B-More-Human
Dynamic Cheering Reactions for Humanoid Football Robots

Bachelor Thesis

Universität Bremen

Leonard Haddad

Matriculation Nr.: 3183338
leonard.haddad@uni-bremen.de

1. Reviewer: Dr. Tim Laue
2. Reviewer: Prof. Dr. Rainer Koschke

March 18, 2022

Disclaimer

I hereby declare that I have solely composed this thesis on my own and have not used any external sources except for the ones denoted as such. All content taken literally or analogously from external works has properly been indicated as such and is included under the Bibliography section.

Leonard Haddad

March 2022
Bremen

Contents

Chapter 1

Introduction

1.1 Motivation

Robot football can be quite entertaining - all these cute little guys running after a ball, desperately trying to get it into their opponents' goal while occasionally tripping and falling only to miraculously get back up on their own - it is quite the experience. Albeit, it can also be quite boring for the non tech-geeky spectators.
While goals and falling robots are quite funny, most of the time it is not really clear what the robots are up to. All a distant observer sees is robots either turning their head left and right while standing still, or running after the ball with each of their eyes blinking in some weird color combination, without understanding the purpose behind these color combinations.
In real football, it can be quite clear what players are up to through their body language while on-camera. This is not the case with robot football, since robots don't naturally show emotions, unless they are explicitly programmed to do so.
Furthermore, the robots do not react in any way when goals are scored. While in real life football the players jump around and roll on the grass, the robots just calmly walk back to their initial game positions and wait for the next round to start, or get on their knees if the game is over, which only further contributes to the emotionally-cold image given to robots in general.

1.2 Goals

The ultimate goal of this thesis, and the follow-up thesis by Mr. Phillip Dolata "Situative Darstellung von Emotionen durch humanoide Fußballroboter", is to solve the abovementioned problems by providing the robots with ways to express emotions, so that a distant observer can both understand what a robot is up to and sympathize with it, seeing it as a "living being" with emotions, rather than a tin-can with a microchip.
However, in order to make a robot's emotions seem natural it should be able to perform its reactions dynamically, both dynamically in sense of picking how it showcases its current emotional stage, as well, as performing the motions associated with the emotional reaction dynamically, since merely playing back pre-recorded motions in a pre-set pattern takes away the sense of realism and eventually becomes boring when the emotional reactions are repeated often enough.
Although there exists lots of research on dynamic robot movements and human-like robot behaviour, such as the juggling robot from Disney [1, Kober et al., 2012], which is intended to directly interact with humans in a fun way, there exists no research on allowing robots to act more human-like during robot football games, even though RoboCup aims to eventually play against real human football players (see section 2.3). Therefor, granting the robots the ability to showcase emotions can help in creating a more realistic game experience in the long run, both for distant observers and for the human soccer players who will play against the humanoid robots.

1.3 Structure

The second chapter gives a brief history of soccer playing robots, an overview of the robots this thesis deals with and the B-Human collegiate project. Chapter 3 then continues with one of the conducted user surveys based on which the robots' emotional reactions were picked. Chapter 4 proceeds to showcase the implemented emotional reactions and the dynamism and mechanics behind these. The evaluative chapter 5 utilizes the second conducted survey to evaluate the reactions described in the fourth chapter. Finally, chapter 6 concludes with future plans concerning the robots' reactions and showcases rooms for improvement. Last but not least, chapter seven includes the bibliography.
Note: In this thesis the expressions "soccer" and "football" are used interchangeably. When talking about American football, it is referred to with the initial "American" before "football". Furthermore, the term "emote" is later used as a synonym for "emotional reaction".

Chapter 2

Background

This chapter gives an overview of the robot model used in this thesis, a brief history of the RoboCup initiative and an overview of the B-Human collegiate project.

2.1 SoftBank Robotics

SoftBank Robotics Holding Corp. is a French robotics company founded by Bruno Maisonnier in 2005. Originally known as Aldebaran Robotics and renamed into SoftBank Robotics on the 19th of May 2016 [2, SoftBank Robotics], SoftBank started out by selling its humanoid robot NAO (discussed below 2.2). It operates in multiple countries around the world, employing hundreds of people in its offices in Paris, San Francisco, Boston, Shanghai, and in its headquarters in Tokyo, Japan.

Today SoftBank Robotics employs a variety of robots in different industries, including healthcare, finance, education and tourism. It continues to employ the NAO (now on its 6th generation) along with its other humanoid robot Pepper, its enterprise smart vacuum sweeper robot Whiz, and SoftBank Robotics Japan's exclusive food delivery robot SERVI [3, SoftBank Robotics].

Figure 2.1: SoftBank Robotics' other robots, Pepper (left)[4] and Whiz (right)[5]

2.2 The NAO Robot

The NAO Robot is the first humanoid robot created by SoftBank Robotics. Proposed in 2006 and first sold in 2007, the NAO has become "[. . .] a standard in education and research" around the world [6, SoftBank Robotics]. When Bruno Maisonnier first proposed the NAO, he had the goal of creating an affordable robot, without sacrificing quality and performance [7, Gouaillier et al., 2009]. Whereas most other humanoid robots at the time came at a price of at least $50,000 like Fujitsu's HOAP, he envisioned the NAO to not cost more than $10,000 for academics and even as little as $4,000 for public use.
Another goal was performance; a lightweight robot, in contrast to heavy robots, needs smaller and less powerful motors, has to dissipate less thermal heat, has better dynamic capabilities and offers a larger acceleration range. Furthermore, thanks to their little weight, robots like the NAO are far less dangerous and subject to breakdown than large, heavy robots [7, Gouaillier et al., 2009].
Today the main uses of the NAO are as assistants in companies, as well as in healthcare centers to interact and inform visitors. Other use-cases include education and academics, where the NAO has become a major point of interest in the scientific community.

Figure 2.2: The NAO robot[6, SoftBank Robotics]

The NAO Robot comes in at 58 cm in height, and offers a variety of sensors [6, SoftBank Robotics]:

- Seven touch sensors on the head, hands and feet. These work along with sonars for the robot to locate itself in an environment and perceive its surroundings.
- Four directional microphones along with support for voice-commands and speech recognition in 20 languages, as well as speakers for playing sounds and interacting with its human superiors.
- Two 2-Dimensional cameras, one located in its "mouth" and the other on its forehead, which can be used to identify objects and people.

Furthermore, the NAO uses an Open-Source programmable platform, allowing users to run custom code on the robot, like a software for playing football.
The NAO is currently on its 6th generation, which was released in 2018 and comes with a superior CPU compared to its predecessors to enhance performance. The first NAO robot was launched in 2007.

2.2.1 Joints

A major concern in the topic proposed in this thesis are the NAO's joints. Since these will later be manipulated (first statically then dynamically to make the robot's movements seem more realistic), it is important to stay within the joints' angles' limits in order to not break the robot (as the joints' gears are made out of fragile plastic).
The NAO offers a total of 25 degrees of freedom (or 25 "joints"), 11 of which are in the legs and pelvis, 12 of which are in the chest and arms and two of which are in the robot's neck [7, Gouaillier et al., 2009]. The illustration below 2.3 gives a graphical representation of the locations and rotational directions of the joints.
For the following sections a motion denoted with "Roll" performs a rotation about the X axis (marked in red), a motion denoted with "Pitch" performs a rotation around the Y axis (marked in green) and a motion denoted with "Yaw" describes a rotation around the Z axis (marked in blue).
The joints' names and movements are as follows:

- The head's forward / backward motion is denoted by "Head Pitch", and its sideways movement by "Head Yaw".
- The shoulders upward / downward motions are denoted by "Shoulder Pitch", the upper arms' sideways motion by "Shoulder Roll", the elbows' sideways motion by "Elbow Roll", the elbows' rotations by "Elbow Yaw" and the wrists' rotation by "Wrist Yaw". *The letters "L" and "R" in the names are used to denote the arm that is being addressed, where "R" stands for "Right Arm" and "L" obviously stands for "Left Arm".*
- Similarly, the legs' upward / downward motion is denoted through "Hip Pitch", the knees' upward / downward motion by "Knee Pitch", the ankles' upward / downward motion by "Ankle Pitch", the hips' rotations by "Hip Roll", and the ankles' rotations by "Ankle Roll". *Again, the letters "L" and "R" are used to denote which leg is being targeted.*
- Finally, the pelvis is also able to turn both hips simultaneously inwards and outwards, and is denoted by "Hip Yaw Pitch".

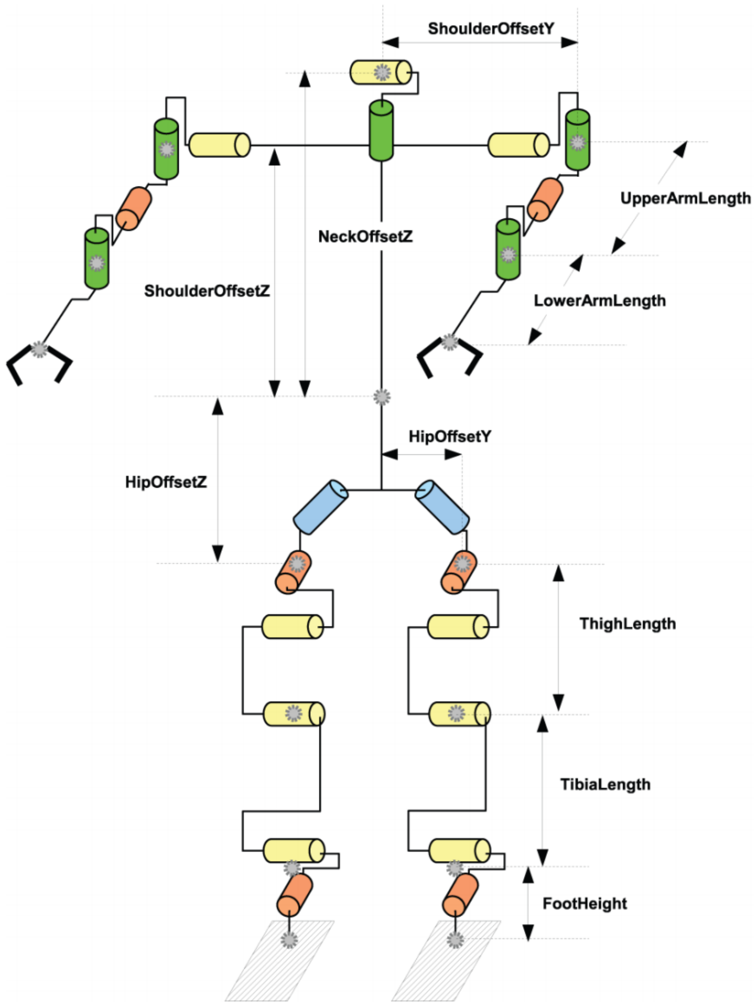

Figure 2.3: The NAO's joints [7, Gouaillier et al., 2009]

The images below give an overview of the different joints, their minimum and maximum angles, and their rotational axis. All images were taken from the Aldebaran documentation's section about the NAO's joints [8]. *Note that even though the company was renamed into SoftBank Robotics, the documentation still holds the old name, Aldebaran.* The same point applies to the documentation cited in the LEDs' section 2.2.2.

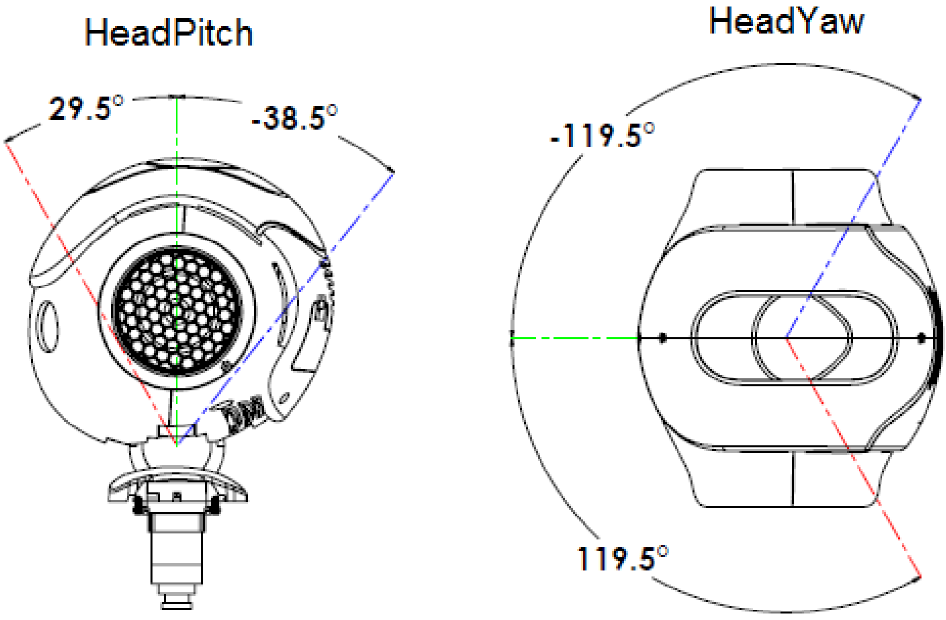

Figure 2.4: Head Joints

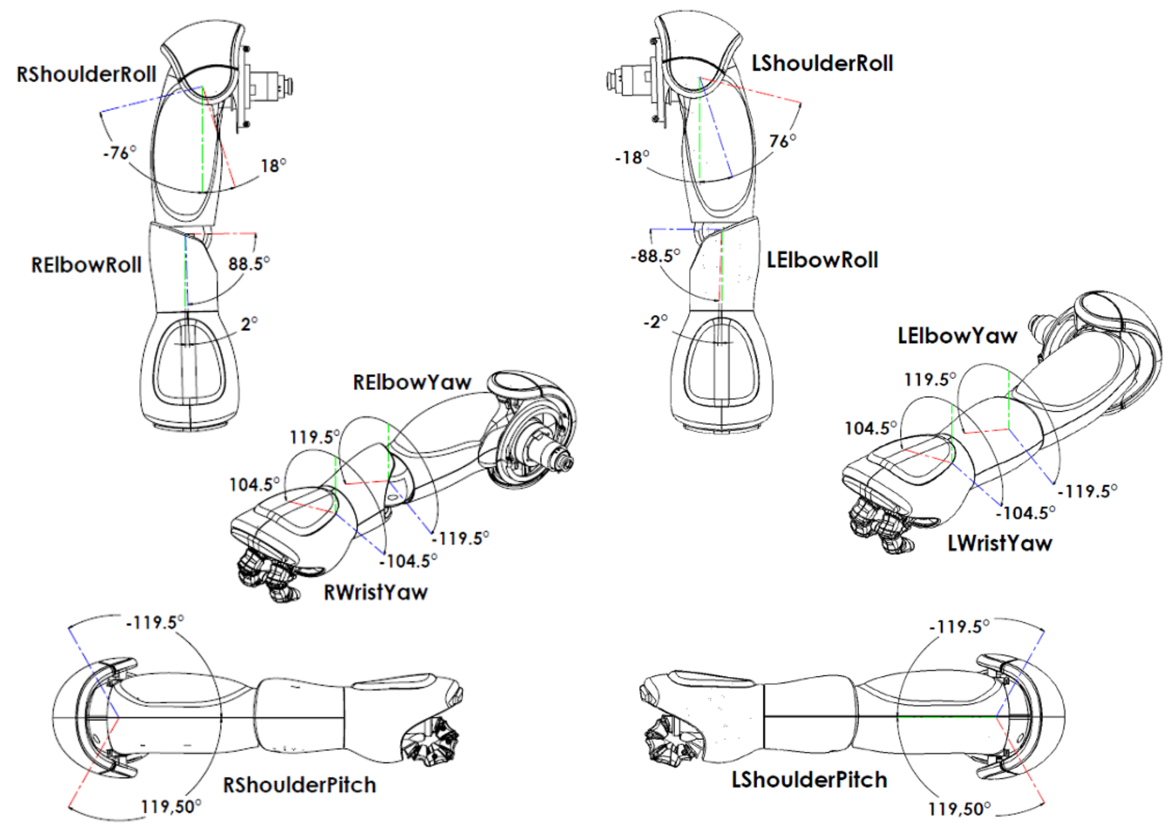

Figure 2.5: Arm Joints

The NAO has an additional joint in each of its fists which can be used to open and close the robot's hands. Since these are not used within a RoboCup game, where the hands are taped into fists, they are not listed above.

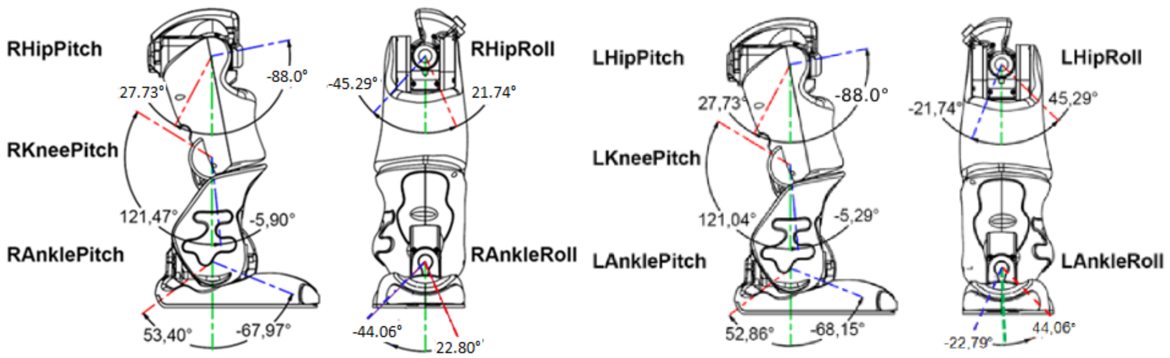

Figure 2.6: Leg Joints

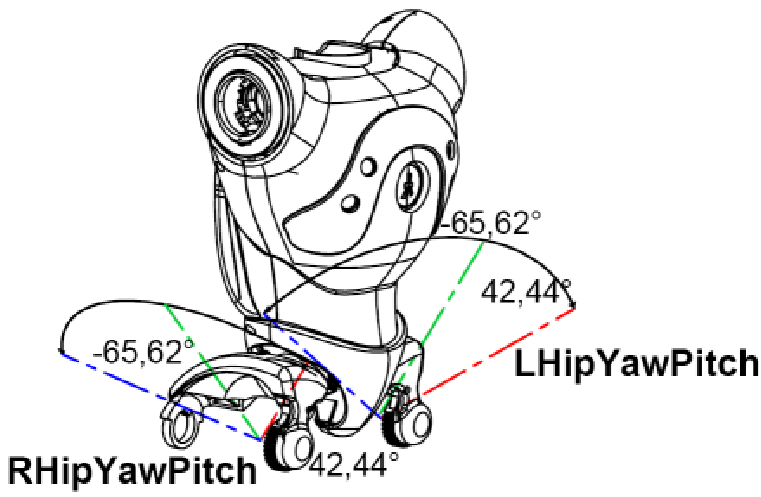

Figure 2.7: Pelvis Joints

The importance of listing the joint angles will become clearer later (See section 4.1).

2.2.2 LEDs

Besides the joints the NAO also offers a total of 50 LEDs all around its body, 19 of which have all three RGB (Red-Green-Blue) LEDs and are therefor able to display any desired color, while the other LEDs are either blue or white.
The LEDs will play a central role in making the robots' behaviours seem more authentic, as they can be used to express how the robot is feeling in given scenarios based on the colors or patterns displayed by them (See section 4.2).
All the LEDs are individually addressable. For the RGB LEDs each color diode of the LED is also individually addressable, meaning that the LEDs themselves cannot understand a color like "magenta". Rather the colors "blue" and "red" of the RGB LED need to be lit manually. This means that for every RGB LED each of the three LED diodes (Red, green and blue) needs to be addressed individually. Each LED diode can be set to a state of either "ON", "HALF" or "OFF", where "ON" lights up the LED in its maximum brightness and "HALF" lights the LED in medium brightness.
The locations of the LEDs are as follows (the images are again taken from the Aldebaran documentation [9]):

- 12 LEDs reside on the robot's head (also referred to as "brain" by SoftBank). These can only glow in the color white and are also used by the pressure sensors on the NAO's head, where touching the sensors turns the LEDs on. Even though they have a diffusion ring for spreading the light, there still exist visible gaps between the LEDs.

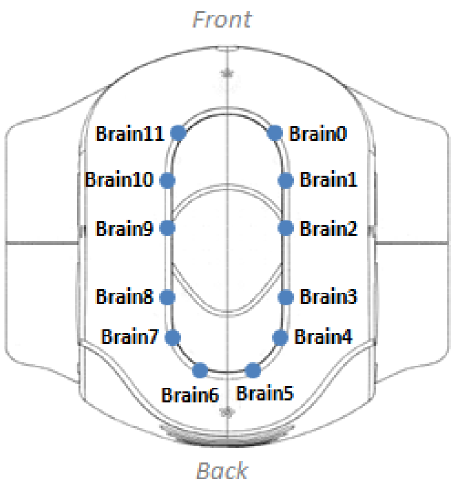

Figure 2.8: The NAO's "brain" LEDs

- 10 blue LEDs are installed inside each of the robot's ears. They are installed in a ring around the speaker grills (where the speakers are in place of ears) and use a diffusion ring with much more light bleed than the head (with absolutely no gaps between the LEDs), which can make ear effects less visible (as will become apparent later 4.2).

