raspberry Pi 2** was used. It provided close to acceptable motion of the servos when running at a low, **240** times **180** pixels resolution. This work sufficiently enough if the person being tracked is close, around **1 meter** from the camera, but has a hard time finding faces further away.

To examine the full potential of the software, the **Raspberry Pi** was switched to a **regular PC with much stronger computational power**. Uncompromisingly, the PC provided a much higher rate of detection, at close to three times the resolution.

The device produced a proof of concept that this method could be used and implemented in the robots in order to make them able to look a person in the eyes.

4.2 Arms

The critical function of the mechanical structure is mainly to offer a framework for the motors and to move within some range at certain numbers of degrees of freedom. The appearance of the mechanical structure is also one of the critical functions, while it would not be considered in our current simple prototype. The relative friction of the mechanical structure should be as small as possible when the robots are moving. The mechanical structure of the arm is finished and the motors are implemented at the joints directly without any mechanical driving system involved. One motor is fixed to the wood board by a motor rig as the shoulder of the robot. Its shaft drives the upper arm with a motor hub fastened in between. The other motor is fixed to the upper arm as the elbow joint. The shaft of the elbow motor drives the lower arm of the robot. The choice of the material of the arm is wood and is manufactured by laser cutting machine. In order to make the prototype look more realistic, a skeleton hand is attached to the lower arm as decoration. When the motors are controlled to make the arm stretch horizontally, the upper arm suffers the maximum stress due to the weight of the other parts. With verification, it turns out that the arm is robust enough and capable of holding the load.

4.2.1 Results of the Arm Prototypes

The elbow prototype is unnecessarily heavy due to the skeleton parts. The idea of having a robot built upon a skeleton model is to get an already finished frame of a robot. Due to the way of connecting the actuators to the joints of the skeleton, the skeleton parts are more of a decoration than a frame. In this case the prototype is striped on skeleton parts and has thereby reduced weight. To have a skeleton as a frame in the first place turns out to be an improper idea. One of the questions answered with the elbow prototype is: How heavy does the arm need to be? The answer is it should have relatively low weight. This is the beginning of a new prototype: the arm prototype. It has two degrees of freedom, one in the shoulder and one in the elbow. The arm prototype has a platform so that it can lie on a table. The weight of the arm prototype is lighter than the elbow prototype. The prototype shows parts of the hushing motion that the end product robot can perform.

4.3 Mechanical

The manufacturing and assembling phase of the mechanical parts of the project should be finished by the end of October. What are needed to be done are designing, order material or parts, manufacturing and assembling all the mechanical parts of the robots such as the upper body, including neck and arms. For autumn the plan is also to write an assembly manual. During the autumn, also the project documentation and reporting part will be highly prioritized. The final mechanical functionalities of the robots are movement of shoulder (with 1 Degree of Freedom, DOF), elbow (1 DOF), finger (only one finger on each hand will be moving, 1DOF) and neck (1 DOF). If time permits also torso will be considered as a movable part with 1 DOF. The plan is to use plastics (ABS and PLA) and aluminum as construction material of the robot parts. A work breakdown structure has been made and will be used during the autumn phase of the project. The work breakdown structure regards the overall functionalities of the final product and a time estimation of the different tasks needed to be done.

4.4 Motor control

One of the critical functions is defined as having a robotic arm able to perform a hushing motion, i.e. moving the arm from free hanging position to put an index finger against the mouth. This critical function is demonstrated through a two DOF prototype, showing the basic functionality of rotating the arm in two joints to move the hand towards the head as depicted in Figure 4.2.
This movement is realized with two DC motors equipped with integrated quadrature encoders, controlled with a driver board and an Arduino mega 2560. These motors are chosen for their high torque to weight ratio. Due to the high resolution of the encoder accurate control can be achieved with the help of a trajectory plan and a velocity controller. These are implemented in C code and run on the Arduino.

4.4.1 Simulink model

To get sufficient insight into the required motor specifications in order to achieve proper actuation of the arms, a model of the system has been designed in Simulink as depicted in Figure 4.3 prior to the motor selection.

