Aalborg University Copenhagen

Tangible User Interface for Exploring Expressivity in Semi Autonomous Robots

Authors:
Jannik Jepsen & Niels Peter Rasmussen

Supervisor:
Daniel Overholt

May 2014

AAU
Medialogy CPH

Abstract

Tangible User Interface for Exploring Expressivity in Semi Autonomous Robots

by Jannik Jepsen & Niels Peter Rasmussen
M.Sc. Medialogy Thesis

This project aims to explore the use of point lights and movement in autonomous robots to effectively communicate and express the internal state of the system. The project uses a tangible user interface platform to model contextual scenarios. Different expressive models are designed and evaluated iteratively. It is the idea that the use of expressivity in autonomous robots will allow robots to integrate better in a human robot interaction. The project describes the development of an autonomous robot with motor and light actuators used to prototype three different behaviours. The behaviours are evaluated by 13 test participants. The conclusion of this study rejects the null-hypothesis, and provides information of how gestures in robots may be more effective than light behaviours.

Preface

This report is a documentation of the Master Thesis for Jannik Jepsen and Niels Peter Rasmussen during the spring of 2014.

In the back of the report there is a CD including video material, and a digital copy of this report.

Contents

Chapter 1

Introduction

Within the field of human computer interaction there is an increasing focus on robotics, as these are becoming more common in productivity, safety, learning and consumer products. Within computers and mobile phones, graphical user interfaces and more recently touch and gestures have become second nature to end-users. It is believed that robots will undergo the same development, where they will move from advanced machines, designed and used by engineers and specialists to a more user friendly domain where people will interact with them in everyday situations. A common example of this development is the iRobot Roomba™which is now known and used in many homes. This development has led to believe there is much work that can be done within understanding robots better, by improving the way robots can express their actions and intentions.

Together with the increasing accessibility of prototyping tools such as 3D printing and lasercutting technologies, the decreasing size and price of development platforms such as Arduino-compatible microcontrollers, the goal of this project is to develop a semi-autonomous robot that will be used to prototype expressivity and robot behaviours that users can effectively understand. Semi-autonomous robots are most of the time able to work independently, but they do still require some instructions from users. If robots were able to effectively express their actions and intended actions it might improve the relationssship and user experience. Given a better understanding of a robots' behaviour and underlying intelligence could hopefully improve users ability to correspondingly make more informed decisions in the interaction with the robot.

This report will describe the process of developing a tangible interface that allows exploration of simple autonomous behaviours in robots. The idea is to explore how to design an interaction language between the user and the robots based on locomotion and visual modalities.

Chapter 2

Preliminary Analysis

In the design of human robot interaction, there are several aspects to consider in order to understand how the user interacts with interfaces and how this interaction affects the user. This chapter will be covering the general approaches to designing human computer interactions with conceptualised models from interaction of objects in general, and also in the human computer context. Furthermore specific approaches to human robot interaction, which can be seen as a subset of human computer interaction, will be covered to understand robotics in general and be able to clearly describe why robot behaviours and expressivity are relevant in human robot interactions. The main areas of interest in the understanding of human robot interaction is to begin with an insight in the current use of robots and different types of robots for which this concept of expressivity would be relevant. The overall relevance will be described from a human computer interaction perspective, where approaches to developing machines become user centered with focus on how we experience the state of the machine and how well that information integrate into our existing knowledge of the world and our expectations.

Much of the research that go into designing robot expressivity and behaviours draws from other fields of study such as personality models from psychology and more specific HCI related studies regarding expressivity and the use of motors to model behaviours. These will briefly be presented along with contemporary uses of autonomous robots in the home.

2.1 Human Robot Interaction

As the agenda of this report is to explore human robot interaction (HRI) in autonomous robots, it must be clarified how both HRI and the level of autonomy is defined for this project. Robots are many things and in robotics research there are many different types of robots, such as *industrial robot arms* which are highly precise systems that can complete trivial productivity tasks, *humanoids* which typically have highly complex intelligence and behavioural systems for moving and interpreting the world around them, *social robots* where focus typically is on the interpretation of human expressions and the ability to respond to humans in a social context and also simpler *domestic robots* such as the iRobot Roomba which in essence can move around on a floor and only detect walls and cliffs[1].

Much of the research in autonomy and operation in HRI is focusing on search and rescue-, military- and space exploration robots. These are all examples of robots which utilises different levels of autonomy. Some of these are fully autonomous robotic systems with highly developed artificial intelligence models and complex communicative abilities.