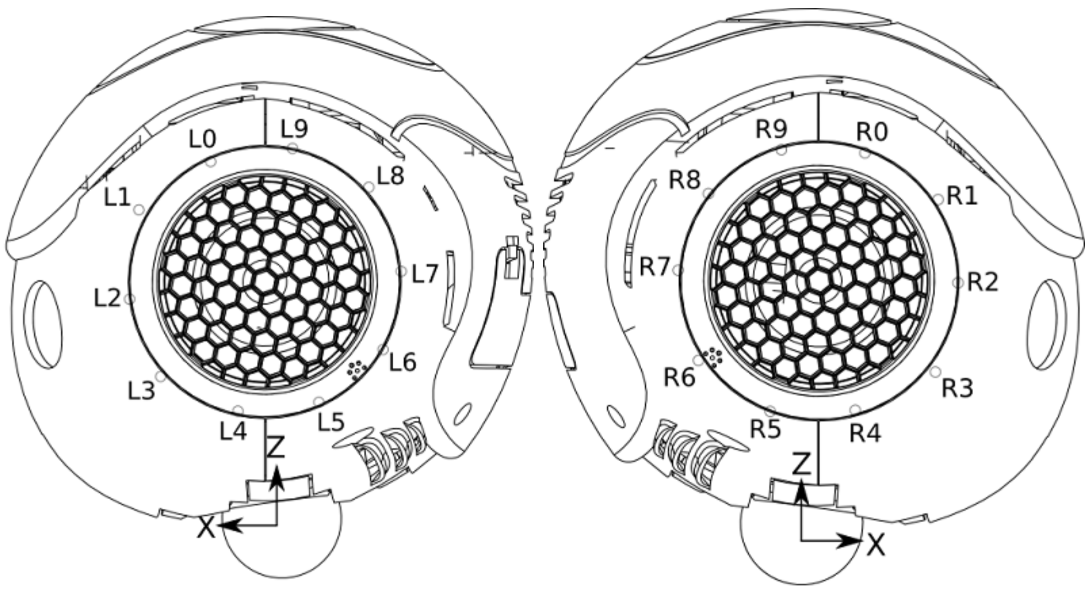

Figure 2.9: The NAO's ear LEDs

- The chest button offers a single RGB LED (3 diodes). *No picture has been added at this point as it is missing in the documentation.*
- Each of the eyes features eight RGB LEDs (24 diodes, 3 per RGB LED) with a diffusion ring which is slightly curved towards the center of the robot's eyes.

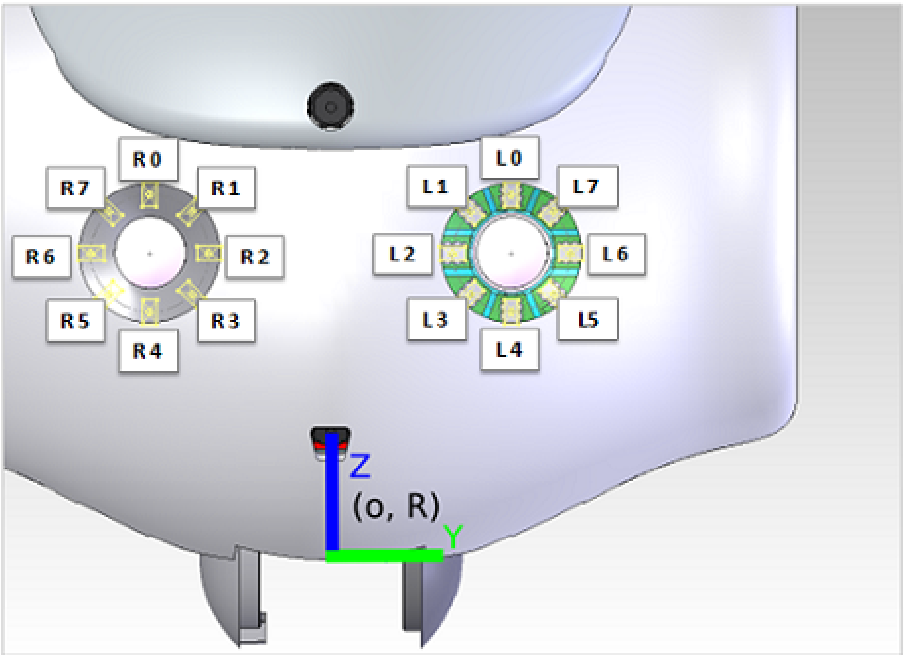

Figure 2.10: The NAO's eye LEDs

- The feet each offer one RGB LED (3 diodes). These are by far the dimmest LEDs that the robot possesses, and are therefor the least visible for spectators, even if these are no more than a couple meters away.

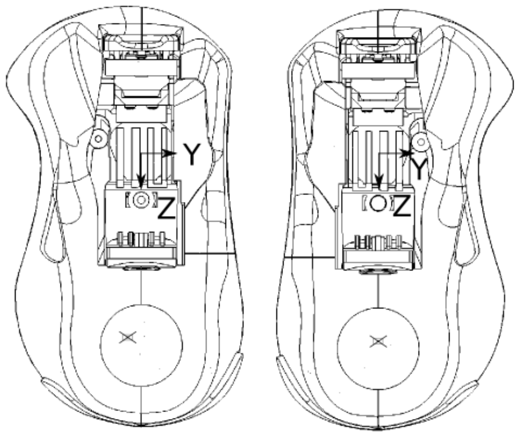

Figure 2.11: The NAO's feet LEDs (marked by "x")

The most noticeable LEDs are the eyes, as they are the second brightest (beat only by the ear LEDs, since they are much larger) while also being RGB and at a physical location that humans are used to. Therefor these will play a very important role in creating emotional reactions as will become apparent later 4.3.

2.3 RoboCup

The Robot World Cup Initiative was founded by a group of Japanese researchers [10, RoboCup] who in 1992 gathered for a Workshop on Grand Challenges in Artificial Intelligence (AI) in Tokyo, Japan, during which a discussion was held about using the game of soccer as a standardized problem in an attempt to promote scientific research in the fields of AI and robotics research [11, Kitano et al., 1998].
After assessing the feasibility of the idea and developing special game rules, soccer robot prototypes and simulator systems, in 1993, Minoru Asada, Yasuo Kuniyoshi, and Hiroaki Kitano along with a team of researchers launched a robot competition initially named the "Robot J-League" where the J-League was the name of Japan's newly established professional soccer league [10, RoboCup].
As a result the researchers were overwhelmed with reactions from all over the world requesting to extend the initiative on a global scale. Consequently, the initiative was renamed to the "Robot World Cup Initiative", or "RoboCup" for short. The first official RoboCup games and conference were hosted shortly afterwards, in 1997.
RoboCup's ultimate goal is beating the most recent winner team of the FIFA World Cup in 2050 using a team of fully autonomous robots while complying with the FIFA rules of soccer [12, Ferrein et al., 2016].
RoboCup hosts yearly events and offers a variety of robot challenges for different leagues. Today, these leagues are [12, Ferrein et al., 2016]:

- The Soccer Simulation League, a virtual soccer league with a focus on multiagent systems and high-level decision-making.

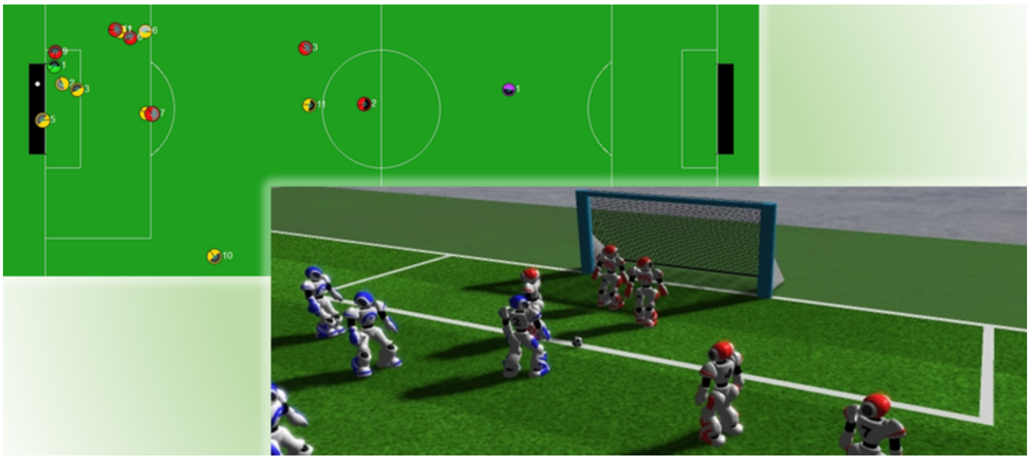

Figure 2.12: RoboCup Soccer Simulation League [13, RoboCup]

- The Small-Size (SSL) League, which uses semi-autonomous robots with a diameter of 18 cm and a height of up to 15 cm. Robot tracking is done using overhead vision systems and processing is done on off-side computers, which relay information to the robots. It is one of the first RoboCup leagues along with the Middle-Size League and the Humanoid League which are still hosted today.

- The Middle-Size League (MSL) uses slightly larger robots on a 12 x 18 m field today, which is quite a step-up from the 3 x 4 m field originally used. The robots evolved from slow driving to extremely fast robots capable of achieving speeds up to 4 m/s. Unlike the SSL robots, the MSL robots are fully autonomous, where all the sensing and processing is done on-board rather than on off-side computers.

Figure 2.13: RoboCup Small-Size League [14, RoboCup]

Figure 2.14: RoboCup Middle-Size League [15, RoboCup]

- The Standard Platform League (SPL), explained in the following section 2.4.
- The Humanoid League, where humanoid robots play soccer on a 9 x 6 m field. There are three different sizes of robots in this league: KidSize, TeenSize and AdultSize. This league is mainly working towards RoboCup's ultimate goal of beating the FIFA World Cup champion by 2050.
- The Rescue Robot League (RRL). RoboCup has evolved beyond only having soccer playing robots onto offering leagues for robots with real-life applications. The Rescue Robot League consists of robots developed to help first responders in mitigating disaster or accidents in industrial environments, with the goal being to keep humans out of harm's way while still providing decent reconnaissance and manipulation skills where needed.
- The Rescue Virtual Robot League, where high-fidelity simulations of disaster sites are used. Such simulations can include collapsed and burning buildings, floods or similar disaster sites. This league allows realistically simulating large-scale scenarios where a multitude of robots, both homogenous and heterogeneous can be deployed. Although no real hardware is involved, the tasks the virtual robots need to execute are still very similar to those of the RRL robots.

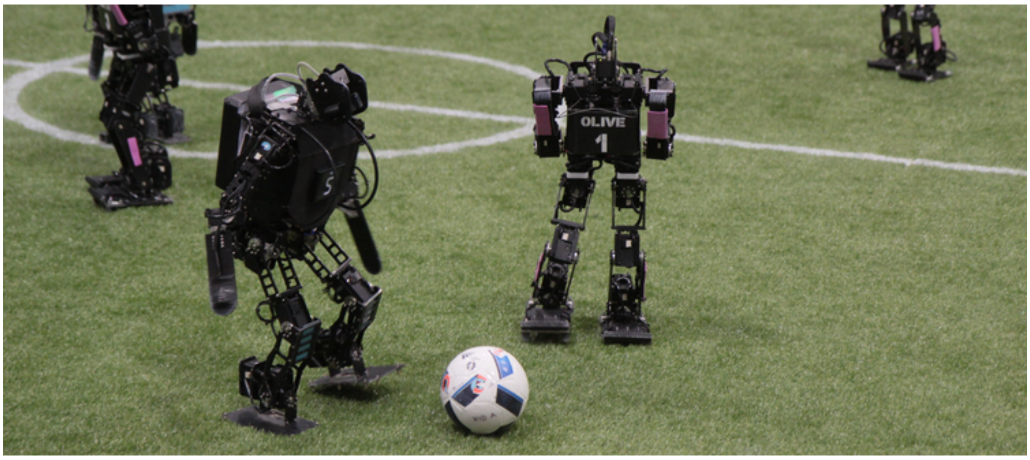

Figure 2.15: RoboCup Humanoid League with KidSize robots [16, RoboCup]

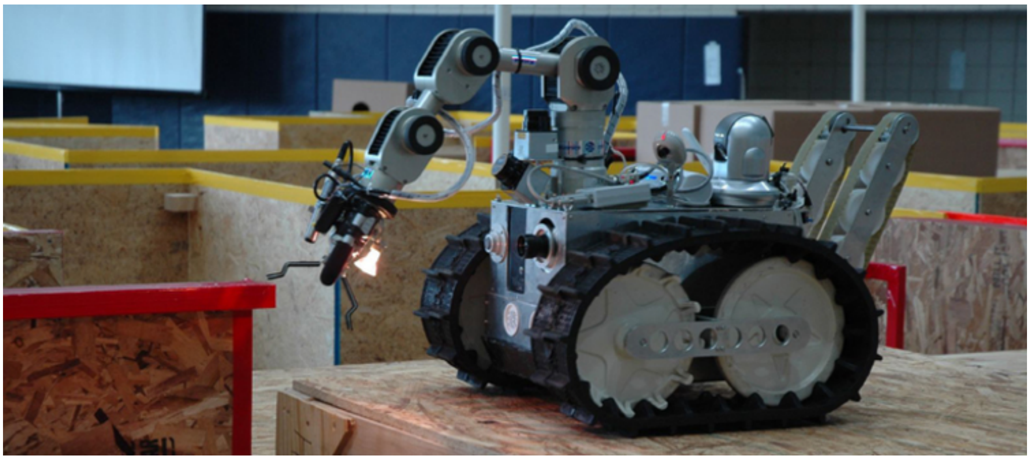

Figure 2.16: RoboCup Rescue Robot League [17, RoboCup]

Figure 2.17: RoboCup Rescue Virtual League [18, RoboCup]

- The RoboCup@Home League (created in Bremen) aims at developing domestic service robots, which can be deployed in day-to-day scenarios. Tasks these robots execute can include doing chores such as helping with groceries, finding lost objects or cooking.

Figure 2.18: RoboCup@Home League [19, RoboCup]

- The Industrial Leagues where robots are deployed for helping in manufacturing scenarios of all kinds, such as supplying machines with raw materials or delivering final products to fulfill production plans.

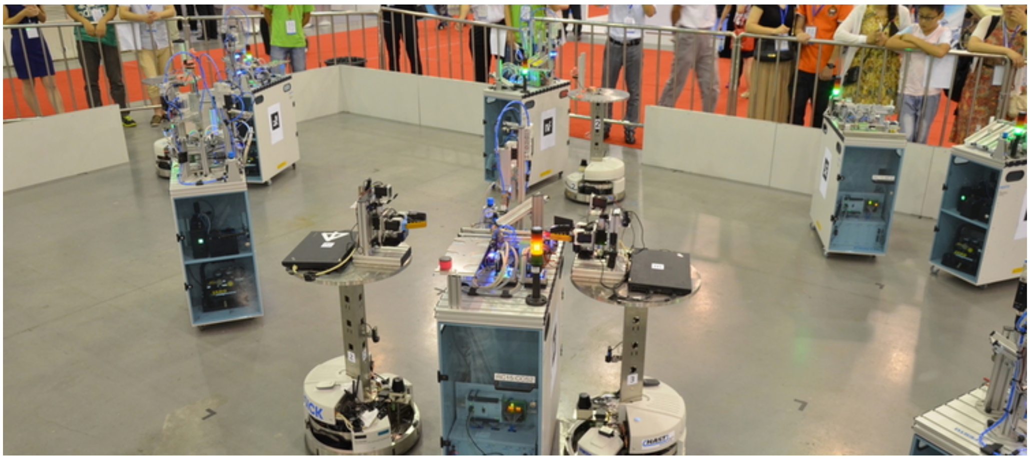

Figure 2.19: RoboCup Industrial League (Logistics)[20, RoboCup]

- The RoboCupJunior League, which includes the RoboCupJunior Soccer competition (Soccer), the RoboCupJunior Rescue competition (Rescue) and the RoboCupJunior Dance competition (OnStage). This league is intended as an educational tool for young people to get into programming and robotics.

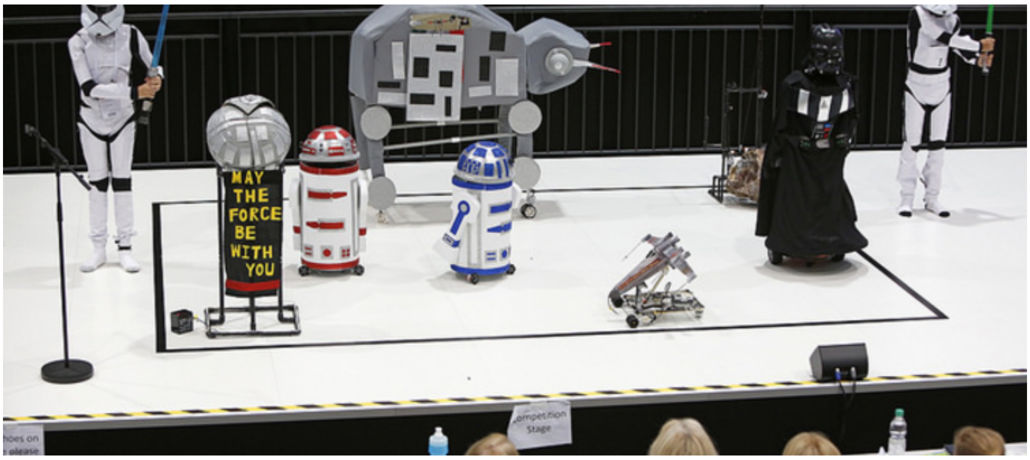

Figure 2.20: RoboCupJunior League OnStage dance competition [21, RoboCup]

The NAO robots participate in the so-called Standard Platform League (SPL), explained below.

2.4 The Standard Platform League (SPL)

RoboCup's Standard Platform League (SPL) is the successor to the Sony Four-Legged League [12, Ferrein et al., 2016]. In the Four-Legged League, teams used to participate in games of soccer using Sony's AIBO robot (a small, four-legged, dog-like robot. See the picture below 2.21). Since Sony stopped production of the original AIBO robot (production has been revived with a new version of the AIBO robot in 2018 [22, S. Knight, 2017]), the Four-Legged League was replaced by the Standard Platform League, in which the humanoid NAO robot is used instead. In the Standard Platform League all teams use the *same unmodified hardware* (similar to the original Four-Legged League) rendering the competition a pure software competition, in which the main focus is on software design and creating the best possible control algorithms for the NAO [12, Ferrein et al., 2016].

Figure 2.21: Sony's AIBO Robot [22, S. Knight, 2017]

The robots need to operate fully autonomously with all the computations and image processing

being done with the on-board sensors and CPU. The robots are not allowed to communicate with external computers (except for one referee computer), they are however allowed to have inter-robot communication in order to operate as a team.
The robots play on a 10.4m x 7.4m artificial grass soccer field with 8 mm thick turf mounted on a flat wooden base [23, SPL Rulebook]. There are set dimensions and colors for the goals as well as the paintings on the turf, these have however changed throughout the years (today the goals are white, compared to the original SPL games in which the goals were pink. The lighting conditions and field size have also changed; In 2016 a first SPL game was attempted outdoors with real grass and extremely variable lighting conditions [12, Ferrein et al., 2016]). A game is played in constellations of 5 versus 5 robots (plus 2 coaching robots, one for each team), two of which are goal-keepers, while the rest are "field players".

Figure 2.22: SPL Playing Field, laid out at the MZH Uni Bremen [24, L. Plecher, 2021]

2.5 B-Human

B-Human is "[. . .] a collegiate project at the Department of Computer Science of the University of Bremen and the DFKI research department Cyber-Physical Systems"[25, B-Human] with the goal of developing software suitable for participating in several RoboCup events and for motivating students for an academic career in artificial intelligence.
The B-Human team consists of researchers from the University of Bremen and the German Research Center for Artificial Intelligence (DFKI), as well as students from the University of Bremen [25, B-Human].

Figure 2.23: A B-Human Robot, [26, A. Popp, 2019]

B-Human has been a major competitor in the Standard Platform League since 2009. Since then B-Human has managed to win the RoboCup World Championship 8 times, coming in second twice and third only once, never dropping below third place. B-Human has also won RoboCup's German Open almost every year consecutively since 2009, for a total of 11 German Open wins, coming in third only once, again never dropping below third place.
B-Human develops and maintains the B-Human framework, which is the software that allows the robots to play soccer. It is a key factor in this thesis, as all implemented features are extensions of B-Humans codebase. It offers (almost) yearly code releases, allowing other RoboCup teams to download, modify and use the B-Human software during their own soccer matches.

Chapter 3

Movements and Emotions

This chapter describes the process by which the robots' emotional reactions were picked.