From the model, an estimation of the required torque is obtained to determine whether a motor is sufficient or not together with the corresponding control signals. In the model a PID controller is implemented and tuned to produce a smooth

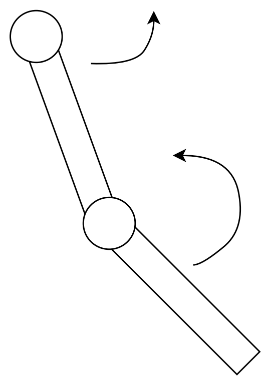

Figure 4.2. Rotation of the DOF arm prototype

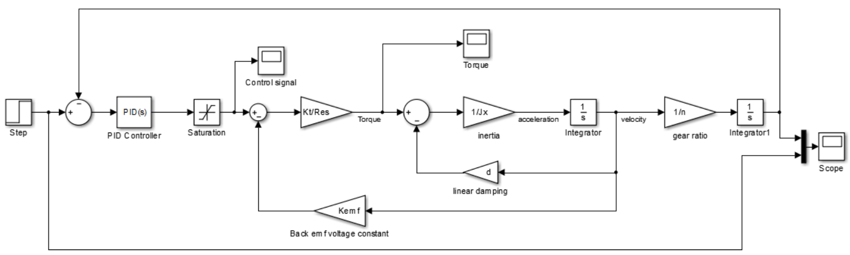

Figure 4.3. A model simulating control and actuation of the robot arm made in Simulink.

response. A resulting step response from 0 to 180 degrees of the upper arm motor is depicted in Figure 4.4.

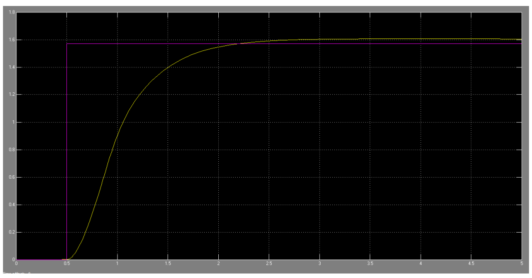

Figure 4.4. Step response from 0 to 180 degrees of the upper arm motor

4.4.2 Code implementation

The resulting control implementation on the Arduino consists of a trajectory planner and a velocity controller to actuate the arm in the specified motion. A block diagram of the code logic is depicted in figure 4.5.

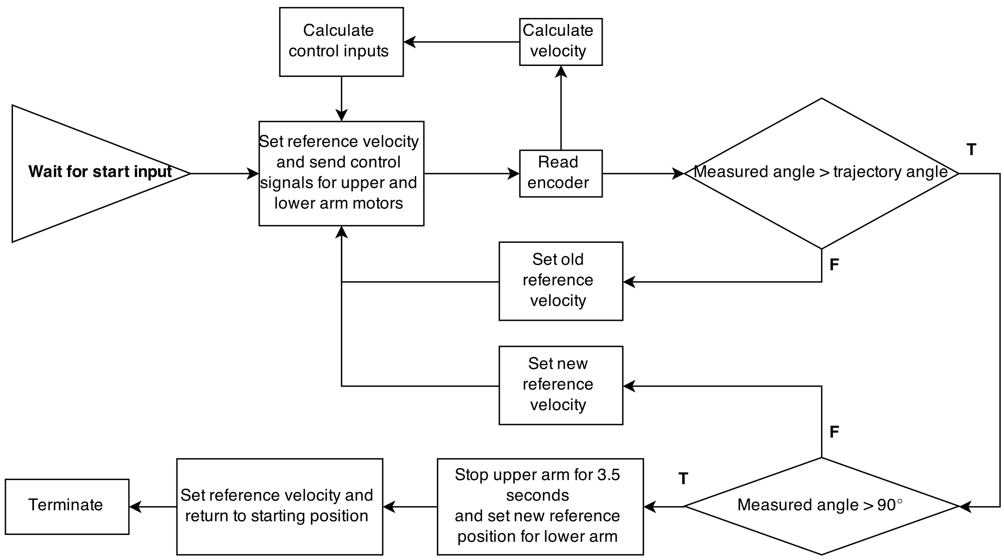

Figure 4.5. A model simulating control and actuation of the robot arm made in Simulink

4.5 Speech

To be able to ensure that the robots' speech function is a possibility, a critical function prototype is constructed. The prototype uses Raspberry Pi as a platform to ensure satisfactory playback quality and robustness. Two passive speakers are connected to the 3,5 mm output of the Raspberry Pi via a 5 W amplifier, representing the two robots. A Python program is created to control OMXPlayer, which is used as the main software for the audio playback. By putting the pre-recorded audio files in a linked list the program is able to play the files in a correct order while knowing which file that was recently played.