This project will be focusing on domestic and utility robots with a lower level of autonomy and limited abilities to express itself motorically or visually. Typically these domestic and utility robots are designed as semi-autonomous robotic systems. The semi autonomous robots are defined as robots that the human controls at times dependant on tasks. Within autonomous robotic systems, human interactions can be divided into to two types of control; *supervisory-/ supervised control* and *shared control*[2][3]. In *supervisory control* the human may assign some tasks to the robot to complete autonomously and only observe the progress, this is often times used in remote operation control situations. In *shared control* the human assigns tasks to the robot but may intervene with input such as perception, additional instructions or to cancel execution. Shared control is what you would also experience with the iRobot Roomba™, if you at some point decide to change the way it is vacumming or turn it off by clicking the button on top of it. Shared control can also be described as a ratio of intervention vs. autonomy[3]. As an example Yanco describes a project called Wheelesley, where low level tasks such as path centering and obstacle avoidance were autonomously controlled and the user were responsible for high level task. This division of control would be classified as autonomy=75% and intervention=25%[3].

Whether or not to control or interrupt a robot depends on the human perception of what the robot is currently doing or about to do. It is the human interpretation of the robot that defines whether we as humans should take over control and alter the instructions for a robot. Efficient communication of internal states of the robot should make the decision of taking over control more intuitive. Furthermore if the non-verbal HRI would be extensive enough, the communication could develop beyond just functionality and action driven communication.

2.2 Human Computer Interaction

In this project a prototype of a semi-autonomous robot will be proposed which meets the criteria for simulating autonomous behaviour in domestic and service robots. The design of this interaction will be approached from a human computer interaction (HCI) perspective, with its focus on expressivity and the physical space in which the robot exists. To be able to assess and evaluate the interaction scenario between robot and human, this project will draw on concepts from HCI with a main focus on usability testing.

Evaluation of the interaction design can generally be assessed in two ways, one being usability as the quantitative measure to evaluate the performance of a design and the other being the user experience, a qualitative measure to evaluate how the interaction feels to the user [4]. Both measures are to ensure the best possible interaction model in the specific context. Generally good interaction design can be defined as *"Designing interactive products to support the way people communicate and interact in their everyday and working lives.*[4] In this project the interaction scenario will be assessed in terms of the level of interaction and how it supports the communication between human and robot.

Usability engineer Donald Norman with a background in cognitive science has authored the 7 steps of action [5], which describes how people approach the task of completing arbitrary goals. These steps are described as a general model for all interactions and are not specific to HCI. The steps consist of three overall stages that go into the interaction being one overall stage of execution relating to the *goal* a user wishes to accomplish, three stages of *evaluation* and three stages of *execution* [5].

These steps are not necessarily always possible to measure accurately, but they do however evaluate interactions objectively according to how users perceive the interactive object and how it relates to how a user would expect to interact with the object to complete a given task.

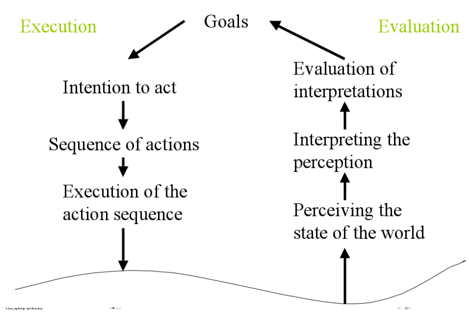


FIGURE 2.1: Norman - 7 Stages of Action

The seven stages of action assess two issues of interaction: *the gulf of execution* and *the gulf of evaluation*. The gulf of execution is when the possible interaction of the system does not match a user's intentions. The gulf of evaluation is when the systems representation does not match the user's expectations[5].

The model is also used in HRI design to imitate the human action selection model[2]. As robots also need to perceive the world around them and be able to act accordingly it must be able to interpret the world, act in the world and evaluate its actions. The action selection can occur in several layers of interaction, some of these will be described in more detail in the development of action selection models.

Changes in HCI

HCI has traditionally been about usability, however it is changing towards taking user experience into consideration as there a studies showing positive correlations between usability and aesthetics[4]. Contrary to usability the aesthetics, more precisely the experience, requires an understanding of what the user felt or experienced throughout the interaction. These observations could be acquired through ethnographic studies, interviews or questionnaires. Examples of what the user experience is, or could be identified as, is a long list of emotional reactions. The experience is a complex thing to measure as it could be comprised of some of the below elements.