3.1 Inspiration

Note: This thesis will not discuss the automatic triggering of different emotional reactions during the various game-scenarios present during an SPL match. It will however, mention the various scenarios as "emotional injection points" which were used to determine what reactions should be introduced. The game-scenarios are discussed in the previously mentioned thesis by Mr. Phillip Dolata 1.2.

Since the ultimate goal is giving the robots ways to express emotion one might start by giving them the ability of using one of the oldest existing ways of expressing emotions: dancing. Other ways of expressing emotion could include hand gestures, sounds and targeted use of the robot's LEDs, to compensate for facial expressions which the robots are unable to do.

Since the robots are quite dull when it comes to scoring goals, this "game scenario" (scoring a goal) could therefor be a perfect fit for adding some positive emotional reactions. The same could be said for when the opposing team scores a goal, where the robots could instead show sadness or anger. Other scenarios could include:

- Game-Over: reactions would be based on whether the game was won or lost.
- Missing scoring a goal: could be used for angry reactions.
- Causing a corner shot: anger or annoyance can be used.
- Shooting the ball out of the playing field (causing an "out"): the robot(s) can show annoyance.

With the scenarios and their corresponding emotional stages split into different categories, the question that remains is: what should these reactions look like?

3.2 Initial User Survey

To settle that question an initial user survey was conducted. "Initial" since there will be a follow-up survey at the end of the project to determine how realistic the robots' behaviours look to a distant observer.

3.2.1 Preparation

Questions were designed to be easily comprehensible by anyone unfamiliar with IT. They asked participants to "imagine" the robots in pre-specified game-scenarios and pick what reactions they found to be fitting.

The survey was split into three sections:

1. The first section was used to determine how a **single robot** should react in specified game-scenarios.
2. The second section was used to determine how a **robot's teammates** should react in specified game-scenarios (if for instance the robot scored a goal).
3. The third section was used to collect missing game-scenarios, which are unused in *this thesis*. They were instead used in the thesis by Mr. Phillip Dolata, as previously mentioned 3.1.
4. The forth section allowed participants to add their own robot behaviours and reactions.
5. The fifth section collected feedback for improving the follow-up questionnaire.

Furthermore, each section allowed participants to add their own reactions to pre-defined game-scenarios (in addition to section 4) if they didn't like any of the existing ones. Participants were allowed to select multiple reactions for each game-scenario to add some dynamism in the robots' behaviour, since it would be dull if the robots kept reacting the same way every in every scenario.
At the beginning of the survey, participants who had never seen the robots or heard of RoboCup were presented with two videos: One showcasing the NAO robot in a review video by "Unbox Therapy" [27] and another showcasing a robot football game using an excerpt of the RoboCup 2019 SPL Finale between B-Human and the HWTK RoboCup Team of the HWTK University of Applied Sciences in Berlin [28].
The specific game scenarios used in the survey were the following:

1. **Single robot** behaviour - participants were asked to choose what a robot should do in specific game-scenarios, based on its role in the team. The scenarios were:
   - The robot scored a goal.
   - The robot missed scoring a goal.
   - The robot scored, and its team is now leading.
   - The robot is a goalie, the opposing team scored a goal.
   - The robot caused a penalty.
   - The robot is penalized and standing at the edge of the field.
   - The robot got knocked over by an opposing team member.
   - The robot caused an "out".
   - The robot caused a corner shot.
2. **Team** behaviour - participants needed to select what a robot's teammates should do in the following scenarios:
   - The own team has scored.
   - The opposing team has scored.
   - The own team won the game.
   - The own team lost the game.

Participants were provided with the following possible robot reactions. If they didn't like any of them, they had a text field to add their own suggestions.

Default: Do nothing.
Joyful reactions
The robot cheers without sound.
The robot cheers with sound.
The robot dabs (as in the popular "Dab" emote, see the following section "Reactions" 4.3).
The robot gets on its knees and throws its arms in the air, joyfully.
The robot dances.
The robot high-fives a teammate or the B-Human developers.
The robot salutes.
Sad reactions
The robot shakes its head.
The robot sadly gets on its knees.
The robot throws its arms at its face.
The robot covers its face shamefully.
The robot angrily shakes its fists.
The robot angrily tramples its feet.
The robot cusses/gives the finger.
The robot sadly looks at its feet.

3.2.2 Execution and Results

The survey was conducted using a Google Forms questionnaire [29] without age, career or technical proficiency restrictions. The link to the survey was sent through the University of Bremen's broadcast system and distributed in dozens of social media groups. It was live (accepting results) for a period of 4 weeks. In total, it received 38 responses with an average participant age of 25, with the oldest participant being 57 years old and the youngest participant being 17 years of age.

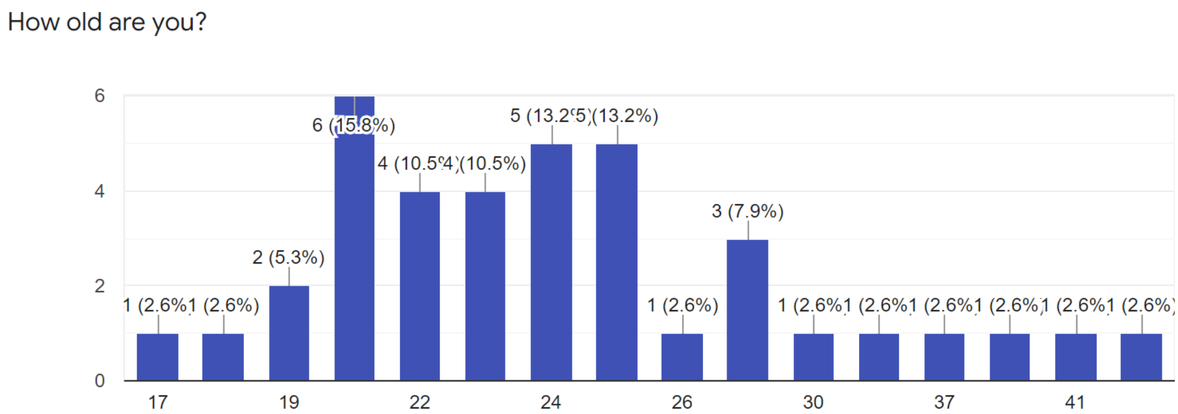

Figure 3.1: Demographics of the initial user survey.

Participants were also asked whether they were students and what fields they were studying. This was done to separate between results from participants active in IT fields from results from other participants. The majority of participants were indeed active in information technology (IT) fields, some however, were active in other fields such as psychology and engineering, which is crucial since the reactions are mainly targeted at the average non-IT spectator. The charts below illustrate the different fields the survey's participants are active in. *Note that a lot of the fields are actually "Computer Science", they are simply split into different columns since Google Forms distinguishes results if they are typed even slightly differently (or in some cases in a different language, as seen below).*

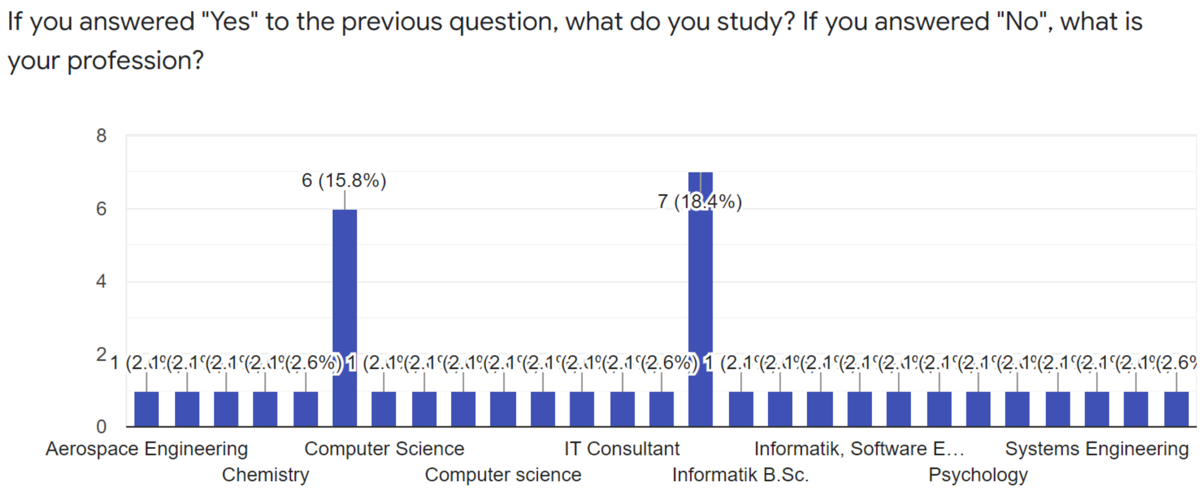

Figure 3.2: Fields that participants of the initial user survey are active in.

One-third of participants were part of the B-Human development team, who have insights into the limitations present in the robots, which improved the evaluation process dramatically and allowed eliminating illegal game-scenarios. Additionally, unusable robot reactions due to joint limitations [1] were also flagged, prompting these to be adjusted accordingly or eliminated entirely.

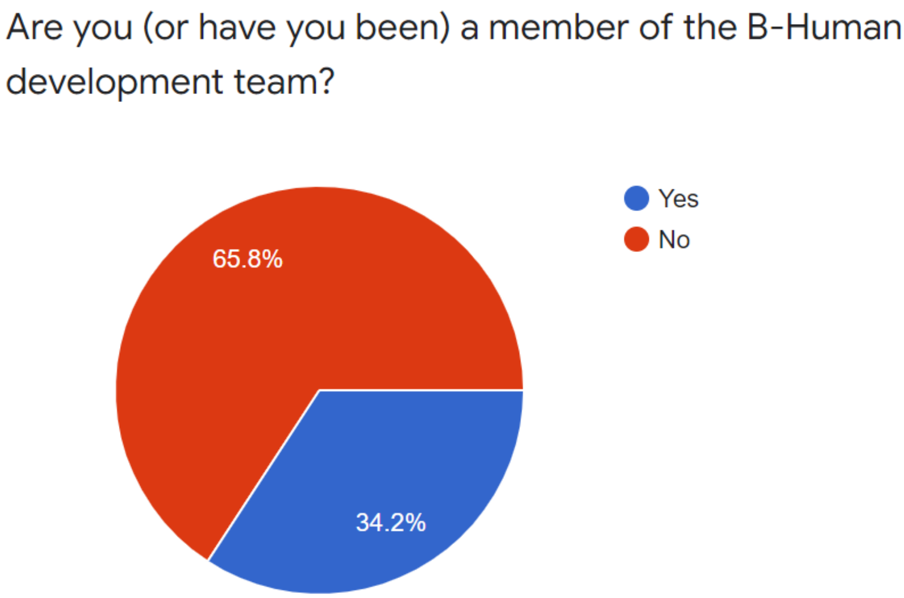

Figure 3.3: One third of participants were part of the B-Human team.

Before participants were asked about how the robots should behave and in what scenarios they should do so, they were asked general questions like "Have you ever seen a human soccer game?", to which, unsurprisingly, 92.1% of participants replied with "Yes" and perhaps most importantly:

[1] See Chapter 2.2.1

"Have you ever watched a robot football game or seen the robots in action?" (excluding the video they were shown at the start of the survey). This was done to get an overview of the mental models present in the participants when it comes to robot behaviour, which are taken into account while evaluating the reactions participants picked in the following section of the survey. Participants were also asked whether they observed and/or paid attention to the behaviours of soccer players during human soccer games. When asked, 39.5% of participants answered with "Yes", 28.9% with "Kind of" and 31.6% said they did not pay attention to soccer players' emotions during football games. This question was both used to probe the participants' emotional awareness, and for determining how much attention they would pay to minor reactions caused by shifts in robots' emotional stages during soccer matches when these are implemented.

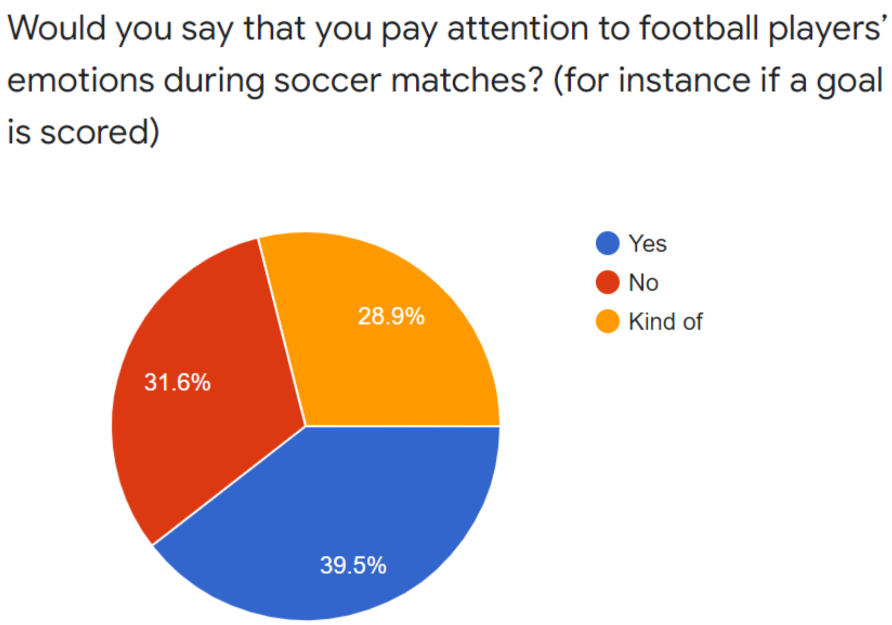

Figure 3.4: One third of participants said they do not pay attention to football players' emotions.

After gathering demographic data, participants were asked what emotional reactions they wanted both single robots and the robot team to perform, and in what game-scenarios these should be performed. A list of the results can be found below 3.3. Here we continue with the evaluation process by picking out the eight most liked reactions and implementing these (initially there were more, but due to lack of quality these were narrowed down to a total of eight reactions). Additionally, two team reactions have been added (high-fiving and fist-bumping), but since these are yet to be introduced during specific game-scenarios by Mr. Phillip Dolata, they are incomplete, and will therefor not be discussed here any further.
Thanks to the participants from the B-Human development team, the following game-scenarios were immediately eliminated because of restrictions in the SPL rules, and game interruptions which would be caused by reactions in these scenarios:

- The robot caused a penalty - eliminated since the robot is not allowed to do anything once it is penalized (SPL rule book, chapter 4 - "Forbidden Actions and Penalties" [23, SPL Rulebook]).
- The robot is penalized and standing at the edge of the field - eliminated for the same reason as the previous scenario.
- The robot got knocked over by an opposing team member - eliminated since it would hinder the robot from going back to playing quickly, losing precious time and causing the

robot to get penalized (If a robot doesn't get up quickly enough and go back to playing it is automatically penalized).

Further scenarios were eliminated due to other reasons, these will not be mentioned here, or in this thesis, at all. They are instead mentioned in Mr. Phillip Dolata's thesis 1.2. The previous scenarios were mentioned since they are directly related to the survey's results.
Furthermore, reactions including any sort of feet movement were eliminated since they would destabilize the robot, causing it to trip or even fall. Some reactions were replaced by other more expressive emotions. These are mentioned below with a *"replaced by"* note.
The entire list of reactions can be found in chapter 4.3.
Overall, the following reactions were the most popular amongst participants (the percentages are calculated from all the survey's answers. They do not sum up to 100% as there were also other less popular reactions which are excluded here):

Joyful reactions (sorted by popularity)	
Reaction	**Votes (%)**
The robot dances joyfully	21%
The robot cheers with sound	17%
The robot dabs	14%
The robot salutes (replaced by "Stand for National Anthem")	12%
Sad reactions (Sorted by popularity)	
Reaction	**Votes (%)**
The robot shakes its head	24%
The robot tramples its feet angrily (eliminated)	20%
The robot throws its arm at its face	15%
The robot sadly looks at its feet (replaced by "Facepalm")	14%

With the most favourite reactions filtered out, the implementation(s) can begin.

3.3 Questionnaire: Full Results

This section showcases the full results of the conducted survey (besides the abovementioned demographics 3.1).
Note: in the following sections, the questionnaire uses the phrase "What reaction(s) do you expect" as a synonym for "what would you like the robot to do". As was pointed out by a lot of participants in the feedback section, this was a mistake. How much it affected the results is unknown, but it doesn't seem like it had much of an impact, considering most questions were still answered with emotional reactions instead of the default answer "The robot does nothing".
Participants were first asked how they wanted **a single robot** to behave in certain game scenarios.

- First off, participants were asked how they wanted a robot to perform when it scored a goal, to which the majority answered that the robot should cheer or dance, as seen below 3.5.

- Next they were asked how they wanted a robot to react after it had shot the ball, but missed the goal. The majority of participants found it appropriate for the robot to shake its head, or to cover its face.

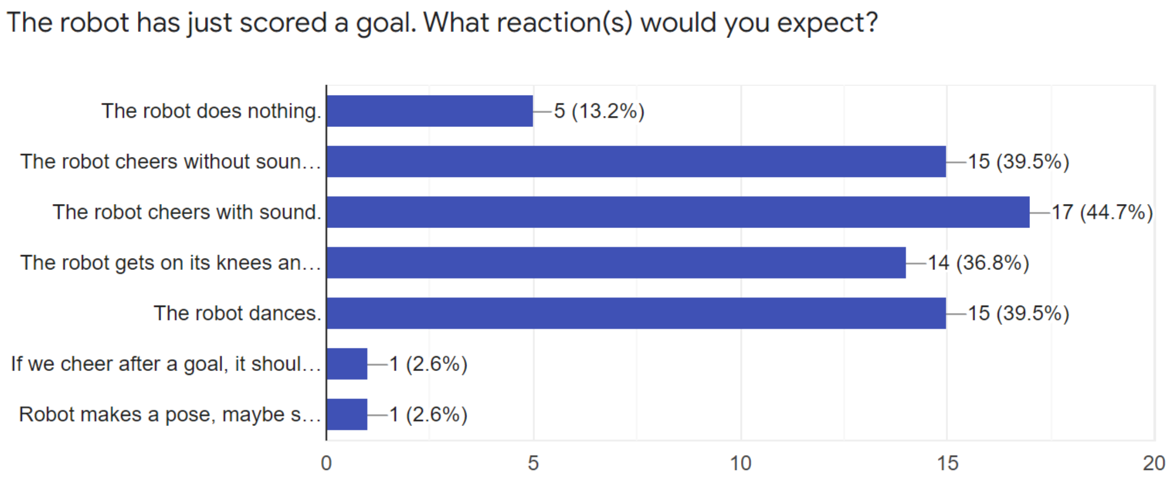

Figure 3.5: Most of the participants wanted the robot to cheer when scoring a goal.

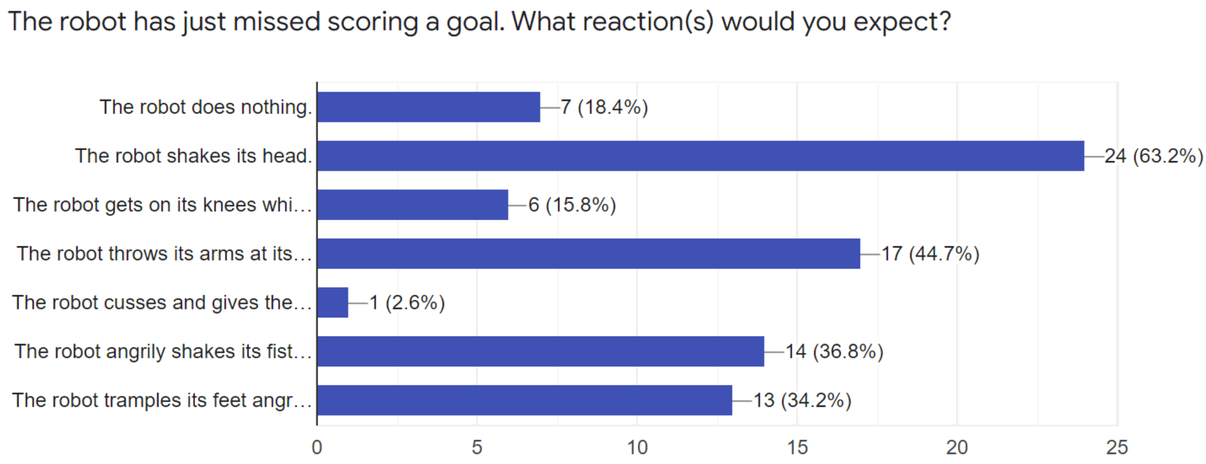

Figure 3.6: Most of the participants wanted the robot to shake its head when missing a goal.

- As a follow-up to the initial goal scoring question, a slight change in the scenario was made. Now when the robot scored a goal, its team became the leading team. This does not necessarily have to prompt a different reaction, but was thought of as a nice opportunity for introducing additional emotions, different from the ones present when simply scoring a random goal. In this case, most participants agreed that the robot should dance joyfully, but cheering still came in as quite a desired reaction. Surprisingly, this scenario gives light to a new reaction: dabbing (see 4.14 for what a "Dab" is), which was also highly popular.