To simulate the robots, speaker one is connected to the left channel and speaker two is connected to the right channel. The sound files are later on exported as a mono sound using left or right channel, not both. This enables the possibility to choose which sound file that should be played in speaker left and which that should be played in speaker right. A text file is then created to pre-define the order of the files, making it easy to control the conversation going back and forth between the two speakers (robots).

A pre-recorded conversation is now held between the two speakers. To integrate the requirement of the robots being interrupted when a person gets too close, an ultrasonic sensor is connected to one of the GPIO pins of the Raspberry Pi. This sensor detects if something is inside a specific pre-defined distance. When so, the sensor sends a signal to the GPIO input of the Raspberry Pi, telling the program to interrupt the conversation between the robots. A specific interrupt sound file is played saying: “Please don’t disturb us, can’t you see that we are having a conversation here?” After the interrupt sound file is finished, the file previously played before the interrupt is then played again and the conversation continues.

4.6 Speech

Although the Raspberry Pi is sufficient for audio playback, a more suitable solution would be to use a laptop. Since a laptop already has an integrated screen, keyboard and touch-pad, programming and reprogramming is a lot easier. The laptop is already going to be used as a node for communication between the various functions of the robots, which means that it is only natural using it for the speech as well to minimize the use of unnecessary components such as the Raspberry Pi. By using a laptop as a platform a new media player has to be determined for the audio playback. Since additional modules of the robot are to be integrated with the speech, the program has to be further developed to ensure compatibility between the modules. As the system is continuously updated, tests have to be performed to guarantee the requirements are met.

To make the hardware robust, an enclosure for the exposed components will be designed. This will ensure that fragile parts will not be handled by inexperienced users, it will also give the end-user an easy way to connect cables and control the sound level of the amplifier.

4.7 Sensors

In this section the developed system for detection of people in viscinity and sound level is presented.

4.7.1 Rangefinding

Since people absorb ultrasonic sound, a high performance ultrasonic range finder, Maxbotix MB1000 EZO, is used and evaluated. Tests of the equipment have been conducted with a person of 62 kilos and 177 cm, standing 1.5 meters from the sensor at a height of 1 m measuring horizontally. Different cases have been tested, in which the person is standing in different positions with respect to the sensors and with different amount of sound absorbing material, i.e. clothing.

Tests of the ultrasonic rangefinder show that the sensor is practical for detecting people within a span of about 50 degrees. As appears from the tests, the sensor is fairly accurate for detection a person standing right in front of the sensor independent of clothing. When the pitch, i.e. the angle between the sensors centre line and the person, exceeds 25 degrees, a considerable part of the readings are faulty if heavy clothing is worn. This is due to too little energy being reflected back from the person for the sensor to detect the person. This is not the case when lighter clothing is worn.

The conclusion is that the sensors are suitable for the intended application. However, multiple sensors are required to monitor a bigger area since each sensor only covers a span about 50 degrees for a radius of 1.5 meters. In addition, due to faulty readings occuring when people are not standing right in front of the sensor, multiple readings should be conducted before results are interpreted as a human standing in vicinity or not.

4.7.2 Sound level measurement

For measuring of sound level a pressure microphone integrated with amplifier is used: MAX4466 by Adafruit. Voltage produced by the microphone is proportional to the sound pressure. For monitoring the sound level and alerting if the sound level exceeds the voltage that corresponds to 65 dB the software design can be seen in the Figure below.