- Satisfying
- Enjoyable
- Engaging
- Helpful
- Cognitively stimulating
- Enhancing sociability
- Fun
- Aesthetically pleasing
- Challenging

2.3 Applying HCI to HRI

While HRI for a long time have been led by engineers, social scientists and designers it seems that the increasing introduction of robots in consumer markets is leading to an increased focus on the HCI aspects of designing HRI[6].

When it is important to distinguish between HCI in general an HCI in relation to HRI, it is because there are several distinct differences in how human perceive autonomous behaviours in robots. According to Sara Kiesler and Pamela Hinds there are three reasons autonomous robots are distinct cases[7].

Firstly it is argued that people tend to personify autonomous robots more than they would other computer systems, among other reasons, due to how we perceive autonomous movement. *Secondly* autonomous robots often exist in the same physical space as us humans and therefore must have some awareness of it surroundings and be able to interact with it. *Thirdly* and lastly autonomous robots are to some extent able to process information and learn more about themselves and their environment over time. [7]

Aaron Powers - the lead HRI researcher at iRobot - describes how HCI is increasingly being applied in their development processes as their robots are moving from research projects towards commercial products. This transition from a technology-centered approach towards a user-centered approach has also led the company to define a list of HCI principles they aim towards applying to their HRI [6].

KEY PRINCIPLES BEING APPLIED TO HRI	SOURCES
Required information should be present and clear	Sholtz, Nielsen, Schneiderman (modified)
Prevent errors if possible, if not, help users diagnose and recover	Nielsen, Schneiderman
Use metaphors and language the users already know	Sholtz, Nielsen (modified)
Make it efficient to use	Sholtz, Nielsen, Schneiderman
Design should be aesthetic and minimalist	Jenson [5], Nielsen
Make the architecture scalable and support evolution of platforms	Sholtz
Simplify tasks through autonomy	(new)
Allow precise control	(new)
Create a positive brand image	(new)
Strive for a natural human-human interaction	(new)

FIGURE 2.2: Key principles being applied to HRI by iRobot

The first six key elements originate from traditional HCI research, and will also be considered in the relationship between HRI and HCI. However there are three new key principles that will be in focus when working with autonomous robots and their expression. The key principles are: *simplify task through autonomy*, *allow precise control* and *strive for natural human-human interaction.* The remaining principle of creating a positive brand image is understandable in a product company, but will not be considered in this project.

It is with a basis in the idea of using HCI principles and reasonings in the design of HRI this project aims to explore how an interaction language for autonomous behaviour in robots can be designed using a tangible user interface.

2.4 Tangible User Interaction

Robots are essentially physical embodiment of data and transforms computational objects into the physical space. As described previously this is one of the key characteristic of robots that distinguish them from traditional computational systems. To successfully approach expressivity in robots within the physical space, a tangible user interface (TUI) can be used as a development platform. TUI's encompasses natural interactions with its richness of tangibility, movement and direct manipulation. TUI's are relatively new but are increasingly popular in areas such as HCI, computing, product design, robotics and the interactive arts.[8]

TUI is formed from the many of the same principles as HCI, and especially evolved through the 90's from the concept of graspable interfaces, manipulating wooden blocks, to interact with the digital domain. TUIs are applied in many different applications and research areas. Through the following section, an overview and examples of these application domains and research fields will be summarized from the monograph "Tangible User Interfaces: Past, Present, and Future Directions" [9] [Jannik: Duplicate reference]

TUIs for Learning

> *"...research and theory on learning stresses the role of embodiment, physical movement, and multimodal interaction[9]. "*

TUIs naturally accompanies the important aspects of learning applications and is therefore a very suitable interaction method for learning. The area is extensively researched throughout the past years and is applied in many applications especially aimed for children where learning combined with entertainment is the focus for many educational toys and museum installations. Many of the learning applications overlap with some of the upcoming described research areas, such as problem solving and programming with TUIs.

Problem solving and Planning

TUIs have shown effective in supporting problem solving, especially three aspects: *epistemic actions*, *physical constraints* and *tangible representations*. These aspects of TUI is the most effective in problem solving situations due to their tangibility.