- The next game reaction puts a spotlight on the robot: the opposing team just scored a goal, and the robot is the goalie who allowed it to happen. This seemed like a proper scenario, since in real soccer the goalie gets quite mad when the opposing team scores a goal. Just like in real soccer, the robots should share the same emotions, especially since the goal is making them seem more human. To this reaction, participants found it most appropriate for the robot to shake its head. They furthermore found it fitting for it to trample its feet, or to cover its face with its hands, as seen below 3.8.

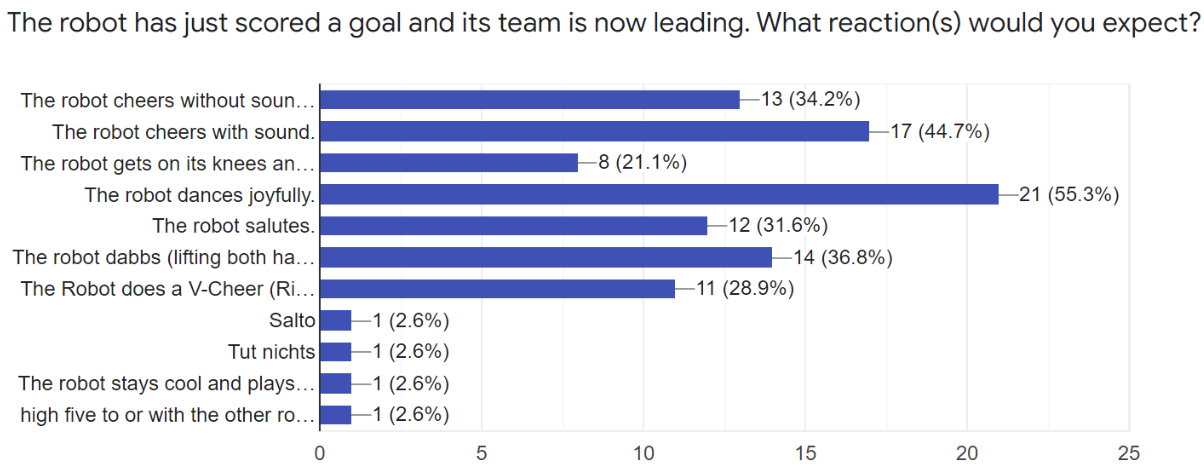

Figure 3.7: Most of the participants thought the robot should dance when it scores a leading goal.

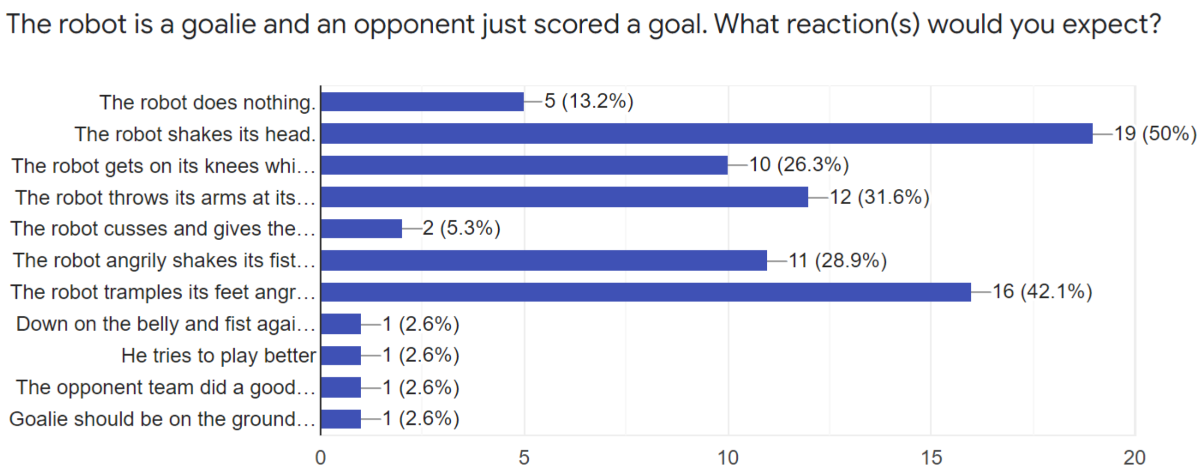

Figure 3.8: Participants thought that the goalie robot should shake its head if a goal is scored by the opposing team.

Next the participants were asked what they thought the robot should do when it caused a penalty or is standing penalized at the edge of the field. As mentioned previously in 3.2.2, thanks to the survey's B-Human participants, these scenarios were classified as illegal during RoboCup games. Nonetheless, here is what participants thought should happen during these two scenarios:

1. The robot caused a penalty: when causing a penalty, participants thought it would be fitting if the robot covered its face or shook its head (the answer "The robot does nothing" is ignored, since it is answered by the B-Human team members). Participants additionally thought proper reactions could include the robot trampling its feet, or angrily shaking its fists at the opponent. 6 Participants thought it would be cool if the robot cussed or gave the finger.

2. The robot is penalized and standing at the edge of the playing field: Participants thought it would be cool, if the robot would sadly look down at its toes while standing on the side of the field. They also thought it would be proper, if the robot cheered for its own team to lift the team spirits. 11 Participants found that the robot could feel offended and/or sad and play crying sounds.

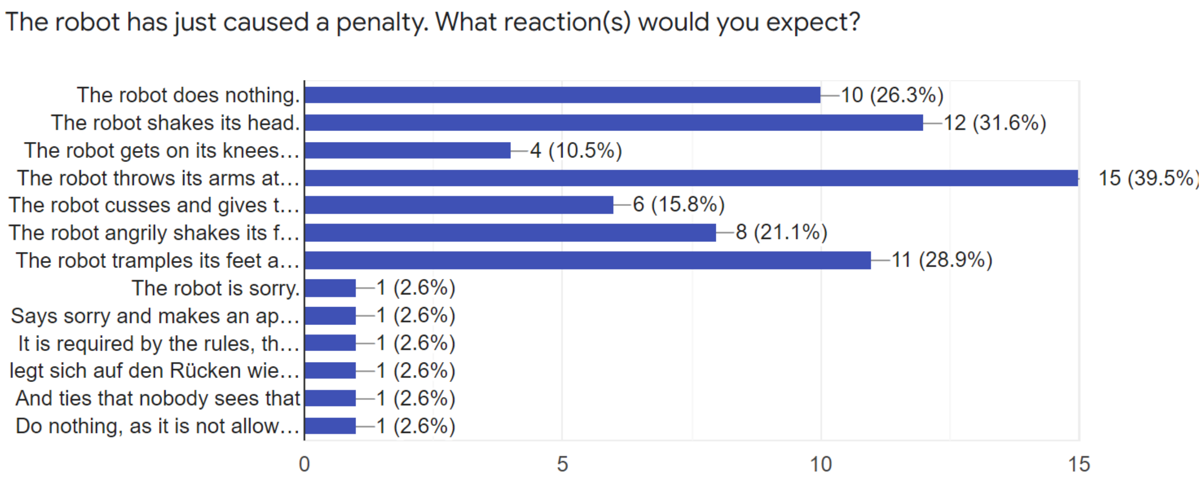

Figure 3.9: When the robot causes a penalty, participants thought it should cover its face.

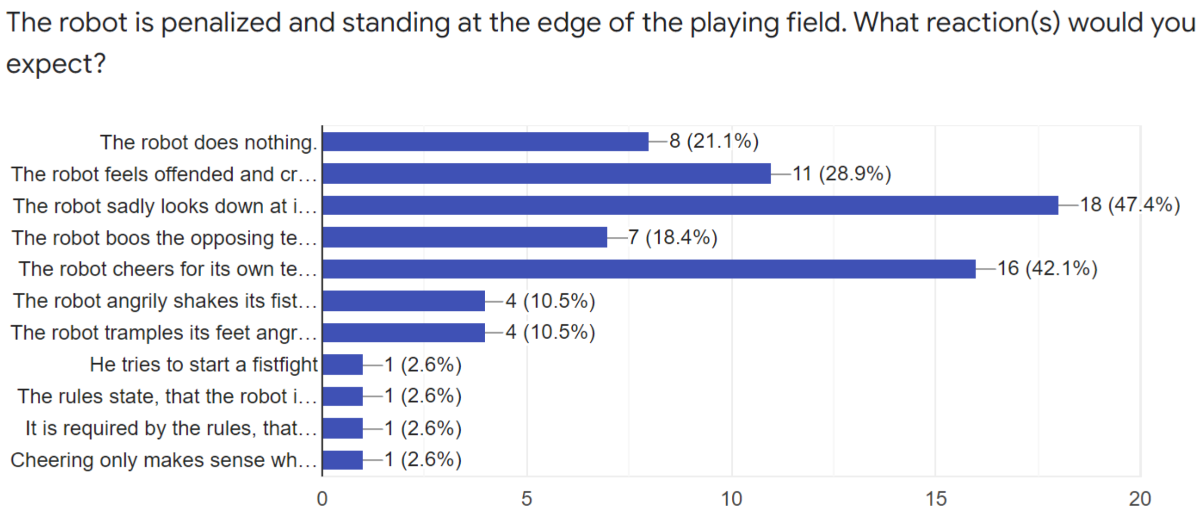

Figure 3.10: While the robot is penalized and standing at the edge of the playing field, participants thought it should sadly look at its toes.

- Yet another scenario was the robot getting knocked over by an opposing teammate. Again, just like in real soccer, where football players get extremely mad at one another when fouled, the robots could act angrily when an opposing teammate knocks them over (which happens quite a lot). This scenario, like the previous two, was however eliminated since there is a given time in which the robot needs to get back up on its knees and play, otherwise it would get penalized as per the rules [23, SPL Rulebook]. Regardless, participants thought the robot should angrily shake its firsts at its opponent if they knocked them over. Eight Participants would have liked the robot to cuss and/or give the finger.

- Furthermore, participants had to decide how they wanted a robot to behave if it caused an "out". The majority of participants again chose for the robot to shake its head, while others wanted it to trample its feet or cover its face. The B-Human team members thought the robot should not do anything in that case, which was due to another RoboCup rule and which is the reason that the scenario was eliminated as well.

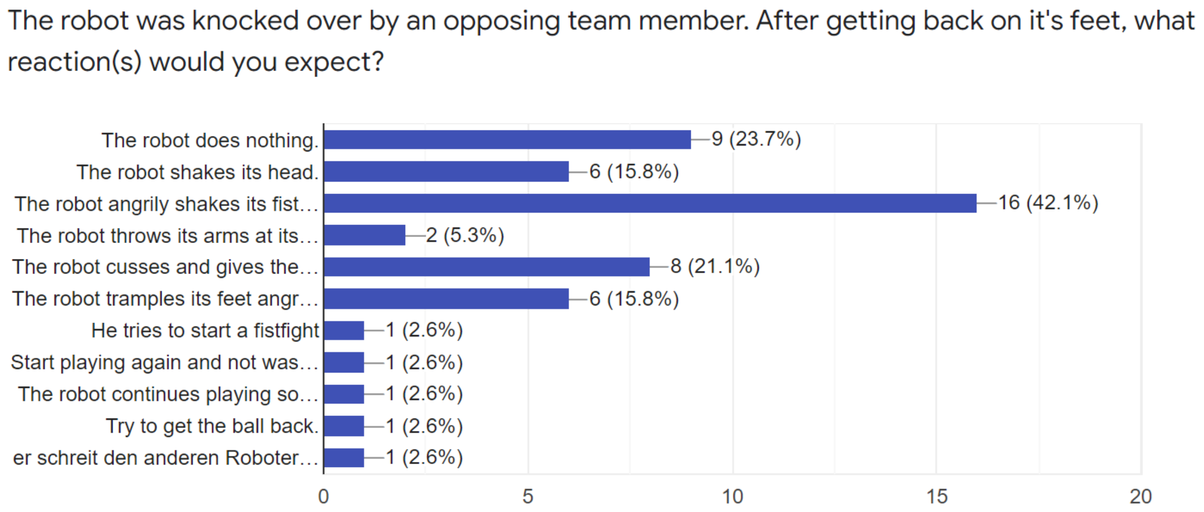

Figure 3.11: When the robot gets knocked over by an opponent, participants found it should shake its fists at the opponent angrily.

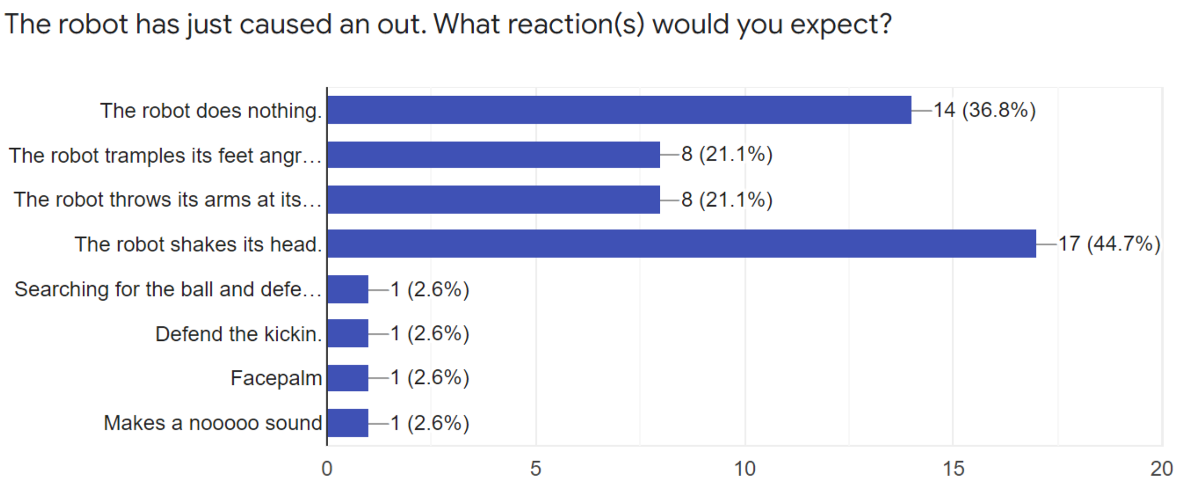

Figure 3.12: Participants thought it would be cool if the robot shook its head after causing an "out".

- Lastly, participants were asked how they wanted the robot to react when it caused a corner shot (which does exist in RoboCup!). Again, participants thought the robot should shake its head or cover its face.

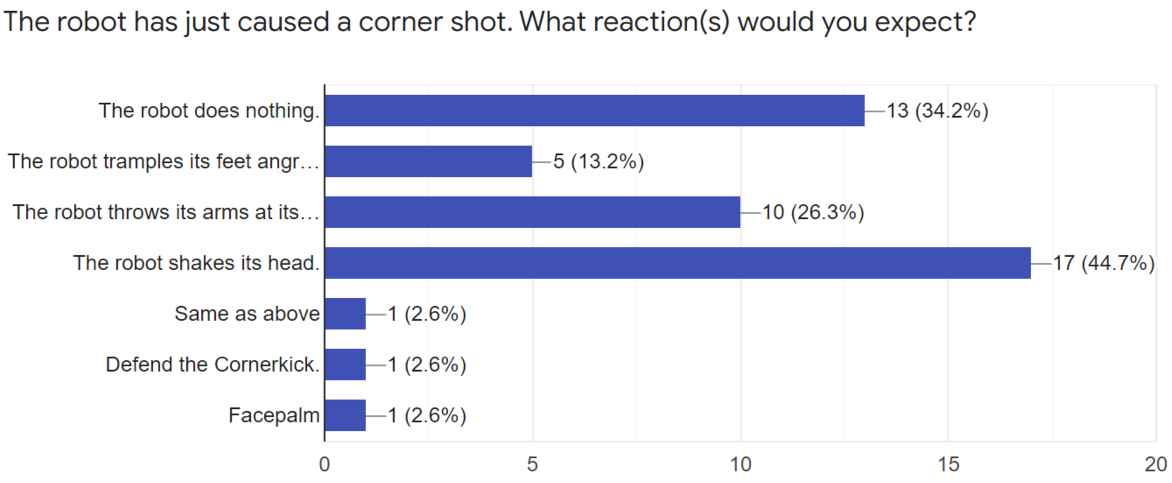

Figure 3.13: Participants thought the robot should shake its head after causing a corner shot.

Following the solo robot reactions was a chapter asking participants what they thought a robot team should do in the previously mentioned game-scenarios 3.2.1. The results were as follows:

- "A robot just scored a goal, what reactions should its teammates perform?": participants favored robots high-fiving the robot which scored a goal. Moreover, they thought the robots should cheer.

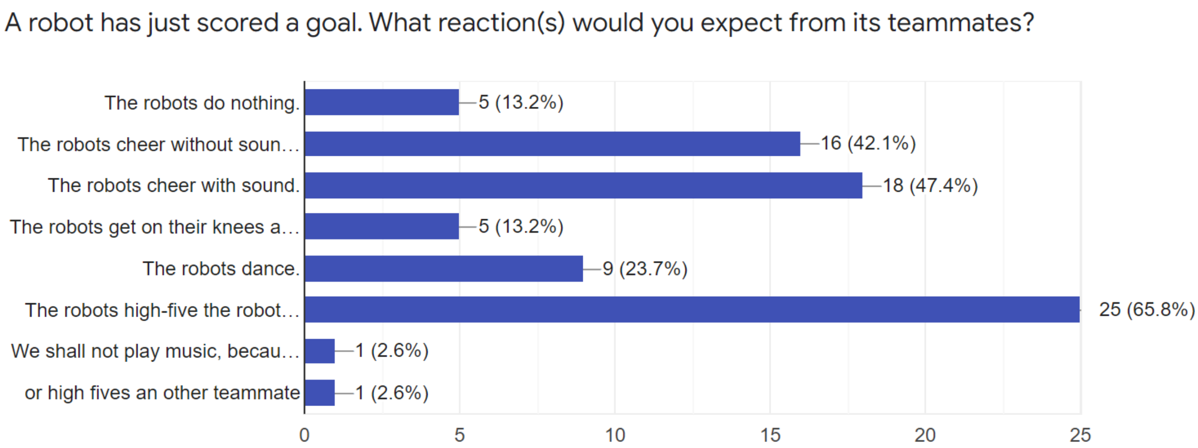

Figure 3.14: Participants thought the robots should high-five their teammate when they scored a goal.

- "The opposing team scored, what should the robots do that are not a goalie?": most participants thought the teammates of the goalie robot should shake their heads or cover their faces.
- "The team just won, how should the robots react?": participants agreed that the robots should dance and cheer wildly if they won the game. They also thought it would be proper if the robots high-fived the B-Human developers.
- And last but not least - "The team just lost, how should the robots react?": most participants found that it would be most appropriate if the robots sat down sadly, shook their heads or shamefully covered their faces.

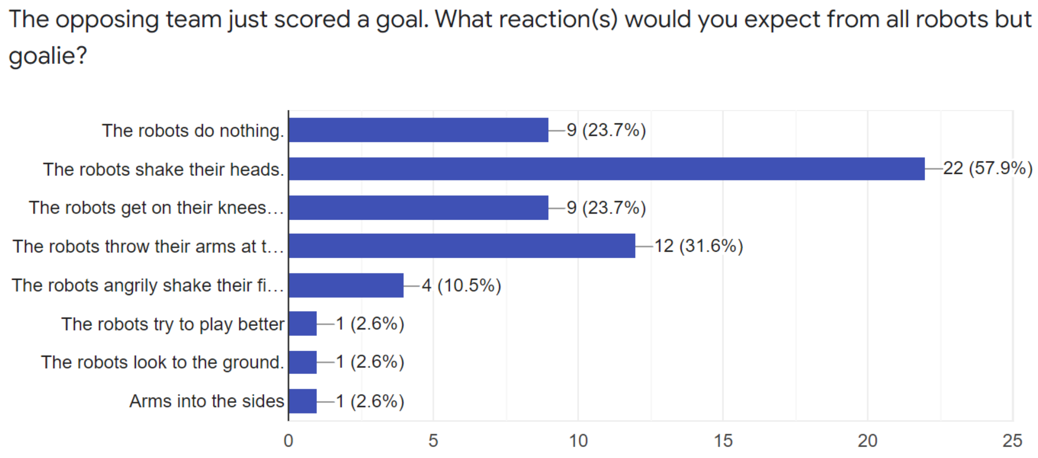

Figure 3.15: Participants thought a goalie's teammates should shake their head if an opponent scored.

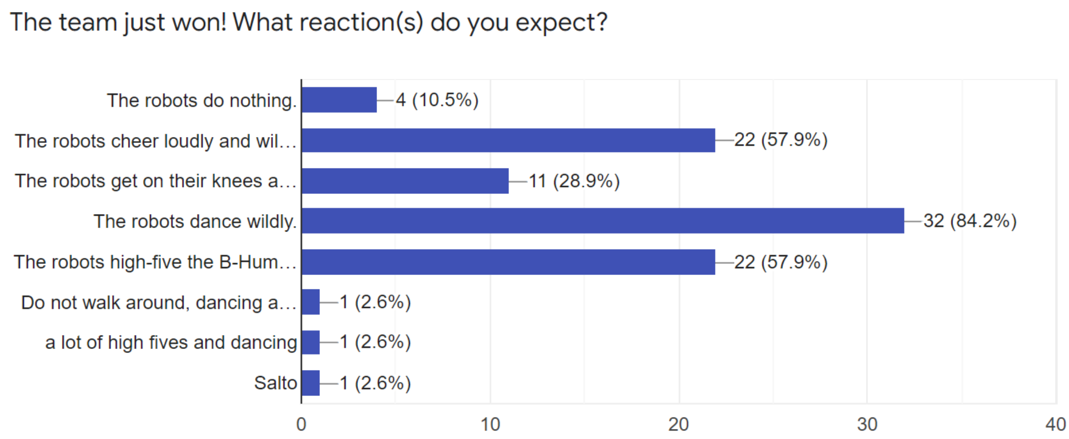

Figure 3.16: When the robot team won, participants thought the robots should dance and cheer wildly.