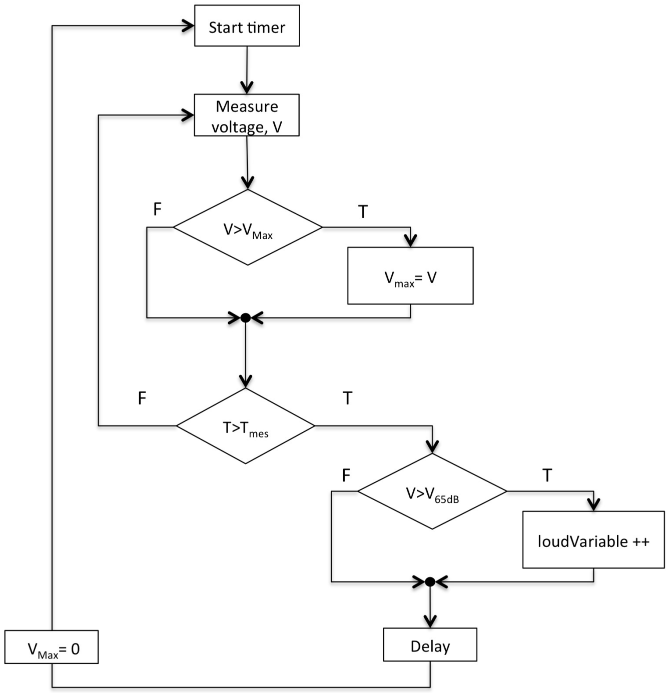

Figure 4.6. The function for measuring of sound level

To avoid transient high sounds such as smashing from doors to be interpreted as a audience speaking too high, a series of measurements are conducted within a given time window. The readings occur with a delay in between. Several independent readings must indicate high sound level for the sound level to be interpreted as caused by a loudly speaking audience, i.e. the variable loudVariable must exceed a given integer.

The code and equipment for monitoring the sound level works as intended and transient noise is ignored. However, the gain of the module is very low and therefore the resolution of the readings is low as well. Hence, obtaining a similar module with higher gain would enhance the reliabillity.

Chapter 5

Discussion

During this project a lot of time has been allocated to develop the critical functions. Since the robots have to be finished by the end of December, the critical functions had to be completed as soon as possible to validate their functionality. The team is satisfied with the results of the prototypes that show the critical functions of the robot. A challenging part of the project is that the stakeholder is an open-minded artist who has a habit of changing design requirements during the process. This has lead to change in the direction of the development and to ensure effort is put on the correct task, certain aspects of the design need to be locked. The stakeholder is aware of the time constraint of the project and decisions regarding designs should be easier to make now that the demonstrators have been shown.

Due to insufficient project planning in the start of the project along with suggestion to strongly focus on development of prototypes and to deliver results, little time was distributed to research. This has resulted in poorly based design decisions and a frequent change of track. But despite the hardships and lack of research, the prototypes have provided insight in how to continue the development next semester. The mechanical part should have highest priority so assembly and construction of the robots can be initiated. If testing of the critical functions is completed early, time can be distributed to optimize the functions and the overall design of the robots.

For future projects in need of quick developed prototypes, assurance of enough project planning and development strategies must be made to ensure that the development goes in the right direction.

Chapter 6

Future Work

The plan for next autumn is to finish the project at the end of December. The final product consists of two robots communicating with each other. The manufacturing and assembling phase of the project should be finished by the end of October. What are needed to be done are designing, order material or parts, manufacturing and assembling all the mechanical parts of the robots such as the upper body, including neck and arms. For autumn the plan is also to create an assembly manual. During the autumn, also the project documentation and reporting part will be highly prioritized. The final mechanical functionalities of the robots are movement of shoulder (with 1 Degree of Freedom, DOF), elbow (1 DOF), finger (only one finger on each hand will be moving, 1DOF) and neck (1 DOF). If time permits also torso will be considered as a movable part with 1 DOF. The robots will use sensors such as ultra-sonic sender and receiver for distance measuring, microphones for measuring level of the sound in the room and cameras with face tracking functionalities that provides coordinates for movement of the eyes of the robots. A work breakdown structure has been made and will be used during the autumn phase of the project. The work breakdown structure regards the overall functionalities of the final product and a time estimation of the different tasks needed to be done.

Bibliography

[1] Papageorgiou, Oren and Poggio, "A general framework for object detection", International Conference on Computer Vision, 1998

[2] Viola and Jones, "Rapid object detection using a boosted cascade of simple features", Computer Vision and Pattern Recognition, 2001

[3] Lu, Colbry. "Three-Dimensional Model Based Face Recognition*, Pattern Recognition, 2004. ICPR 2004

[4] http://apcmag.com/arduino-project-5-digital-audio-player.htm/
Last visited: 2015-05-18

[5] http://www.arduino.cc/en/Tutorial/SimpleAudioPlayer
Last visited: 2015-05-18

[6] Spinello L, People Detection in RGB-D Data, 2011

[7] Mathas C, Sensing Motion With Passive Infrared (PIR) Sensors, 2012

[8] Bonar T, Ultrasonic and People Detection, 2012. Found at: http://www.maxbotix.com/articles/037.htm

[9] New ASIMO, Mobile Robot modeling, Raphael Cardinaux, Ecole Polytechnique federale de Lausanne, 2014 https://www.cyberbotics.com/reports/asimo.pdf