Epistemic actions, are the manipulation of objects performed by the user while understanding the problem and the context of the objects and task at hand. Epistemic actions are a wide range of manipulation such as rotating in space or rearranging objects to see objects in different contexts.

Physical constraints make use of the natural constraints in the objects form or the solutions space, to limit the use of explicit syntax and explanatory rules. This limits the need for cognitive use of learning compatibility of objects and their rules.

Tangible representations utilizes the physical representation of the problem to its full extend and is most compelling with the spatial or geometric application domain. E.g.direct manipulation of a representations of an architectural design og urban planning model. This can support the designers by using direct manipulation which reduced the cognitive load and enables more creative immersion.

Tangible Programming

The use of tangible interaction for creating computer programs has been among the first TUI research and applications, with tangible objects used to create combinations of physical algorithmic structures controlling a digital domain. Tangible programming has especially been found useful for educating elementary programming techniques and advanced problem solving for young children. Some of the techniques used for programming with tangible interfaces can be classified as *constructive assemblies*, where by combining different physically constrained objects in a connected modular system creates a form of physical syntax programming language. Furthermore *programming by demonstration* is similarly used as a tangible interaction method for instructing different motions and gestures, especially for toys, which is rehearsed and repeated by the system. The use of tangible objects and especially robots in programming has existed for some time through Lego Mindstorms™ [1], Curlybot [2] and Topobo [3].

Entertainment, Play and Edutainment

TUI is a central part of entertainment and is used throughout most toys, especially modern toys which employs the physical interaction with the digital domain. Besides the elementary effect of entertainment, TUI also employ a wide range of edutainment often used through museum installations for educating and awareness of specific topics through the use of embodiment and entertainment. Some of the more recent examples that employ robotics systems in combination with entertainment and play are ANKI Drive™[4] and Ozobot™[5].

[1] http://www.lego.com/en-us/mindstorms/
[2] http://tangible.media.mit.edu/project/curlybot/
[3] http://www.topobo.com/
[4] http://anki.com
[5] http://ozobot.com

Social Communication

Embodiment is natural aspect of human communication, which is why TUIs can contribute to social communication between persons, by utilizing physical tokens as part of communication through the digital domain. A common goal is achieving remote intimacy, by experimenting with different modalities for communicating e.g. body language or as simple as another person's heart rate to give a sense of intimacy.

2.4.1 Robots as TUI

As it has already been mentioned, the TUI platform is especially interesting when working with robots particularly when developing and exploring interactions with physical objects that have movements and spacial presence. As defined by Fitzmaurice the TUI provides a "physical handle to a virtual function where the physical handle serves as a dedicated functional manipulator"[10].

The physical handle gives users "concurrent access to multiple, specialized input devices which can serve as dedicated physical interface widgets"[11]. This access affords physical manipulation and also encourages spatial arrangement. The list below is a summary of the properties of a TUI:

- Space-multiplexing,
- Concurrent access and manipulation (often involving two- handed interaction),
- Use of strong-specific devices (instead of weak-general, that is generic and non-iconic),
- Spatial awareness of the devices, and
- Spatial reconfigurability

[11]

Use of TUIs in development of tangibility and embodiment of digital system with the above properties - robotics as a specific example - is supported by several studies, as it opens up to a wide variety of interactions. Some of these interactions are *collaboration* [12][13], *natural interactions* that undermines the need for specific input devices, and *focuses on objects* [12][1][10] and generally adding to the sensory experience[14][13][15].

2.5 Related Work

As robots enter homes of consumers, research and development within HRI are moving towards the social aspects of robotics and how people tend to attribute robots with personality based on movement and appearance[16]. The idea that humans induce inanimate objects with being life-like and having a personality based on movement and patterns was demonstrated in 1944 with movements of simple dots in a 2-dimensional space[16]. Meerbeek et al. uses a personality model called the Five-Factor Model of personality to investigate desirable personality traits in robots. These traits are categorised under the Five-Factor Model. The desirable *neurotic* behaviours were "calm", "not easily upset" and "relaxed". Classified under *extravertion* characteristics, the desired traits were "introverted", "reserved" and "withdrawn". The desired traits classified as *openness to experience* was "likes routines". In *agreeableness* users desired "cooperative-", "distant", "friendly" and "polite" characterics. Lastly the desirable characteristics classified as *conscientiousness* were "efficient", "serious" and "systematic"[17].