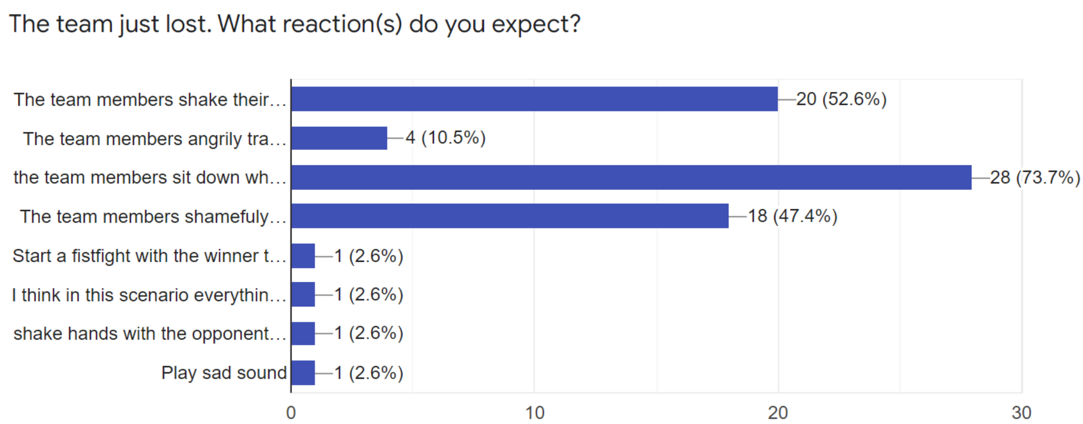

Figure 3.17: When the robot team lost, participants thought the robots should sadly sit down.

With all the answers gathered, the evaluation from section 3.2.2 was done. Some improper game-scenarios and behaviours were removed, while others were replaced. The feedback and scenarios sections offered a great insight into what game-scenarios were disallowed and what reactions the robots can't perform. They did not however include any "useful" game scenarios or behaviours and consisted mostly of people's fun fantasies instead, and were therefor disregarded.

Chapter 4

Implementation

This chapter describes the implemented features and the mechanics behind them. Where necessary, it provides both pseudocode and graphic images for further context.

4.1 Dynamic Motions

This section describes the mechanisms that introduce dynamism to the robots' motions. These mechanisms provide interfaces which also allow transforming any of the robots' existing motions into dynamic ones.

4.1.1 "Parameterizeable" Key-frame Motions

The motions described in the following section 4.3 use so called "key-frames". The way the key-frames work in the B-Human framework is by setting the target joint angles that the robot should be steered to, the motion duration and the interpolation type (linear/sinus interpolation), which determines whether the motion is done linearly or trigonometrically, in a configuration file, which is loaded onto the NAO during software deployment.
The problem with these pre-configured joint angles is obvious: there is no way to change them after deployment. This is where so called "Parameterizeable Key-frame Motions" come in. Whereas beforehand the joint angles needed to be hard-coded into the configuration file, now one only has to set placeholder values inside the configuration file instead (since the configuration file parser still requires numeric values) allowing one to dynamically set the angles during runtime. The mechanism works by receiving a list of target joint angles, one for each placeholder, during runtime and then executing the motions using the given angles.
The pseudocode below illustrates the way this works for some head joints:

The joints' configuration file:

```
head = [1234, 1234] // where 1234 is a placeholder value
```

A C++ class sets the joint angles:

```
vector angles = {10, 10}; // simple vector containing joint angles
DynamicJoints joints; // data class for storing joint angles
joints.setAngles(angles);
execute(someMotion, angles); // execute a given motion
```

The joint limitations mentioned in section 2.2.1 are handled by the code, where not providing a value to the placeholder simply causes the robot to use the corresponding joint's default angle (90 and -90 degrees for the left and right arms, 0 degrees for all other joints). Providing a too great or too small angle is dealt with by simply setting the joint angle to the maximum or

minimum value the joint can be steered to, based on whether the angle was greater or smaller than allowed.
While this mechanism is quite useful, since it removes the reliance on predefined joint angles, it still uses statically hard-coded values for executing motions (note that interpolation between key-frames, while being dynamic, is still only used between hard-coded, predefined angles, thus the motions are not truly dynamic). The next section attempts to solve this by introducing some randomness to the hard-coded angles, such that these might appear more natural to a distant observer, and have somewhat of a dynamic nature.

4.1.2 Dynamic Motions using Randomness

To make the hard-coded key-frame joint angles more dynamic, one might use randomness for adjusting the angles. The justification for this is that even humans, when moving their limbs, will never move them to the *exact same location* when doing the same motion multiple times in a row (obviously, since there is no way to measure what angle one's limbs were at during the first execution of a specific motion and because human motions are dynamic). After a lot of trail and error, the following *custom* formula was introduced:

$$\alpha' = \alpha + \frac{rand(offset)}{100} \cdot \alpha \cdot (-1)^{rand(1)}$$

Where:

- α' is the actual joint angle used for the key-frame.
- α is the hard-coded joint angle that is passed during runtime to the motion engine, which executes the behaviours on the robot.
- $rand(upperBound)$ is the randomizer function, which receives the upper bound as its one argument and generates a number n with $0 <= n <= upperBound$
- $offset$ is a number used to specify by what percentage the joint angle is allowed to be modified. *It has a lower bound of zero and an upper bound of 100.* If for instance, a joint angle should be offset by a maximum of 1%, $offset$ would be set to 1. The fracture division by 100 is used to convert it to a floating point number between zero and one. The reason this is done is denoted below 4.1.2 (*).

The $(-1)^{rand(1)}$ is used for randomly choosing an offset direction in either the positive of negative direction, e.g. the joint angle can be modified to both be greater or smaller than the original joint angle.
(*) The randomizer function uses the pre-existing randomness function from the B-Human framework which is only able to generate uniform integers. Therefor, in order to get a floating point number corresponding to the percentage of the offset the random $offset\_percentage$ has to be divided by 100.
In principle, all the formula does is generate an angle α' with:

$$\alpha - offset\_percentage \cdot \alpha <= \alpha' <= \alpha + offset\_percentage \cdot \alpha$$

for a joint angle α, but since the randomizer function can only ever generate uniform integers the abovementioned formula is used.
With the above formula applied the joint angles begin to show some signs of dynamism. *Since this change is barely visible, images have been omitted at this point.* The dynamism does however cause a new problem, which is quite visible: since the existing balancer in the B-Human framework is only able to account for key-frame motions with predefined, hard-coded joint angles within the configuration files, and with predefined joint offsets used for balancing out these

motions, it is unable to cope with dynamic joint angles generated at runtime. This causes the robots to tip and fall when losing their balance, or shake heavily when they don't. To solve this, a dynamic balancer was introduced, which is discussed in the following section.

4.1.3 Dynamic Balancing

Due to the dynamism present in the joint angles resulting from the mechanism discussed in the previous section, the robot requires a new balancing mechanism that can account for variable joint angles generated at runtime.
Since all the emotional reactions the robot will do (see section 4.3) only ever move the arms, the balancer does not need to take into account movements of the robot feet's joints, even though these can be dynamically manipulated, as it would greatly complicate the balancing mechanism. The balancer proposed here takes advantage of the fact that the robot stands up straight right before executing any of the dynamic movements and will therefor always have its Center of Mass (CoM for short) parallel to the ground before executing any of its motions.

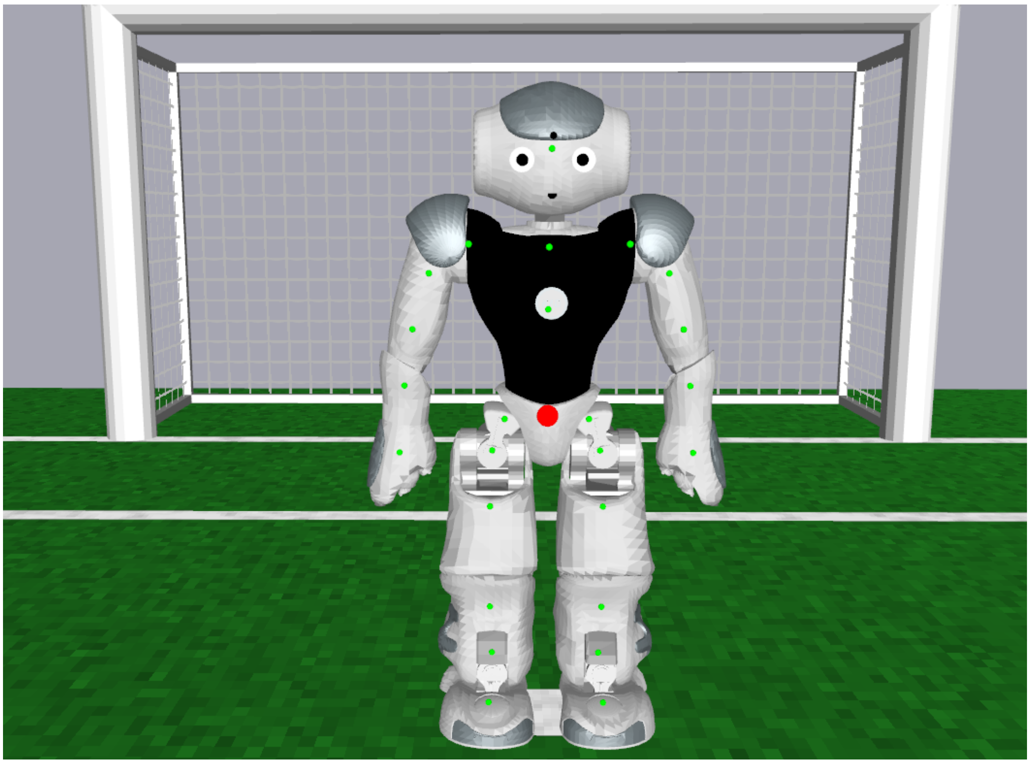

Figure 4.1: The robot's center of mass (red dot in the robot's hip).

In order for the robot to keep its balance, it needs to keep its center of mass (CoM) above its feet, such that the feet can still hold the robot's weight, because the entire weight of the robot's upper body, arms and even legs rests on its feet. If the CoM is outside the area of the robots' legs (as seen below 4.2), the robot will instantly tip and fall (likewise, if you attempt to tilt your body too far away from your feet, you will fall forwards/backwards).
The balancer works by saving the robot's CoM right before the balancer becomes active (just before executing a dynamic motion), and then computing the differential of its current CoM relative to its initial CoM (during the execution of the motion) and correcting its CoM accordingly using existing CoM manipulation functions from the B-Human framework. Technically speaking, the robot computes the coordinates of its CoM before executing the motion by using B-Human's inverse kinematics functionality to convert the current joint angles into a three-dimensional position vector $\overrightarrow{P_1}$. In the next frame (when the motion becomes active) it computes its new CoM,

again using inverse kinematics, to get a second three-dimensional position vector $\overrightarrow{P_2}$. It then computes the vector $\overrightarrow{P_2P_1}$, multiplies it by a "magic" number used for correcting the CoM and then uses B-Humans forward kinematics engine to compute the joint angles that should be used in the next frame by adding or subtracting the vector $\overrightarrow{P_2P_1}$ from the current CoM. The vector $\overrightarrow{P_2P_1}$ is multiplied by the "magic" number in order to shorten it, moving the CoM in the next frame further towards the original CoM. The vector is either positive or negative based on where the current CoM is (whether the robot is leaning forwards or backwards). This multiplicative process of shrinking the vector is done exactly once, since testing has shown that there is little to no effect in iteratively repeating it (similar to Newton's Iteration, where the factor by which the number shrinks starts getting smaller the more iterations are used). The "magic" number used is 0.01, which is the same number used in the balancer which B-Human uses when the robot is walking (the walk-balancer couldn't have been used here, since the arm movements are far too fast and chaotic in contrast to when the robot is strolling around the field, whereby it never moves its arms upwards).

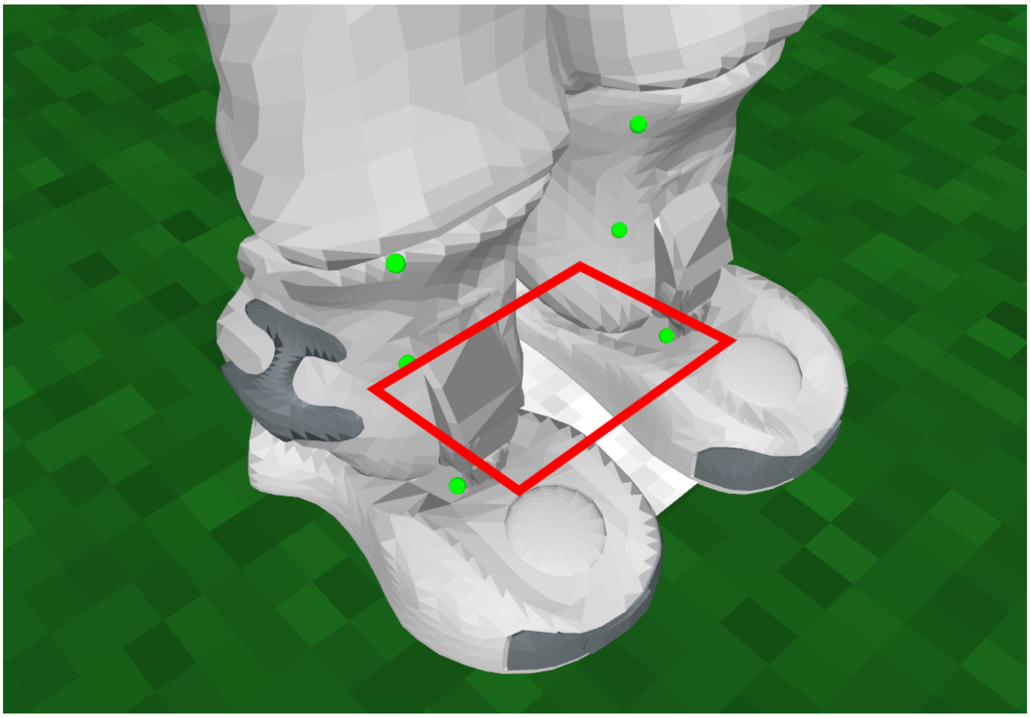

Figure 4.2: Constrains for the Center of Mass (CoM), leaving these causes the robot to fall.

After the motion ends, the balancer is deactivated and the robot retains its balance from the previous frame.
With the balancer implemented, the robot finally stops shaking when executing fast movements such as *dab* and *clap* (see section 4.3) and stops falling over when the motions are too quick and chaotic.

4.2 LEDs

As mentioned in Chapter 2.2.2 the robot has LEDs on its head as well as in its eyes, ears, chest and feet, all of which can be used in correlation with arm and head movements and sound effects to create somewhat "realistic-seeming" emotional reactions.
Where beforehand the robots would light one eye in blue and the other in a different color based on a robot's role in the team[1], now the robots should use their eyes along with all their other

[1] As seen in the LED source-code

LEDs to create an overall "image" describing the way a robot is currently "feeling". To achieve this, a dozen effects were created (*Note: the following pictures are partially taken using a low camera exposure time. This had to be done, as the LED effects would otherwise only be captured in white color, due to the camera being unable to take images in which some parts are very bright (the LEDs) and others are quite dark (the robot's limbs/body)*). In order to create these effects, a new interface had to be added to the code to actually be able to control the LEDs (as they were initially controlled by external factors). The following effects were added to this interface:

4.2.1 Eye LEDs

Note: each LED on the NAO can be set to one of three states, "ON", "OFF" or "HALF" (also called "middle").

- Green winking eyes (**default mode**) - Introduced for the robot to seem in a good mood whereby the winking denotes that the robot is a "living being", rather than a cold machine with static green eyes. As seen in chapter 2.2.2, the robot's LEDs are individually addressable. To create the winking eyes, the robot's eye LEDs on the bottom were set to the "ON" state while the rest of the LEDs were set to the "OFF" state. After the B-Human application is launched the robot starts a timestamp counter which increases by 1 every frame (there are 83 frames in a second with the NAO robots). The rest of the eye's LEDs only ever turn on if the counter modulo 89 results in zero. The number 89 was chosen with trial and error by adjusting it and watching the robot, then assessing how realistic the winking looked. With the number 89 the robot winks about once every second, which is quite a reasonable amount. *This same counter is also used for other animations, such as the spinning ear rings found below 4.10.* To prevent an integer overflow and reduce computation times (since the modulo operation on large numbers rises in complexity exponentially), the counter is reset once it reaches a number greater than 696 (a randomly chosen value which does not result in zero when modulo 89 is calculated, such that no animation is repeated twice: Once when the counter reaches the maximum value and once when it is reset to zero).

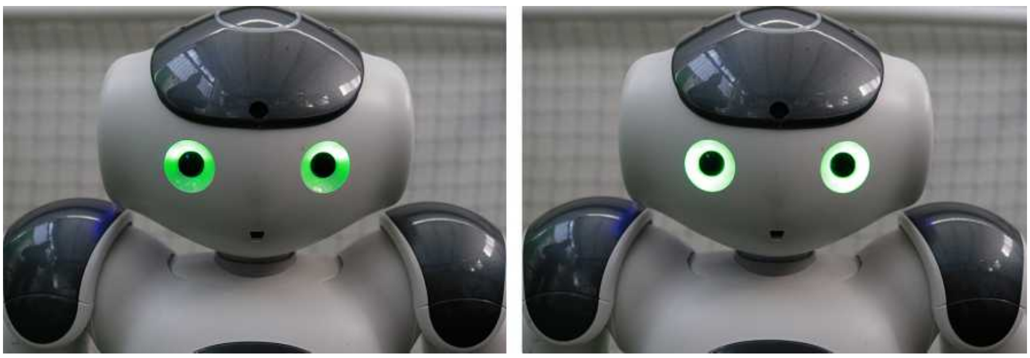

Figure 4.3: The robot's winking eyes.

- Static rainbow eyes - Used in "special" reactions wherever needed, like when dancing. See the pictures below 4.7, where the effect is additionally animated. This animation was created by setting each of the eye LEDs' diodes according to the location of the LED, where the first LED starts with the red diode set to "ON" and the blue and green diodes set to "OFF". The next LED sets the green diode to the "HALF" state, while keeping the blue diode off. Then the next turns the green LED to full brightness, and so on, until eight of the rainbow's colors have been created (because each eye only has 8 LEDs). Since some colors don't seem realistic when created (such as the color orange, which looks quite green), these are not used.

- Red "sad" eyes - The bottom half of the eyes is colored in red, denoting that the robot is sad. This is used in reactions where the robot should show annoyance or sadness, if for instance the opponent scores a goal. This was simply done by setting the robot's lower eye LEDs to the red color, while turning off the rest of an eye's LEDs.

Figure 4.4: The robot's "sad" eyes.

- Red eye ring that goes around in circles (nicknamed the "red ring of death" after the famous "red ring of death" that occurred in Xbox game consoles [30, Xbox Wiki]) - Used when the robot is very angry or sad. To create this effect, within each of the robot's eyes three red LEDs are set to the "OFF" stage, three to the "HALF" stage and two to the "ON" stage. A new variable "rotation" is introduced and assigned the initial value 0. For each of an eye's LEDs, the counter is incremented by one and calculated modulo 8 to determine what state the current LED should be set to ("ON", "HALF" or "OFF"). To make the eye-ring spin and control the spinning speed, the timestamp counter from 4.3 is used to change the rotation counter's initial starting value every n-th frame, again using modulo operations (for instance if the timestamp counter modulo 3 equals zero, the counter would be incremented by one, thus making the eyes spin. To control the speed, this number of frames can be set to either 5 or 7 as well, thus slowing down the rotation. The higher the number, the more frames need to pass for the animation to continue, the slower the animation).
- Flashing white eyes (disco effect) - Used for dances and special emotes. The timestamp counter is used here as well to determine whether the eye LEDs should be turned on or off. The divisive numbers 7, 5 and 3 are again used to control the speed of the flashing, offering either slow, medium or fast flashing eyes.