[10] Wallin H P, Ljud och Vibrationer, KTH, 2001

[11] Dollar P, Pedestrian detection: An Evaluation of the State of the Art, IEEE TRANSACTIONS ON PATTERN ANALYSIS AND MACHINE INTELLIGENCE, 2008

[12] $https://www.societyofrobots.com/member_tutorials/node/71$
Last visited: 2015-04-18

[13] http://www.robotoid.com/howto/choosing-a-motor-type.html
Last visited: 2015-05-22

[14] http://www.modmypi.com/blog/whats-the-difference-between-dc-servo-stepper-motors
Last visited: 2015-05-22

[15] http://handyboard.com/hb/faq/hardware-faqs/dc-vs-servo/
Last visited: 2015-05-22

[16] http://www.micromo.com/applications/robotics-factory-automation/humanoid-service-robot
Last visited: 2015-05-22

[17] http://www.inmoov.fr/
Last visited: 2015-05-22

Chapter 7

Appendices

7.1 Appendix A

Alan project
Risk analysis
2015-04-23

Face tracking – Risk analysis

Cause	Effect	Severity	Probability	Product	How to avoid	What to do
Loose wires	Loss of datapackages and or signals	5	2	10	Clamp cables and ensure good connectivity and use good connectors. Maybe keep cables inside a box.	Try to find the cable that is loose and plug it back in. Or perhaps get a new cable/connector
Servo malfunction	Produce weird results for the eye movement	3	2	6	Use stable servos of good quality and make well-constructed test to verify them	Try a different servo, restart system
Face tracking software crashes	Eyes won't move	5	3	15	Construct well defined tests to verify the software	Restart system, try to conclude the cause of the crash
Power supply issues	Face tracking stops to work	5	2	10	Well-connected wires, steady power supply correctly adapted for each system	Check wires, power supply, measure voltage..
Camera malfunction	No face tracking	5	1	5	Make sure the wiring is correctly done and with sufficient length to move around	Reconnect, restart system, (change camera?)
I2C issues	Incorrect communication	4	2	8	Keep wiring as short as possible, I.e. keep the Arduino close to the pi	Check wiring, restart system, check ports so that SDA and SCL are working as they should

7.2 Appendix B

Alan project
Risk analysis
2015-04-23

Speech – Risk analysis

Cause	Effect	Severity	Probability	Product	How to avoid	What to do
Loose wires	Loss of data packages and or signals	5	2	10	Clamp cables and ensure good connectivity and use good connectors. Maybe keep cables inside a box.	Try to find the cable that is loose and plug it back in. Or perhaps get a new cable/connector
EMI on speaker cable	Noise signal out from the speakers.	2	2	4	Keep speaker cables away from high voltage cables.	Change position of cables.
Wrong names on files	The sound file will not be played. Perhaps the program will crash.	3	2	6	Be careful and double check that it is spelled correctly	Remove the SD-card and fix the naming error.
Program crashes during runtime	No sound will produce and maybe the whole robot stops functioning	5	1	5	Do rigorous testing of the software.	Try resetting the whole robot. Otherwise it needs to be reprogrammed and retested.
PC restarts	One sound file while not be played or dialogue will reset.	3	1	3	Have a reliable power supply.	Nothing, maybe check if the power supply is misbehaving.

7.3 Appendix C

Alan project
Risk analysis
2015-04-23

Motors – Risk analysis

Cause	Effect	Severity	Probability	Product	How to avoid	What to do
Loose wires	Motional disfunction	3	3	9	Clamp cables and ensure good connectivity and use good connectors. Maybe keep cables inside a box.	Try to find the cable that are loose and plug it back in. Or perhaps get a new cable/connector.
Burnt windnings	Motor breakdown	4	2	8	Make sure motors run under specified circumstances and have robust controllers.	Replace motor/motors.
Incorrect controller inputs	Uncontrolled motion	5	2	10	Design robust controllers by careful testing.	Restart robot, unplug and replug motor.
Program crashes during runtime	Stopped motion / uncontrolled motion	3	2	6	Do rirgurous testing of the software.	Try resetting the whole robot. Otherwise it needs to be reprogrammed and retested.
Physical breakdown due to interference from spectators	Displacement or destruction of electrical and mechanical components	4	3	3	Protects with casing and have clear and visible instructions. Robust construction.	Spare parts and assembly manual.