Anthropomorphism can also be seen in robotics, and research has shown that people tend to attribute personality to robot vacuum cleaners such as ascribing them a name, gender or other personality characteristics[17]. With the Roomba as an example, research has shown that users describe its personality as *stubborn*, *silly*, *crazy*, *intelligent* and *dumb*[17].

One of the arguments for employing behaviours and expressivity in robotic systems, is that these autonomous systems are reaching a level of complexity where you would not expect users to be able to comprehend the control systems and underlying implemented sensor and actuating technology of a robot. Therefore robotics systems should portray a mental model that users can understand and accept in their home[16].

2.6 Commercial Application

Autonomous robots are already making their way into households through commercial products and a lot points towards an increase in the use of robotics in domestic applications such as household chores and toys[1]. This section will describe some of the most recent advances in the commercial application of robots in the household.

iRobot

iRobot™is a very relevant example of how robotic development is moving towards the home and domestic appliances. Coming from an idea of developing robots for space exploration in colaboration with NASA, focus has shifted towards commercial implementation of autonomous robots in a variety of domestic robots. iRobot™offers home robots for *vacuum cleaning*, *floor scrubbing*, *gutter cleaning* and *floor mopping*. iRobot™also develops robotic systems for defence and security. Especially the iRobot Roomba™has proven how robots can impact peoples every day lives. Other than being a popular home product the iRobot Roomba™and iRobot Create™has been used as an "out of the box" developmental platform for research within TUI[18], constructive assemblies[19], expressivity of personality[17] and artificial intelligence[20][21].

ANKI Drive

ANKI Drive™is a company that utilises robotics, autonomy and artificial intelligence in toy cars.

> *"The company was founded in 2010 by Carnegie Mellon Robotics Institute graduates who sought to create new consumer experiences using technology that was once confined to robotics labs and research institutes.*[6] *"*

Many of the features in this car racing game have until ANKI Drive been standard features in console and PC games for the screen. ANKI Drive bear witness of a transformation of existing computer systems that move into the physical world. The racing game's artificial intelligence (AI) is designed to express different behaviours through little robotic cars, that a user will be able to compete against. The cars can express different levels of *defensive* or *aggressive* behaviours through the way they move on the race track. These behaviours are used so players can adjust their own actions and better be able to compete with the AI cars.

Play-I

Play-I is another company who have introduced robots in toys. The products Bo & Yana are programmable robots aimed at kids. They combine the fun of playing with robots and the educational element of learning to program robots. The robots can be programmed through different levels of visual and easy-understandable programming languages. The programs allows children to experience how robots can use actuators and sensors to sense and move around in the world. The physical appearance of the robots are developed

[6] http://anki.com/in-the-news

around the personality of the two different robots, where one is described as *"Bo is an explorer. Bo is playful and curious. Bo loves going on adventures and making new friends. As you play together, Bo learns new skills and becomes a more capable robot. Together, there's no stopping where you and Bo can go."* and *"Yana is a storyteller. Yana is clever, imaginative and full of dreams. Yana can surprise and entertain you by bringing characters to life as you play! Use the power of your imagination to unlock Yana's potential."*[7]. Play-I uses personality in robots to engage children in learning tasks such as programming. By designing a personality around a robot the interaction would be expected to move closer to a human-human interaction than traditional programming is.

2.7 Delimitation

In the previous chapters the research areas for this project has been introduced with common theories and practises related to each area. This sections aims to specifically describe how they will be used in this project. Furthermore these practises will be illustrated to clearly define how they each relate and the impact they each have on the project evaluation.

In this project robots should be understood as mobile robots that can navigate on a plane surface with the use of predefined behaviours. The expressive means are limited to visual and locomotive output as these are commonly seen in domestic robotic systems. These behaviours are interchangeable and can be manipulated according to user interaction with the robot. As it has previously been described, this type of robot can be referred to as a semi-autonomous robot with shared control.

The user interaction to instruct these behaviours in a robot will be applied through the use of tangible user interaction. Although graphical user interfaces and other social interfaces such as speech and face detection are being used widely in social robotics - and do inherently contain the possibility of complex expressivity, this project will aim to investigate the physical space in which the robots exists and design an interaction around the tangibility of the robot and other objects that surround the robot with focus on its movement and lights.