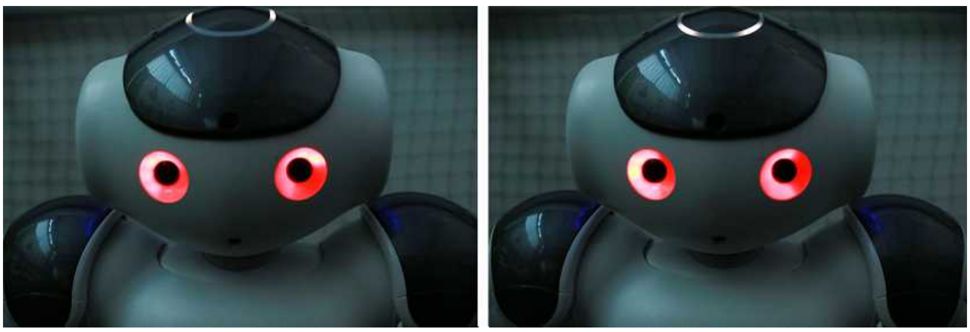

Figure 4.5: The "red ring of death" effect.

- Rainbow ring that goes around in circles - Use cases are the same as for the previous effect. This ring uses the same mechanism that the "red ring of death" uses to rotate 4.5.

- Red-Green-Blue (RGB) cycle - Not used in this thesis, but can be used for dances and special effects. This effect also uses the timestamp counter to determine which color the eyes should be set to, with the speed being controllable as well.

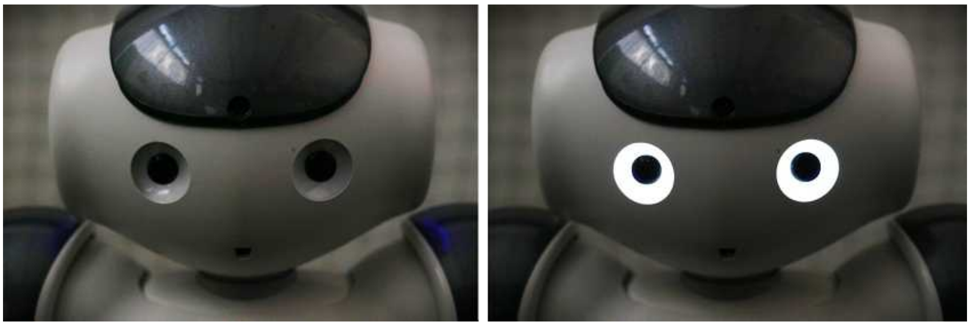

Figure 4.6: The disco eye effect, where the LEDs rapidly flash on and off.

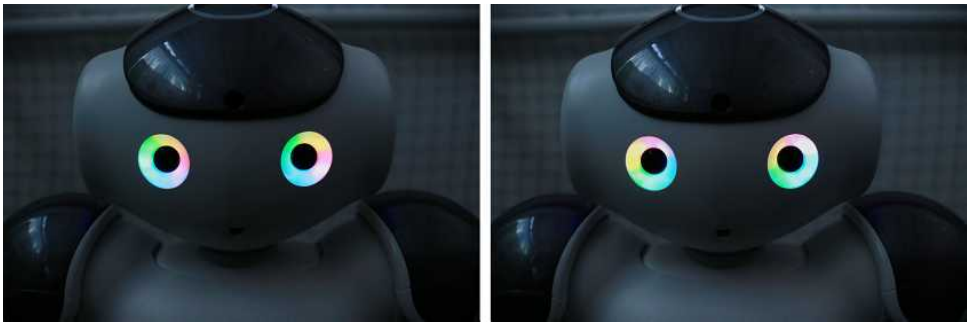

Figure 4.7: The rainbow ring effect. The ring is spinning anti-clockwise.

Figure 4.8: The RGB cycle effect. The colors jump from one to the next.

- Static colors - Can be used whenever desired for debug purposes or for creating custom reactions. Pictures have been omitted, since it offers most of the colors from the RGB cycle, adding only yellow, magenta and white.

Since the eyes have the most LEDs compared to the robot's other body parts and all of them offer all three RGB (Red-Green-Blue) colors, they are the main contributor to the overall emotional "image" portrayed by the robot.

4.2.2 Ear LEDs

- Static blue - **Default mode**

- Flashing on/off (disco effect) - Used for special reactions, such as dances. This also uses the frame-based animation mechanism that the eyes use (See 4.5).

- All LEDs off - Can be used when desired, *unused here.*

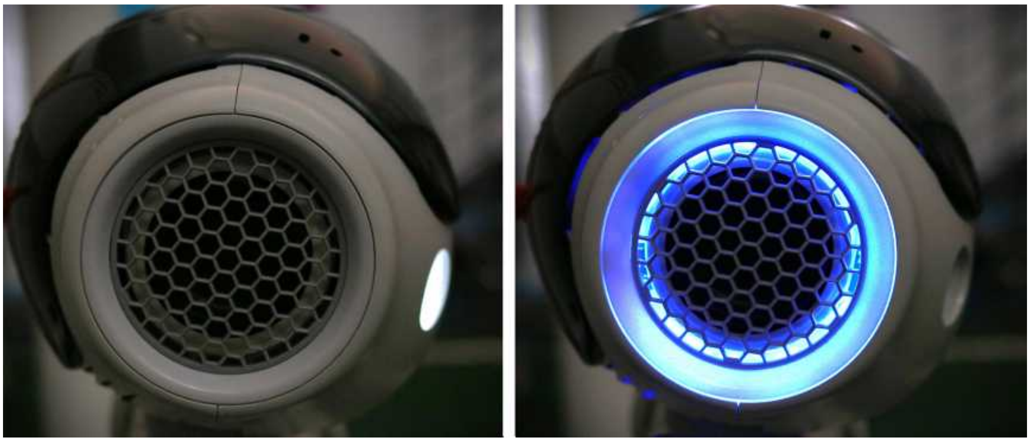

Figure 4.9: On the left: all ear LEDs are off. On the right: all ear LEDs are on.

- Ear LEDs rotate pairwise (one LED on, the next off) in circles - Used when dancing. This effect uses the timestamp counter described in 4.3 to control the rotation and its speed. To determine what LEDs should be lit, the effect uses an additional logical switch (an integer set to zero or one) to check whether the odd or even LEDs were lit in the previous rotation, and based on which LEDs were lit, it determines which LEDs should be lit in the current frame.

Figure 4.10: Ear LEDs rotating clockwise.

Because of the ear LEDs' locations and the light bleed from one ear LED to the next (since they have a diffusion ring), these effects have a much smaller effect on the robot's overall "image". However, since even the smallest details help in creating a more realistic reaction, these were included as well.

4.2.3 Head LEDs

- Static all LEDs on - **Default mode**.
- Flashing on/off (disco effect) - Used for special reactions. Uses the same frame-based animation mechanism as the ears and eyes.

- All LEDs off - Used during normal game-play.

Figure 4.11: On the left: head LEDs off. On the right: LEDs on.

- Rotate LEDs in pairs - Flashes LEDs on and off in pairs (one LED on, the next off) in circles. This is used for special reactions, like dancing. Similar to the rotating ears 4.10, this animation also uses a (separate) logical switch for determining which LEDs were lit in the previous frame and thus which should be lit in the current frame.

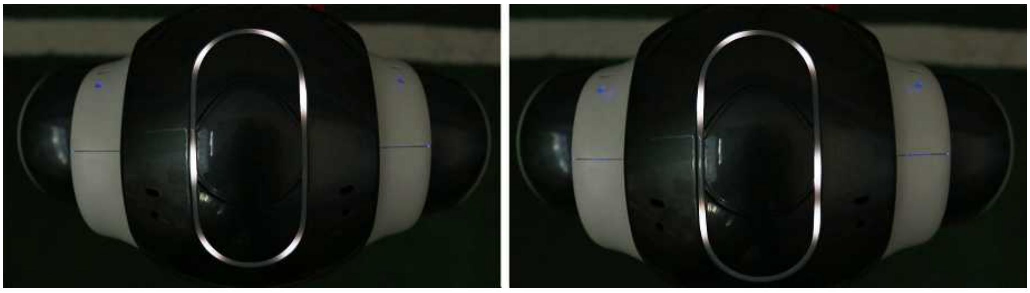

Figure 4.12: Head LEDs rotate pairwise.

The head LEDs are very dim, and the effects are therefor not easily visible on their own. As with the ear LEDs however, using them in combination with other LED effects enhances the overall authenticity of the robot's reactions.

4.2.4 Chest and Feet LEDs

(These work identically, but can be individually controlled. They offer RGB colors)

- Static blue (**default mode**) - Used during game-play.
- Static color - Can be used for debug or custom reactions, *unused here.*
- Rainbow cycle - Since these LEDs only offer displaying one color at a time, effects such as the rainbow-ring in the eyes are not possible. Therefor, the rainbow cycle effect goes through all the static colors available in a timed-cycle. This effect is used in special reactions, such as dances. It also uses the timestamp counter from 4.5 and modulo operations like all other animations do to function.
- Flashing white (disco effect) - used in other special reactions, like cheering. This effect is also animated using the frame-based approach with the timestamp counter, with controllable speed like all other animations.

Figure 4.13: Chest button and feet LEDs showing RGB colors

The feet LEDs are very dim and therefor contribute little to the overall "image" expressed by the robot. The chest button, however, is very bright and quite big and when used along with the eyes contributes a lot to the overall appearance of the reaction.
When used together in sync the LEDs can be used to create quite unique reactions, and when coupled with movements the overall "image" portrayed by the robot looks quite extraordinary.
With the LED effects implemented, they can now be coupled with movements to create emotional reactions.

4.3 Emotional Reactions

Inspired by pop culture (and Fortnite [31] emotes) and based on the initial user survey, the following emotional reactions (also called "emotes") were created (pictures are included in each reaction's subsection):

- Dances:
  - The all too famous "Macarena" dance from Los Del Rio [32, YouTube].
  - The "Floss" dance, which originated from then 16-year-old Instagram celebrity Russell Horning [33, O. Waring, 2018] and has become quite famous in the past two years thanks to Fortnite and teens on TikTok.
- Reactions:
  - A basic "Cheer" emote, which the robots can use whenever a goal has been scored, or just generally, if they are standing on the side of the playing field.
  - A "Clap" emote, which can be used if a goal is scored or for demonstration purposes.
  - The famous "Dab" emote, which is believed to have originated in the Atlanta/Georgia rap scene [34, Merriam-Webster] and has also become quite famous in the past years thanks to Fortnite and TikTok. Most famously, a slightly modified version is used by Usain Bolt for celebration after winning a marathon (As seen below 4.14).
  - A so-called "Facepalm" emote, where the robot throws one arm at its face.
  - A "Shaking Head" emote, where the robot covers its face with both of its hands and shakes its head from side to side. And finally,
  - a "Stand for National Anthem" emote, where the robot stands up-right, places its right hand on its chest, sets all of its LEDs to the color(s) of its country and plays the national anthem.

Videos for all the reactions are referenced in chapter 5.2.5. The reactions can be played back as often as desired by changing the amount of times they should be executed during runtime.

Figure 4.14: Usain Bolt's "Lightning Bolt" celebration[35, B. Smiley, 2016]

4.3.1 Macarena Dance

As mentioned above, the robot dances the "Macarena" dance to the all famous song by Los Del Rio. The dance starts by stretching out one's arms (first the left, then the right), then twisting them. One then proceeds to place one's arms on one's shoulders, then on one's head and afterwards on one's hips. Finally, one shakes one's hip, turns 90 degrees clockwise, and repeats. The steps are illustrated in the image below 4.15.

Figure 4.15: "Macarena" dance steps [36, Pinterest]

Sadly, since the robot has limited joint flexibility, the dance had to be slightly modified. The robot starts off the dance by raising its arms and then twisting them, just like the original dance, but then proceeds to turn its arms inwards, since it cannot place them on its shoulders. It then proceeds to lift them above its head and then lowers them towards its hip. To finish off, it throws them up into the air and then gently brings them back down. Then, it turns 90 degrees

clockwise and repeats the motion.
The robot can repeat the motion as often as desired by setting the amount of 90-degree-turns it should do when executing the reaction. By default, it does one rotation and then ends the motion.
While performing the motions the robot plays a shortened version of the Macarena song by Los Del Rio. Copyright is not an issue, as it falls under the Fair Use act of the copyright laws of 1997.

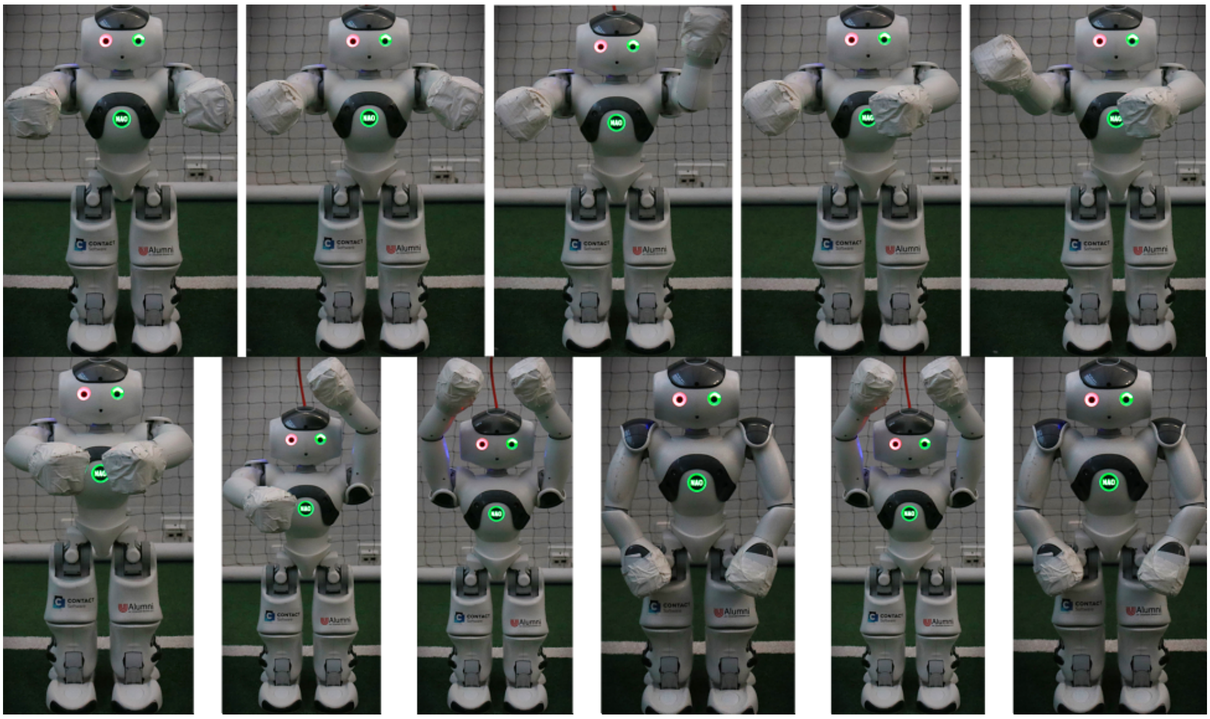

Figure 4.16: The "Macarena" dance as performed by a NAO robot.

4.3.2 Floss Dance

The "Floss" dance by Russell Horning starts by moving one's right arm backwards while moving one's left arm forwards. Then one proceeds to move the right arm behind one's back to the left while simultaneously moving one's left arm to the right across one's chest. Then, each of the arms is retracted, the left arm is moved backwards and the right arm forwards. Afterwards, the right arm is moved across the chest to the left while the left arm is moved behind one's back to the right, before both arms are retracted. These four basic movements make up the so-called "Floss" dance and are usually performed very quickly. There is no pre-set amount of times the motion needs to be repeated - one can do it one time, or a hundred.
With "Floss" the robot has a much easier time than with "Macarena", as the limited joint flexibility allows performing almost all the movements without an issue. The only issue that arises is when moving the arms sideways, as the robot is limited in how far the arms can be twisted before being blocked by the robot's body or legs. This can be countered by simply reducing the angle by which the arms are twisted.
For "Floss" a slightly modified version of the music used in the popular video game "Fortnite" [31] was composed by Mr. Phillip Dolata, and is played while the robot performs the emote.
Along with the music, a couple of the LED effects described in the previous section 4.2 were used to make the emote seem less static. The robot's eyes use the "rainbow ring" effect, the ears use the pairwise LED rotation effect, the head uses the "Rotate LEDs in pairs" effect and

the chest and feet use the "rainbow cycle" effect. Additionally, some dynamism is added to the joint angles, such that the robot's arms are never quite at the same position when repeating the motion multiple times (see section 4.1).

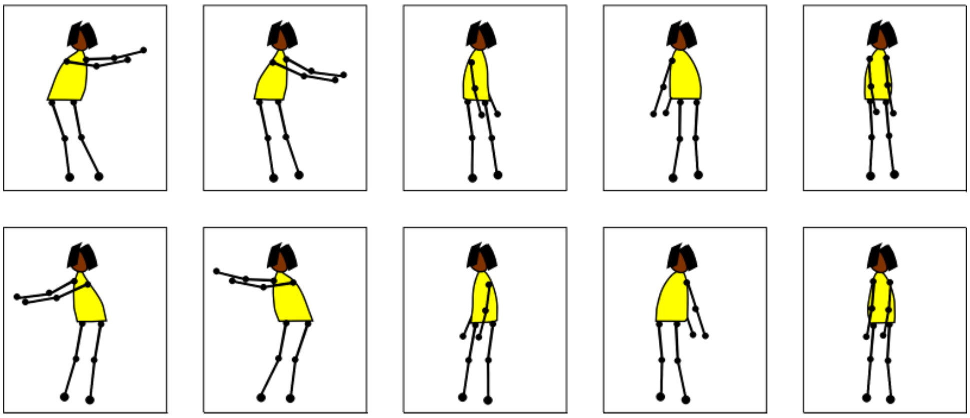

Figure 4.17: The "Floss" dance [37, NicePNG]

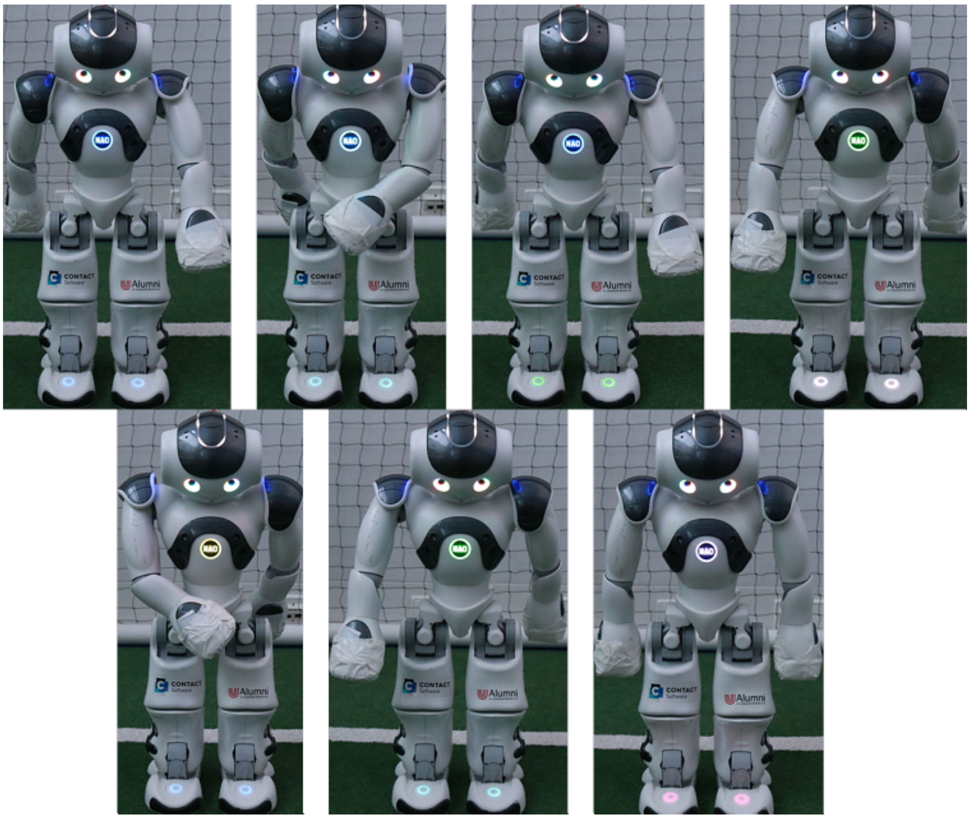

Figure 4.18: The "Floss" dance performed by a NAO robot.

By default, the emote is played once (and the dance is repeated twice) and can be repeated as often as desired. It is pre-configured to repeat the floss dance twice.

4.3.3 Cheer Emote

For the cheer emote the robot simply lifts its hands up in the air and gently moves them up and down. This reaction uses the "disco" LED effect on all the robot's LEDs and plays a basic cheering sound.