7.4 Appendix D

Alan project
Risk analysis
2015-04-23

Mechanical Movement – Risk analysis

Cause	Effect	Severity	Probability	Product	How to avoid	What to do
Loose screws	Unintended movement. Parts starts to fall off	1	5	5	Loctite can be used to lower the risk of screws getting loose	Try to find the screw that is loose and put it back to its position or get a new screw of same size and length and put that back to the screw position
Cracks in solid components	Risk of parts falling apart	2	3	6	Inspect the mechanical parts daily when in use	Order or manufacture a likewise part
Robot overturns or tip over	Parts risk to get deformed or fall apart	4	5	20	Check that the robot is mounted to the chair in a proper way	Tilt the robot back to its original position and inspect the robot for visible damages
Interference from spectators	Permanent deformations, overstressed parts, parts falling apart	2	5	10	Protect the robots with casings	Repair, order or manufacture new parts
Corrosion	Change in color or change in abrasion resistance	1	4	4	Keep the robots stored indoors and in room temperature. Never exposed for moisture or sunshine for longer time periods	Replace the damaged parts with new parts

7.5 Appendix E Project Time Plan

		07-maj	11-maj	14-maj	18-maj	21-maj		Sept	Octo	Nove	Dec
Project time plan											
Speech	30										
- Playback	14										
- Actions	8										
- Play	1										
- Stop	1										
- Repeat	1										
- Specific files	1										
- Selected speaker	4										
- Continious	3										
- Check if playing	1										
- Linked list order	2										
- Initialization	3										
- Read order	1										
- Read folders	1										
- Debug	1										
- Interraction	9										
- Interrupt signal	3										
- Noise	1										
- Proximity	1										
- Repeat	1										
- Send to motors	3										
- Noise	2										
- Proximity	1										
- Send to Face tracking	2										
- Stop playback	1										
- Documentation	3										
- Adding of files	2										
- Directories	1										
- Order file	1										
- Troubleshooting	1										
- Combine	4										
Eye Mechanism	134										
-Face Tracking	53,5										
-Find Faces	13										
-Apply an appropriate algorithm	10										
-Good enough update	2										
-Coordinate production	1										
- Communication	14,5										
- PWM to servos	1,5										
- Eyes	1										
- Neck	0,5										
-Raspberry pi with arduino	12										
-Ultrasound to raspberry pi	1										
-Integration of eye, neck mechanism and sensor outputs	5										
-CAD design	30										
-Order real looking eyes	0,5										
- Assemble Mechanism	8										
- Motors	2										
- Printed parts	2										
- Electronics	2										
- Power Supply	2										
- Integrate arduino& pi used by eye mechanism to over all design	5										
- Tweak eye movement according to spesification	3										
- Optimize the code(make it readable, functional)	5										
- Documentation	24										
- Report	8										
- Manual	8										
- Trouble shooting Guide	8										
Motors	**89**										
-Arms & Neck	**38**										
-Control	**18**										
-Trajectory Planner	**8**										
-Reference model	**5**										
-Model following	**3**										
-PID control	**10**										
-Discretization	**2**										
-Write control code for Arduino	**8**										
-Modelling	**10**										
-Motor parameter tuning	**5**										
-Load paramter tuning	**5**										
-Integration	41										
-Actuation in joints	13										
-Attach correctly	3										
-Test and validate robustness in terms of breaking the motors etc.	10										
-Communication	28										
-Serial communication coding	8										
-Robustness	20										
-Start stop function behaviour	4										
-Operational robustness	8										
-Interference handling	8										
-Documentation	10										
-Handling of physical parts	8										
-Transportation	2										
-Assembly	2										
-Operation	2										
-Part list	2										
-Specifications	2										
-Torso?											
Mechanical Movement	355 - - -										
- Arms	115 - -										
- Prototype	30-										
- Design	10										
- Manufacture	15										
- Assembly	5										
- Real Product	85 -										
- design	20										
- order material or parts	5										
- manufacture	30										
- Assembly	10										
- Neck (connection upper body - head)	35 - -										
- design	10										
- order material or parts	5										
- manufacture	15										
- Assembly	5										
- Torso	35 - -										
- design	10										
- order material or parts	5										
- manufacture	15										
- Assembly	5										
- Upper Body	70 - -										
- design	10										
- order material or parts	5										
- manufacture	50										
- Assembly	5										
- Legs	60 - -										
- design	10										
- order material or parts	5										
- manufacture	40										
- Assembly	5										
- Assembly Manual	10 - -										
- Documentation	20 - -										
- Report	10 - -										