The evaluation of how well the interaction is designed will be evaluated according to traditional HCI criteria such as usability and user experience. The efficiency of the interaction should be quantifiable and as efficient as possible in terms of the time it takes to learn and use the interface. Moreover the interface should - when learned - also

[7]https://www.play-i.com/#our_robots

be efficient in the time it takes to convert intentions to actions. To be able to identify the factors that influence the overall user interaction this project will be using Donald Norman's 7 steps of action in forming an evaluation method.

2.8 Problem Statement

The overall goal of this project is to enhance the expressitivity of domestic autonomous robots, as it is believed to allow users to make more informed decision when interacting with robotic systems. To do this we wish to answer the following question:

How can a tangible user interface in human robot interaction be utilized to effeciently express robot actions/intentions to the user?

Chapter 3

Method

This chapter outlines how the problem is approached as a research topic, it will describe the iterative approach of prototyping a tangible interface for exploring robot expressivity and the emperical method for evaluating such prototypes against the problem. Each iteration contains a more specific description of the evaluation design.

3.1 Iteration structure

This project contains two iterations which approaches the problem in a similar manner, consisting of an *analysis* of specific topics for the iteration, a deduced *design* of a proposed solution approach, a *technical documentation* of the implemented tools to cover the proposed design solution and an *evaluation* of the solution.

Analysis

An analysis of existing related research and products, which serves as a basis for desig descisions and assumptions made throughout the development of the proposed solution.

Design

The project utilizes a prototyping approach of designing and developing continuously while assessing the effects and redesigning throughout the ongoing process of estimating a possible solution. A lot of the design process has been practical adjustments and improvements which are both technical considerations and have been continuously evaluated without user participation. These incremental improvements have not all been documented, but will be reflected in the final designs. Each iteration therefore contains a description of the final design solution, its aspects and important features.

Technical Documentation
The technical documentation for each iteration describes all the elements of the system developed to sucessfully cover the proposed solution design.

3.2 Evaluation

Each iteration evaluates the solution against the problem with a main focus of usability and effect by an emperical data collection and analysis of such. The individual iteration describes the important aspects of the corresponding evaluation towards that solution, mainly the procedure and goal.

Subjects
Subjects are invited to participate in the study through either personal contact or a sign-up form sent via social media and e-mail to all students of Aalborg University Copenhagen(see appendix B). The selection of subjects is ensured to have no prior knowledge of the problem area. The solution and the goal of the study to avoid any bias from the subjects. Beside no knowledge of the study the subjects are not required to have any specific prerequisites for participating in the study. The number of subjects for the study is based on Jakob Nielsens magic number 5 [22] used for usability testing, 5 participants will provide 80% of the usability insight, this hovever is for HCI usability testing of interaction application and not for statistical validity though it is considered a good basis when considering cost benefits of ongoing studies. Each iteration consists of a sufficient number subject, though it can always be discussed that more subjects will give more validity to the evaluation.

Procedure
Each iteration will cover the specific procedure for that specific evaluation, but will generally cover the method of evaluating the solution of the problem, what specifically the subjects will encounter throughout the test and if specified the duration of the test. Both iterations are performed in closed controlled laboratory conditions.

Data Collection
Both iterations are consisting of data collected from a questionnaire designed specifically to the iteration and data from the system itself such as performance and time. Quantitative measures from the system is inspired from general usability testing such as; *Time to complete a task*, *Number and type of errors per task*, *Number of errors per unit of time*, *Number of users making a particular error* [4]. Both iteration's questionnaire

are structured to include both open and close questions which will adress the subjects validity and opinions of the system. The open questions provides the subjects with the possibility to express themselves, but can lead to vague answers. The closed questions are likert scales based on the same structure throughout the questionnaire of a 5 point scale of two opposite statements.

Data analysis

Each iteration contains separate data analysis sections where the results from the study will be presented from an unbiased perspective. According to the iteration and the data it will be presented accordingly with a similar structure of showing likert scale scores as histogram of subject answers, open questioned shown in table form as sample from a complete list presented in appendix. Quantative data will be analysed through linear statistical methods such as, mean, deviation and t-test, and is always checked for any inconsistent data or major deviation in subjects for validity.

Findings

Each iteration will uncover what has been found through the evaluation, what the results means and their effect on the problem and proposed solution. The findings section will try to relate the results to any preliminary research and assumptions made for the iteration.

3.3 Discussion

Finally the discussion for each iteration will draw a bigger perspective on the findings from the evaluation and their use in further development.