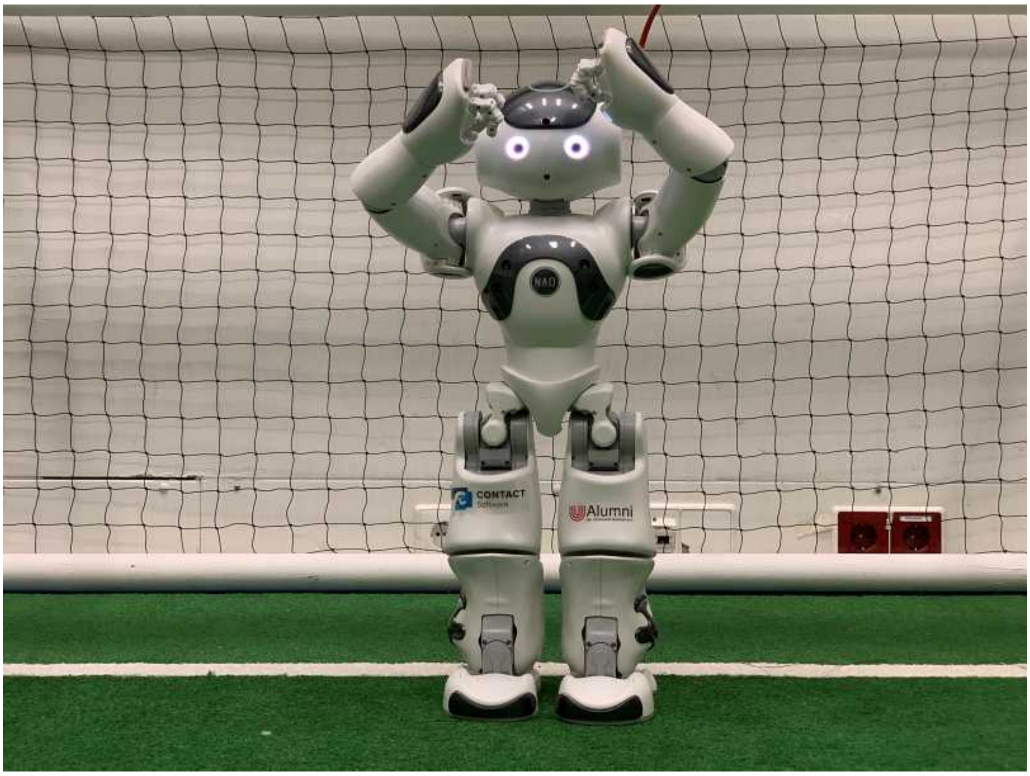

Figure 4.19: The "Cheer" emote performed by a NAO robot.

4.3.4 Clap Emote

Clapping was quite a challenge to implement, as the robot's hands are taped shut during soccer games, and hitting the robot's fists together would damage the hands (which, similar to the robot joints' gears are also made out of fragile plastic). Therefore, instead of actually making the hands touch when clapping, the robot moves its hands very quickly close together, to give the sensation of clapping, without the hands ever touching one another. To still seem realistic, a clap sound is played when the hands are at their closest meeting point.

4.3.5 Dab Emote

For the "Dab" emote the robot needs to shortly lift both arms in one direction. This poses two problems:

- The speed at which the arms need to move and
- The robot's balance while rapidly moving its arms.

The balancer from a previous section 4.1.3 solves both of these problems, which allows the robot to do the "Dab" quite realistically. To make it seem more interesting, a sound is played when the arms are raised and all the robot's LEDs go into disco flash mode. An image of the robot can be seen below 4.21.

4.3.6 Facepalm Emote

With the so-called "Facepalm" emote the robot would need to cover its face with its hand. Since it is unable to do so, both due to its fists being taped closed during soccer games and because its fingers are too short, the robot simply raises its fist at its head in a mediocre speed. To add the sensation of anger or sadness, the robot's eye LEDs are set to the "sad" effect, while the other LEDs are all set to a static red color. Additionally, a short annoyed sound is played. The image below 4.22 showcases this reaction.

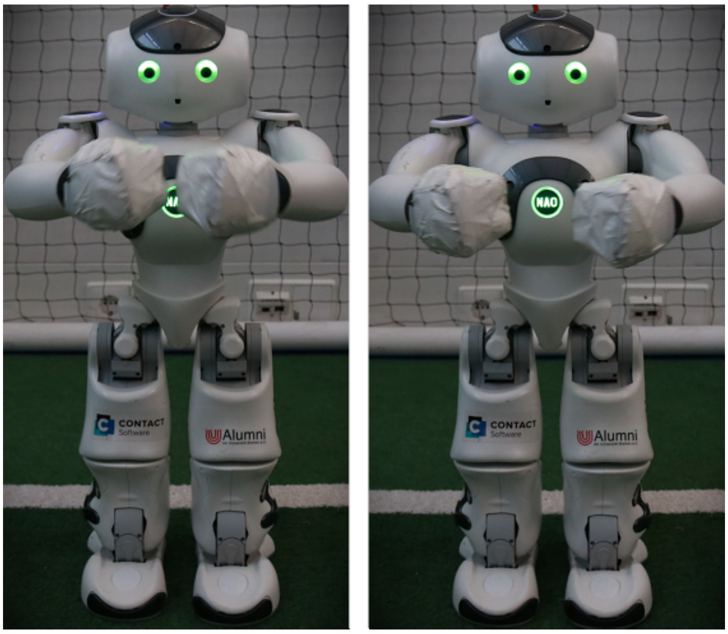

Figure 4.20: The "Clap" emote performed by a NAO robot.

Figure 4.21: The "Dab" emote performed by a NAO robot.

Figure 4.22: The "Facepalm" emote performed by a NAO robot.

4.3.7 Shake Head Emote

For the "Shake Head" reaction, the robot quickly raises its arms and covers its face using both of its hands, then proceeds to shake its head from left to right. It does this twice, stops for a brief moment, and then does it again. As with all other reactions, this can be done as often as desired.
While shaking its head, the robot's eye LEDs are set to the "red ring of death" effect and the rest of the LEDs are set to a static red color. No sound effect is played, as the reaction is already quite expressive, rendering a possible sound effect obsolete.

4.3.8 Stand for National Anthem Emote

Last but not least, just like real soccer players before a football match, the robots have been given the ability to stand "respectfully" for their national anthem.
This reaction is, as in real soccer, intended to be used before a match is started. The robot team set their LEDs' colors to that of their country (or the color of their RoboCup team, whichever the team's developers prefer) and then play the national anthem while holding their right hand over their "heart".

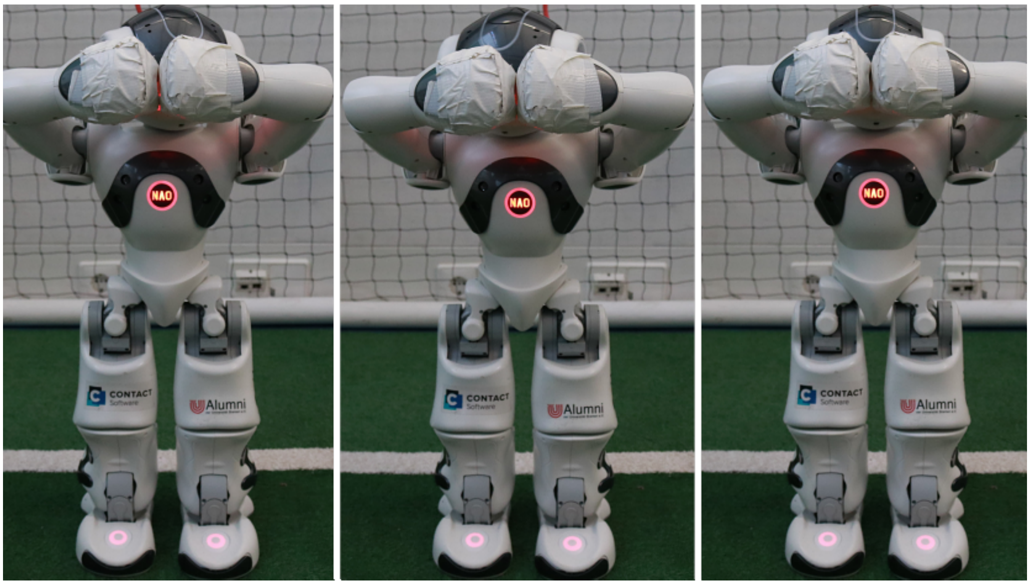

Figure 4.23: The "Shake Head" emote performed by a NAO robot.

Figure 4.24: The "Stand for National Anthem" emote performed by a NAO robot.

With all of these reactions implemented, all that is left to do is evaluate them and then make the robots use them in specific game scenarios. As mentioned in the beginning of this thesis, using the reactions in specified game scenarios will be done in Mr. Phillip Dolata's follow-up thesis "Situative Darstellung von Emotionen durch humanoide Fußballroboter".
In contrast, this thesis will evaluate how authentic the reactions look and whether a distant observer can figure out what a robot is meant to be doing. The evaluation is discussed in the next chapter.

Chapter 5

Evaluation

This chapter evaluates the implemented features from the previous chapter. The dynamism is first evaluated using manual testing, while the reactions are evaluated empirically using a second user survey.

5.1 Dynamism

Since the dynamism is merely used for creating more realistic looking reactions and its effects are very subtle, it was evaluated manually. This was done by repeatedly playing back the robots' reactions and measuring the joints' angles during these. The offset percentage during all gameplay reactions is set to 1% by default, since a larger offset would greatly increase or decrease the angles, in which case the reactions are no longer recognizable.
The joints show a variable yet random range of angles (for instance, a 45° angle shows a wide range of numbers between 44.55° and 45.45°), just as they are supposed to. Too great or too small angles are automatically accounted for and taken care of by the code.
Since these angles were measured using syslogs, there is no graphical representation to be shown at this point.
In contrast, the dynamic balancer was tested in the real world by playing back the robots' reactions dozens of times during the development process. Where beforehand the robots were shaking madly while executing their emotional reactions and actually fell a couple of times (even while filming a demo video), movements like "Dab" which caused the robot to fall are now executed very smoothly, and the robots manage to keep their balance throughout the dozens of photo shoots they did for the pictures found in the previous chapter(s). This is furthermore quite apparent in the videos from the second survey 5.2.5.
As for the parameterizeable key-frame motions, these are a mere technical feature which was evaluated using white-box tests to make sure all joint limitations were accounted for. With all test cases covered (too big/small joint angles and/or no angle provided for some placeholder) the feature is assumed to be working correctly without further needs of evaluation.

5.2 Reactions

To evaluate how authentically the emotional reactions were implemented, a second survey was conducted. For this, the participants from the initial user survey 3.2 were again contacted and asked to evaluate the robots' reactions.

5.2.1 Motivation

Since the goal of this thesis is creating emotional reactions that look as authentically as possible and create engagement with the audience during robot soccer games, keeping users at the center of the development process is essential. Therefor, the best way of evaluating how genuine the robots' reactions feel is by asking "normal" everyday people to evaluate these. Since the participants of the initial survey were the ones that determined which reactions got implemented, it would only be proper for them to evaluate these as well.

5.2.2 Preparation

For the second survey, short videos of each of the robots' reactions were filmed. The survey was again split into multiple sections, each pertaining towards gathering data on different aspects of the robots' reactions:

- The first section was dedicated towards gathering demographic data, although in a much more simplified manner in contrast to the first survey. This time, participants were only asked whether they were a part of the B-Human development team, and whether they are university students, for classifying them with the data gathered in the first survey.
- Following the first section were eight sections, one for each of the robots' reactions. Each section consisted of the corresponding video in which the robot performed the reaction and was followed by two evaluative questions:
  - "Could you identify the emote as what it is supposed to be (Emote Name)?" and
  - "What did you think of the reaction?"

  The goal of both of the first question was to determine the authenticity of the implemented reaction, and the goal of the second question was to determine how deeply participants connected with the robots emotionally while they were doing the reactions.
- The tenth section asked participants to determine whether they found specific reactions fitting for certain game scenarios. This was again due to the joined work with Mr. Phillip Dolata. The results are hence minimally used in this thesis' evaluation of the survey.
- The last section was used to gather further feedback on the reactions and the survey in general, giving participants an option to further share what they thought about the robots' reactions.

5.2.3 Execution

The survey, similar to the first one 3.2, was conducted using a Google Forms [29] questionnaire. Participants received a link with which they could access the questionnaire. No time limit was set and participants were able to jump back and forth between questions as they pleased. The robot videos were uploaded to YouTube and participants could play these back either in an embedded form in the questionnaire or externally in the YouTube app.
To each reaction's question about whether they were able to correctly identify the emotional reaction participants could answer with "Yes", "No" or "Kind of". For the second question, participants could answer that they liked the reaction ("Liked it"), kind of liked it ("Meh") and "Didn't like it". Additionally, a text entry field was provided for providing manual answers.
Participants were unable to see how other participants answered the questions, and did not get an overview of all the survey's answers.
The survey was publicly available for three weeks, after which it was set to no longer accept responses. It was sent by email to the participants of the previous survey and posted on social media groups, such that other interested people might also partake in it.

5.2.4 Evaluation

Out of the initial survey's participants, 24 participants partook in the second survey. This time however, three-quarters of them were *not* part of the B-Human team. This is quite useful, since the reactions are mostly targeted at people not familiar with the robots' behaviours. Most of the participants were students, a lot of which are most likely in IT fields, as the first survey had shown earlier 3.2. Below are the demographics of the second survey.

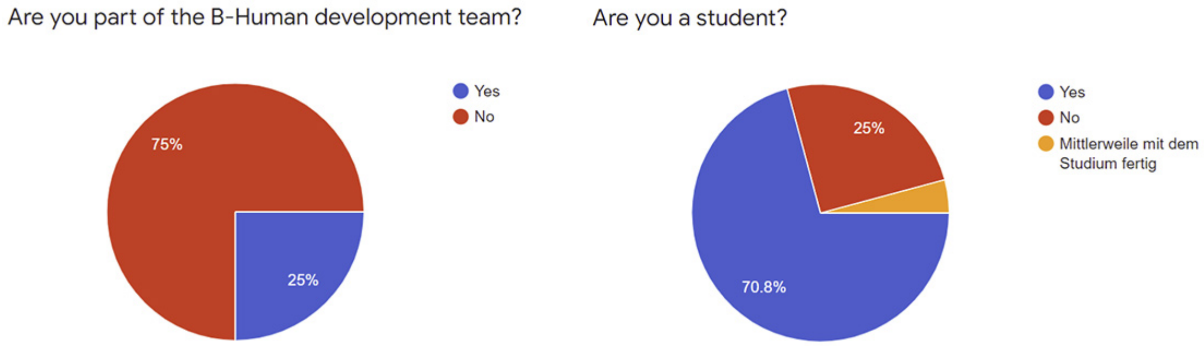

Figure 5.1: Demographics of the second survey.

The first reaction participants were presented with was the "Floss" dance. Half of the participants were able to correctly identify the dance and two thirds liked it. One participant did not know what the floss dance was. Since the majority of participants were students, and one third of them answered the recognition question with "kind of", it can be assumed that the sole participant is probably of an older generation and thus unfamiliar with the dance. Participants that did not like "Floss" thought the music was annoying, and said the dance was too slow. The charts below illustrate this.

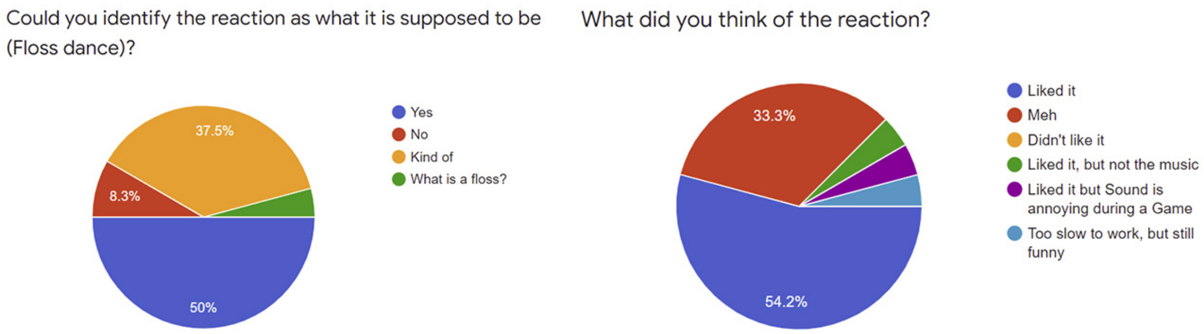

Figure 5.2: "Floss" dance evaluations.

Next participants were shown the "Macarena" dance. Perhaps unsurprisingly (due to the popularity of the Macarena song), all participants were able to at least kind of identify the robot's reaction. When asked whether they liked it, half of participants answered that they did like it while 40% answered negatively.
One participant pointed out that the motion is too long (others pointed this out as well in the feedback section) and claimed the sound can not be used in real games. While the point about the reaction's length stands, the second point does not, as pointed out before 4.3.1.

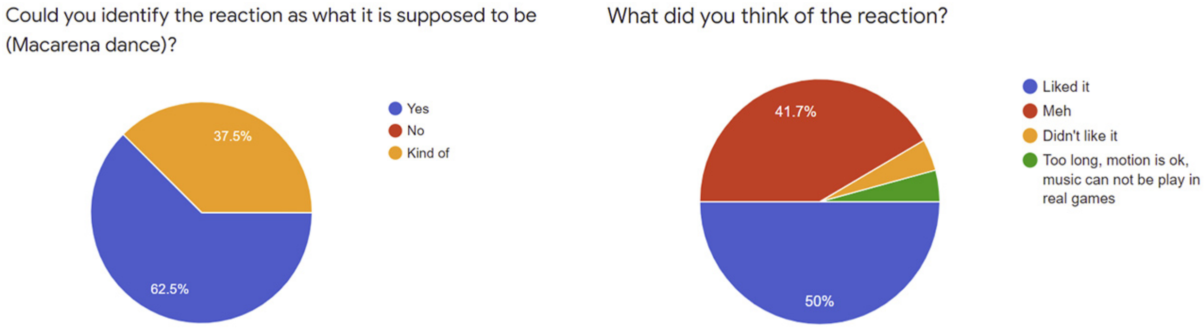

Figure 5.3: "Macarena" dance evaluations.

Following the "Macarena" dance was the "Cheer" emote. Out of 24 participants, 17 were correctly able to identify the reaction as a cheer and 15 liked it. 12% of participants did not like the reaction and two pointed out that the cheering sound is annoying. Another pointed out, that the reaction is repeated too often. Since the robots have the ability to play back the reactions as often as desired, the default amount of repetitions for "Cheer" was adjusted accordingly.

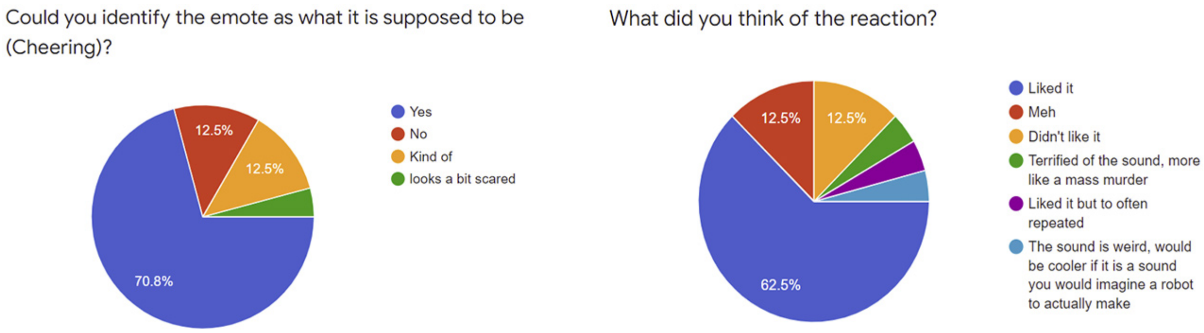

Figure 5.4: "Cheer" emote evaluations.

The next section evaluated the "Clap" emote, which seems to have been the least realistic emotional reaction. Only slightly more than a third of participants correctly identified the emote, while the rest did not. This was also the least liked reaction, getting a mere 41% of participants' positive votes. As was pointed out by one participant (and multiple more in the feedback section), the reason the emote was not liked was because of its speed and appearance. One participant wrote: "The robot raised its hands for the initial clap too fast", while another found that the clapping reaction "seemed weird". The charts below 5.5 further illustrate this.
Out of all the emotional reactions, clap ranks as the least liked and least recognizable, prompting for a tremendously necessary overhaul to it, before it can ever be considered useable in live RoboCup games.
The "Dab" emote followed next. Surprisingly, all participants were able to correctly identify the reaction, and it received a massive 87.5% of participants' up-votes. Only two participants did not like it, while a third only disliked the sound effect that was present during the reaction's execution. With this, the "Dab" reaction ranks as the sole most correctly identified and liked reaction in the entire survey.
The sixth reaction participants had to evaluate was the "Facepalm" emote. Half of the participants were correctly able to recognize the reaction while another third only kind of did. When asked whether they liked the reaction, 41% answered positively, 20.8% negatively and the remainder only kind of did. The main critique point was the odd sound the robot played while the reaction was being executed.
Next participants were asked to rate the "Shake Head" emote. Unsurprisingly, two thirds were

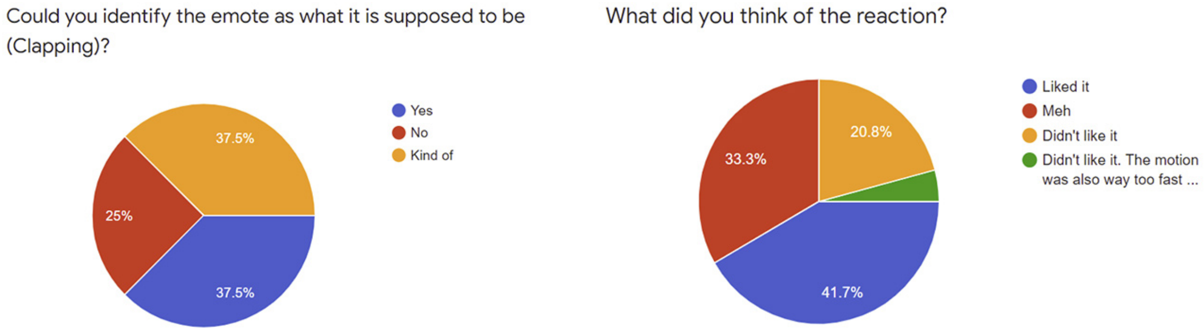

Figure 5.5: "Clap" emote evaluations.

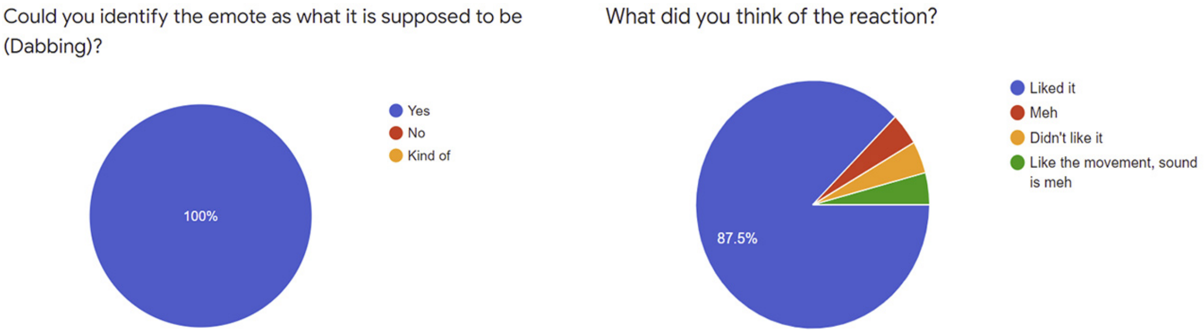

Figure 5.6: "Dab" emote evaluations.

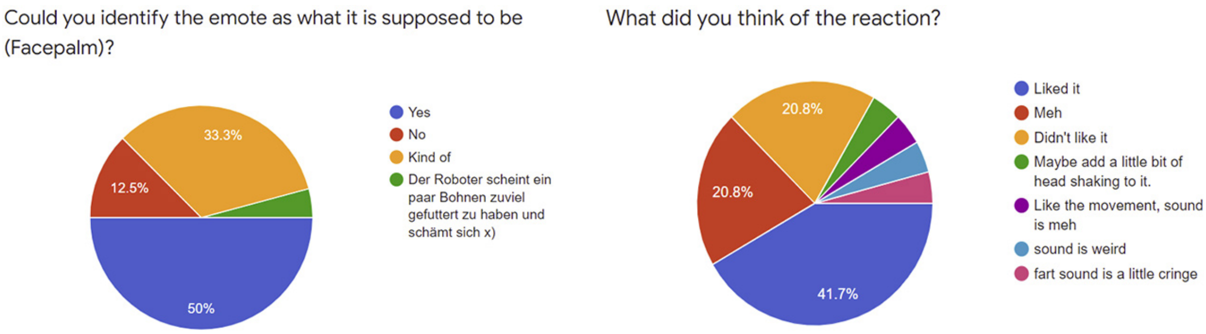

Figure 5.7: "Facepalm" emote evaluations.

correctly able to identify the reaction and 70% of participants liked it. There were no specific reasons provided by participants that did not like the reaction (disregarding the answer about the leg movements, since these do not move during the reaction. Perhaps a spelling mistake). Last but not least, participants were asked to evaluate the "Stand for National Anthem" emote. Surpassed only by "Dab", the "Stand for National Anthem" emote ranks as the second most identifiable and liked reaction, being recognized by 22 out of 24 participants and liked by 83.3% as the charts below 5.9 showcase.

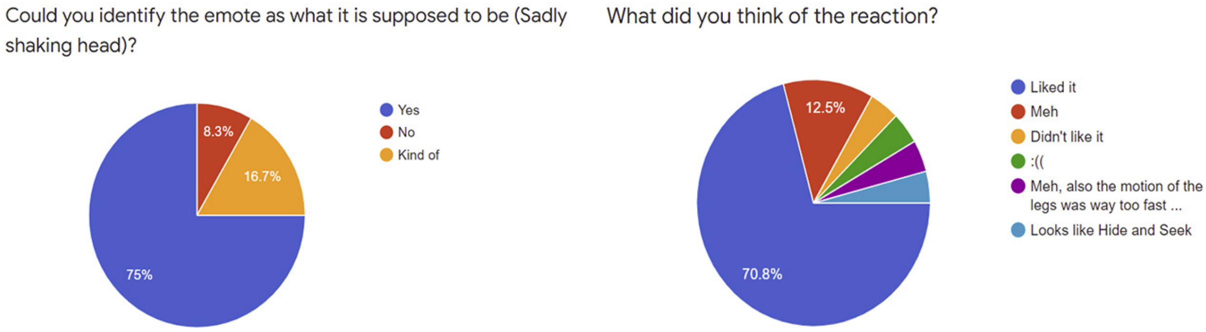

Figure 5.8: "Shake Head" emote evaluations.

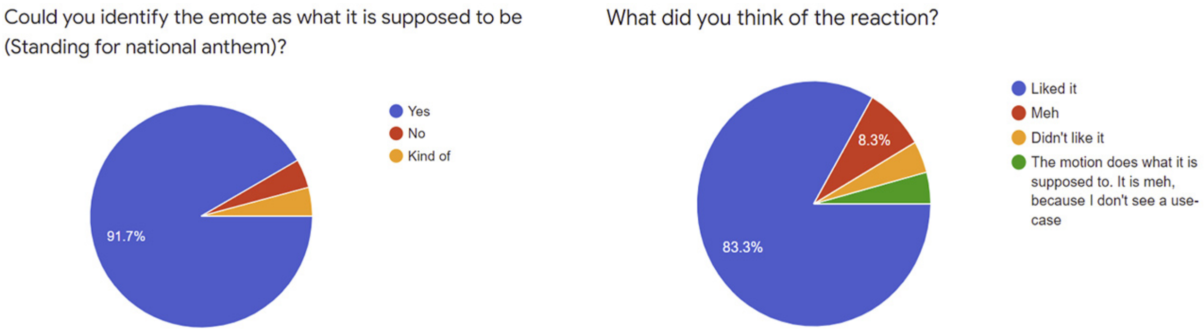

Figure 5.9: "Stand for National Anthem" emote evaluations.

Overall, the majority of participants still seemed to like most of the reactions, with the only main points of critique being the "Clap" emote, the sounds in the "Facepalm" and "Floss" dances, and the duration of the "Macarena" dance and the "Shaking Head" emote.
With the separate reactions evaluated, participants now had to evaluate how fitting they found certain reactions to be when using in certain game scenarios. As mentioned multiple times, this is not a major point of interest in this thesis, but seeing as using the reactions during soccer games is the main objective, and since these can greatly change the perception of the emotional reactions, a brief breakdown of the results regarding using the reactions during game-scenarios is provided below.
Note: The percentages below refer to how many participants votes with "Yes" when asked whether they thought a reaction fits to a specific game scenario. They do not refer to the overall number of votes.

- When asked about reactions being used in specific game scenarios, four fifths of participants found that cheering is the most appropriate reaction to play when scoring a goal, followed by the "Floss" dance and the "Dab" emote, both of which got 70% of participants' votes.
- The least appropriate reactions when scoring a goal were the "Macarena" dance, especially because of its long duration, as well as the "Clap" emote, which participants found the least realistic.
- When the opponents' team scored a goal, participants found it most appropriate for the robot to shake its head ("Shake Head"). Regardless, the "Facepalm" emote still received 70% of participants' votes believing it should be played when the opponents score a goal.
- To the question, whether the robot should stand for their national anthem at the beginning of a game, 70% of participants answered with "Yes".

Based on the feedback section, and the abovementioned results, it seems as though participants enjoyed the robots' reactions and managed to connect with them on somewhat of a deeper level, seeing them as more than mere cans of steel with wires, thus at least partially fulfilling the goal of giving the robots the ability to express genuine emotions.

5.2.5 Videos (References)

The videos of the robots were uploaded to YouTube and set to the "unlisted" status, logging in however, is not required to view them. Below is a list of all the videos used in the survey:

- "Macarena" Dance [38]
- "Floss" Dance [39]
- "Cheer" Emote [40]
- "Clap" Emote [41]
- "Dab" Emote [42]
- "Facepalm" Emote [43]
- "Shake Head" Emote [44]
- "Stand for National Anthem" Emote [45]

Chapter 6

Conclusion and Future Plans

With all the robots' reactions implemented, and the problems these possessed figured out, the main goal of this thesis is partially fulfilled. The robots are able to at least somehow express their emotional stages through various reactions and perform these dynamically, without shaking or tripping and falling, and the audience can more or less tell what the robots are up to.

A lot of work still needs to be done if one wants the robots to actually act human-like. The parameterizeable key-frame motions with the randomness based dynamism are but a starting point for creating dynamic motions. Where currently a lot of options regarding the robots' motions are still hard-coded within configuration files, such as the duration motions take to execute, it would greatly improve the dynamism if these were assignable during runtime. Further dynamism can be introduced by giving the robots the ability of automatically determining how to move their limbs using neural networks, rather than relying merely on pre-configured joint angles.

While making humanoid robots act human-like is a very hard task, giving them some emotional reactions is a further step in the right direction. How authentic the implemented reactions appear during a soccer match and what other problems these reactions present remains to be seen. As seen in section 5.2.4 the reactions given to the robots do have a genuine appeal to them, they are however, in need of further improvement.

Whereas this thesis merely showcased the implemented motions and the mechanics behind them, the follow-up thesis by Mr. Phillip Dolata 1.2 will determine how genuine these reactions look in-game. As mentioned at the start of this thesis, robot soccer games are already quite fun to watch, they miss the interactive part though, since the robots are not programmed to engage with the audience in any way.

If the robots are to indeed play against real human soccer players by 2050, there is still a lot of work to be done. The reactions presented in this thesis are merely the tip of the iceberg when it comes to human-like behaviour. For the robots to truly act human like one would need to give them some way of being somewhat creative like human beings, such that they could make up their own emotional reactions on-the-fly, rather than merely playing back pre-recorded reactions (even if these are dynamic). Furthermore, in a game against human players, robots should be able to engage emotionally with their opponents, such that these might sympathize with them instead of seeing them as mere nuts and bolts with microchips.

For instance, adding reactions where robots high-five their teammates, their developers or even audience members would greatly affect the way the audience perceives a soccer match. Today, all the spectators do is watch while the robots kick the ball back and forth until the game is over. If however, the robots engaged with the audience instead, it would greatly change robot football games. The soccer matches would change from being a mere one-sided static event to being an interactive one. This could be achieved by making the robots turn towards the audience and doing some gestures to which the audience reacts by joining in (similar to real sports, where an athlete might prompt the audience to do a "wave") or the audience cheering for the robots to

which the robots react happily.
Robots could also be given the ability of developing inter-team interactions that are unique to their RoboCup team using emergence algorithms (algorithms that use simple rules to create extremely complex behaviours). They could cheer or dance together for joy after a soccer match, react to comments from the audience, or simply interact with their developers when errors occur or their batteries are too low on power. The possibilities are endless.
Regardless of what features are to be implemented in the future, the goal of this thesis was merely to showcase a fraction of the human-like behaviour humanoid robots are able to perform. How these reactions evolve from here, remains to be seen. Who knows, perhaps in 2050 a robot football game might seem just as authentic as a human one. Only time, will tell.

Bibliography

[1] J. Kober, M. Glisson, and M. Mistry, “Playing catch and juggling with a humanoid robot,” in *2012 12th IEEE-RAS International Conference on Humanoid Robots (Humanoids 2012)*, 2012, pp. 875–881.

[2] “Aldebaran becomes Softbank Robotics | Softbank Robotics,” last accessed: 22. Febuary, 2022. [Online]. Available: https://www.softbankrobotics.com/emea/en/node/72

[3] “Company Profile | SoftBank Robotics,” last accessed: 26. January, 2022. [Online]. Available: https://www.softbankrobotics.com/corp/profile/

[4] “Pepper the humanoid and programmable robot | SoftBank Robotics,” last accessed: 05. March, 2022. [Online]. Available: https://www.softbankrobotics.com/emea/en/pepper

[5] “Whiz | Commercial Robot Vacuum | SoftBank Robotics America,” last accessed: 05. March, 2022. [Online]. Available: https://us.softbankrobotics.com/whiz

[6] “NAO the humanoid and programmable robot | SoftBank Robotics,” last accessed: 26. January, 2022. [Online]. Available: https://www.softbankrobotics.com/emea/en/nao

[7] D. Gouaillier, V. Hugel, P. Blazevic, C. Kilner, J. Monceaux, P. Lafourcade, B. Marnier, J. Serre, and B. Maisonnier, “Mechatronic design of nao humanoid,” 06 2009, pp. 769 – 774.

[8] “Joints — Aldebaran 2.1.4.13 documentation,” last accessed: 22. Febuary, 2022. [Online]. Available: http://doc.aldebaran.com/2-1/family/robots/joints_robot.html

[9] “Leds — Aldebaran 2.1.4.13 documentation,” last accessed: 22. Febuary, 2022. [Online]. Available: http://doc.aldebaran.com/2-1/family/robots/leds_robot.html

[10] “A Brief History of RoboCup,” last accessed: 26. January, 2022. [Online]. Available: https://www.robocup.org/a_brief_history_of_robocup

[11] H. Kitano, M. Asada, Y. Kuniyoshi, I. Noda, and E. Osawa, “RoboCup: the Robot World Cup Initiative,” *Proceedings of the International Conference on Autonomous Agents*, 04 1998.

[12] A. Ferrein and G. Steinbauer, “20 Years of RoboCup,” *KI - Künstliche Intelligenz*, vol. 30, no. 0, pp. 225–232, 2016.

[13] “RoboCup Simulation League,” RoboCup, last accessed: 03. March, 2022. [Online]. Available: https://www.robocup.org/leagues/23

[14] “RoboCup Small-Size League,” RoboCup, last accessed: 03. March, 2022. [Online]. Available: https://ssl.robocup.org/

[15] “RoboCup Middle-Size League,” RoboCup, last accessed: 03. March, 2022. [Online]. Available: https://msl.robocup.org/

[16] "RoboCup Humanoid League," RoboCup, last accessed: 03. March, 2022. [Online]. Available: https://www.robocup.org/leagues/29

[17] "RoboCup Rescue League," RoboCup, last accessed: 03. March, 2022. [Online]. Available: https://www.robocup.org/domains/2

[18] "RoboCup Virtual Rescue League," RoboCup, last accessed: 03. March, 2022. [Online]. Available: https://www.robocup.org/leagues/11

[19] "RoboCup@Home League," RoboCup, last accessed: 03. March, 2022. [Online]. Available: https://www.robocup.org/domains/3

[20] "RoboCup Industrial League," RoboCup, last accessed: 03. March, 2022. [Online]. Available: https://www.robocup.org/leagues/17

[21] "RoboCup Junior League OnStage," RoboCup, last accessed: 03. March, 2022. [Online]. Available: https://www.robocup.org/leagues/19

[22] "Sony's Aibo robotic pet is getting a new lease on life," TechSpot, article by: Shawn Knight, published: October 9th, 2017, last accessed: 08. March, 2022. [Online]. Available: https://www.techspot.com/news/71316-sony-aibo-robotic-pet-getting-new-lease-life.html

[23] "RoboCup Standard Platform League (NAO) Rule Book," May 2019, last accessed: 26. January, 2022. [Online]. Available: https://spl.robocup.org/wp-content/uploads/downloads/Rules2019.pdf

[24] "The B-Human Competition Field; B-Human's full-size field during the GORE and the RoboCup 2021," Lukas Plecher, Licensed under CC BY-SA 4.0, last accessed: 01. March, 2022. [Online]. Available: https://b-log.b-human.de/galleries/virtual-robocup-season-2021/IMG_3759.html

[25] "B-Human | About B-Human," last accessed: 26. January, 2022. [Online]. Available: https://b-human.de/index.html

[26] "B-Human Robot; Close-up of a NAO robot wearing B-Human's black default jersey," Annemarie Popp, DFKI GmbH, Licensed under CC BY-NC-ND 4.0, last accessed: 26. January, 2022. [Online]. Available: https://b-log.b-human.de/galleries/b-human-team-and-robots-2019/20190215_B-Human_ap-0343.html

[27] "$8000 Robot Unboxing!" uploaded to YouTube by Unbox Therapy, 23. May, 2015, last accessed: 22. Febuary, 2022. [Online]. Available: https://www.youtube.com/watch?v=OznNHfJ9P_k

[28] "RoboCup 2019 SPL Final - HTWK vs. B-Human," published to YouTube by HWTK Robots on July 7th, 2019, last accessed: 22. Febuary, 2022. [Online]. Available: https://www.youtube.com/watch?v=4_BWQl91p9Y

[29] "Google Forms by Google," last accessed: 13. March, 2022. [Online]. Available: https://www.google.com/forms/about/

[30] "Red Ring of Death | Xbox Wiki | Fandom," last accessed: 04. March, 2022. [Online]. Available: https://xbox.fandom.com/wiki/Red_Ring_of_Death

[31] "Fortnite | Homepage, Epic Games Inc, Fortnite store page," last accessed: 05. March, 2022. [Online]. Available: https://www.epicgames.com/fortnite/en-US/home

[32] "Macarena - Original Version by Los Del Rio," uploaded to YouTube by Antoine Niagne, 30. January, 2017, last accessed: 05. March, 2022. [Online]. Available: https://www.youtube.com/watch?v=uZW4vbIstoo

[33] "How to floss dance, who created it and why is everyone doing it?" Metro News UK, Article by: Olivia Waring, published: April 18th, 2018, last accessed: 05. March, 2022. [Online]. Available: https://metro.co.uk/2018/04/18/floss-dance-created-everyone-7476359/

[34] "Dabbing | Merriam-Webster," last accessed: 05. March, 2022. [Online]. Available: https://www.merriam-webster.com/words-at-play/what-is-dabbing

[35] "Usain Bolt's "Lightning Bolt" celebration doesn't mean what you think it does | The Daily Telegraph, Article by: Brett Smiley, published: August 16th, 2016," last accessed: 05. March, 2022. [Online]. Available: https://www.dailytelegraph.com.au/sport/olympics-2016/usain-bolts-lightning-bolt-celebration-doesnt-mean-what-you-think-it-does/news-story/628cb62df49776ef07b034b21b27c3a1

[36] "Learn how to dance Macarena," published on Pinterest Russia via JuicyCanvas, last accessed: 06. March, 2022. [Online]. Available: https://www.pinterest.ru/pin/274508539770471323/

[37] "Floss Dance Sprite Sheet," licensed under Public Domain, free for personal use, uploaded on NicePNG, last accessed: 06. March, 2022. [Online]. Available: https://www.nicepng.com/ourpic/u2t4t4w7t4y3i1y3_big-image-floss-dance-sprite-sheet/

[38] "Robot Dancing Macarena," uploaded to YouTube by Leonard Haddad, February 17th, 2022, last accessed: 13. March, 2022. [Online]. Available: https://www.youtube.com/watch?v=zWgDolhWsEk

[39] "Robot Flossing," uploaded to YouTube by Leonard Haddad, February 17th, 2022, last accessed: 13. March, 2022. [Online]. Available: https://www.youtube.com/watch?v=-_vlZ9nGPxQ

[40] "Robot Cheering," uploaded to YouTube by Leonard Haddad, February 17th, 2022, last accessed: 13. March, 2022. [Online]. Available: https://www.youtube.com/watch?v=d9McmzbOyaA

[41] "Robot Clapping," uploaded to YouTube by Leonard Haddad, February 17th, 2022, last accessed: 13. March, 2022. [Online]. Available: https://www.youtube.com/watch?v=adCF4C8HlTs

[42] "Robot Dabbing," uploaded to YouTube by Leonard Haddad, February 17th, 2022, last accessed: 13. March, 2022. [Online]. Available: https://www.youtube.com/watch?v=NRGpaM_M4x8

[43] "Robot Facepalming," uploaded to YouTube by Leonard Haddad, February 17th, 2022, last accessed: 13. March, 2022. [Online]. Available: https://www.youtube.com/watch?v=QJQhpZUD9jg

[44] "Robot Shaking Head," uploaded to YouTube by Leonard Haddad, February 17th, 2022, last accessed: 13. March, 2022. [Online]. Available: https://www.youtube.com/watch?v=C-O1FXAt7Ek

[45] "Robot Standing for National Anthem," uploaded to YouTube by Leonard Haddad, February 17th, 2022, last accessed: 13. March, 2022. [Online]. Available: https://www.youtube.com/watch?v=yvMbB3C7fME