Chapter 4

Iteration 1

In this iteration the process of developing a digital layer on top of the TUI will be described. Firstly there will be an introduction to some of the elements that make out a system that contains autonomy in digital agents. This involves a look into how artificial intelligence is being used in other digital applications such as games, and how some of these game behaviours can be used to also design a sense of personality. Furthermore it will provide a more detailed description of how TUI's can be used as interaction for problem solving to explore the relationship between expressivity and user interaction.

We will investigate popular uses of interaction in TUIs and describe how *modular constructive assembly* and *tokens and constraints* is used to inspire the first interaction with a digital agent.

4.1 Analysis

This chapter will describe some of the previous work that has been done in expressivity and robot interactions. The aim of this iteration is to be able to develop a system that can perform with a high level of precision and speed in order to develop better feedback in the end. The system will be evaluated according to how users perceive the performance of the system and will be presented at the end of this chapter.

In section 2.4 TUI's for programming have been introduced as having an effect on creating algorithmic structures for programming within the digital domain. Mainly two techniques have been described: *Constructive assemblies* and *programming by demonstration*

4.1.1 Game AI

To design a semi-autonomous robot that will be able to navigate in a natural environment, a certain level of intelligence must be implemented. Reynolds describes autonomy in relation to steering behaviours used in animation and games, as a hierarchical model of three different levels.[23]

At the *highest level* an agent must be able to achieve a certain goal by appropriately devising a strategy, planning according to its own abilities and then select a number of actions. It is the ability to choose between several actions and combine these that make out more complex behaviours and would allow an agent to navigate in a natural environment and appear more life like. The action selection can be done in a number of ways, where the combination of different behaviours can either be sequentially choosing one action at the time or in some case blending behaviours for instance by weighing each behaviour according to its priority. The *middle level* of the model is where all the individual steering behaviours are defined. These are all behaviours that each have a specific purpose and will return a direction in which the agent should move if chosen. Some of the important middle level locomotion behaviours mentioned by Reynold are:

- Seek
- Pursuit
- Evasion
- Obstacle avoidance
- Wander
- Path following
- Wall following
- Cohesion
- Alignment

At the *lowest level* of autonomy is the locomotion of the agent, which is the constraints of the agents body - physical and/or virtual. After the agent has - at the highest level - chosen a set of actions and combined behaviours - at the middle level - a set of steering commands will be communicated to the agent. The locomotion of the agent is the motor abilities of the agent and its physical limitation which in the end will affect how the higher and middle level behaviours are executed. For robots these limitations could for example be the weight of its body, speed and precision of electric motors and even constraints of the TUI.

4.1.2 Autonomy and Intervention

This hierarchical model of autonomy will be used in designing a semi-autonomous robot that will be able to take instructions from a user on the highest level, meaning that a user will be able to define a task or goal in the space of the TUI. The division of action selection can as described in section 2.1, be defined as a relation between autonomy and user intervention.

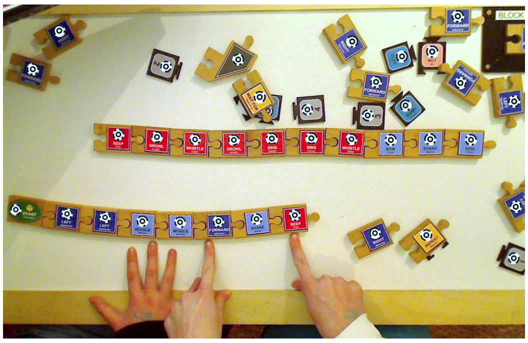
In HRI the spectrum in which autonomy is measured is a continuum between the level of autonomy against the amount of intervention from the user. A high level of user intervention is what is seen in remote controlled robots and require near constant intervention. An increasing level of autonomy would require less intervention[3]. The level of autonomy is measured in percentage of the time the digital agent is carrying out its tasks on its own.

This division of shared control is a static measure and is traditionally used in systems with predefined responsibilities. However Yanco also describes situations where changes in the level of autonomy could occur over time or in context specific situations where the digital agent would overrule certain decision or vice versa[3]. There are different approaches to the use of autonomy and how it is utilised. This project aim to investigate how autonomous systems can be used to express the internal states of a digital agent. It is the goal of this iteration to design an action selection system and a combination of behaviours that matches the expectation of the user.

4.1.3 Constructive Assemblies

Horn et al. [19] successfully implemented a TUI with constructive assembly for programming sequential robot behaviour in a museum installation. Designed for elementary and middle school children they enforced and evaluated upon five design consideration: *Inviting*, *Apprehendable*, *Engaging*, *Supportive of Group Interaction* and *Inexpensive and Reliable*. Making the installation apprehendable for the targeted users can be logically deduced to the most important factor as it effectively changes e.g. the engagement. Their goal was to make the interaction apprehendable for museum visitors with no prior experience to easily learn how to use the exhibit.

The system consists of a TUI in the form of small wooden jigsaw puzzle pieces used for assembling a sequence of different behaviours and logical operators which is executed on a physical robot. The interface consists of eight different types of puzzle pieces which enforces different commands such as, *start*, *motions (forward)*, *sounds*, *Complex behaviours (shake)* and *control statements (loops)*. The puzzle pieces affords natural physical constraints, constraining the user from unintended assembly. Each puzzle piece is labeled with a unique fiducial marker, for camera tracking, and a descriptive text. (see Figure 4.1A). Furthermore the interface includes a designated area for instantaneous execution of a single command by placing the puzzle piece on a RFID receiving plate, giving the user a clear understanding of its behaviour. The constructed program is executed on an iRobot Create™(see Figure 4.1B).



(A) Fiducial tracking of a constructive assembly sequence

(B) Museum installation utilizing constructive assembly for executing robot sequential behaviour

FIGURE 4.1: Horn et al. TUI Museum installation [19]

Tokens and Constraints

As opposed to modular constructive assemblies Shaer et al. proposes a less strict interaction paradigm with the introduction of Tokens and Constraints (TAC). TAC is described to consist of pyfos, Tokens and Constraints. Pyfo is the definition of what could generally be referred to as a physical object. However since pyfo is confined within a TUI context, the physical object is enhanced or associated with certain digital properties. A pyfo can be either a token, a constraint or both[11].

Tokens are graspable pyfos that allow a user to interact with the digital properties of the pyfo. Generally it is then the idea that the physical appearance of a token would imply or afford a certain use closely related to the digital properties it represents, hence using its physical presence to enhance the perception of its properties.

Unlike tokens constraints are not necessarily coupled with any digital properties. Constraints are used to guide the user to interact with the associated tokens by limiting it physically. Shaer et al. describes three ways in which the constraint limits the behaviour of a token.

1. The physical properties of a constraint should suggest how a user can manipulate the associated token and its properties. This can be done by defining the orientation, material or textures of the constraint.

2. The constraints are used to limit the space in which the tokens can be manipulated and confines the interaction space for the user. As an example it would restrain the user from placing tokens in the same place.

3. The third way constraints are used in TUI is to serve as a reference frame for the interaction space. This can be used to represent real environments by depicting everything with miniature models of the environment and thereby use proportions as spatial cues of the size of the interaction space. The reference frame of the constraints can either be numerical values or relative terms, but should enhance the understanding of the environment[11].

4.2 Design

The goal of the iteration is to explore the relation ship between autonomy and intervention in problem solving tasks by the use of a tangible interface. For which there is two major systems to be designed, an environment which encourages problem solving in an intuitive way and a digital agent with a level of autonomy which incorporates a solution estimation algorithm for the problem which can be expressed to the user upon intervention.

The system designed in this iteration is a simulation of a physical space. All objects such as digital agents, obstacles and interactive tokens are designed as a digital layer on top of the physical. Therefore this first iteration is designed around a virtual agent projected onto the surface of the TUI.

4.2.1 Environment

In order to encompass a communication between the user and the digital agent, the environment will encourage a collaboration to solve a set of game-like problems. By using the spatiality of the tangible interface the environment consists of the task of bringing the agent from point A to B through a maze of obstacles. To enforce the collaboration the environment includes different types of obstacles for the user to explore and familiarize by the use of the digital agents expressions.

Part of the process of developing an environment for user interaction with a digital agent involves the design of a local positioning system that will be used when simulating the physical space in which a digital agent should act. With the use of a local positioning system it will be possible to retrieve positions of the interactive objects which form the TUI. The platform will be able to use the positions of the interactive elements to project visual feedback back in these position in order to deliver visual feedback related to specific objects in the environment, and thereby unify input and output spaces. The spatial mapping of physical object with a visually projected layer of the digital space allows the user to naturally associate objects with their functions.

As previously mentioned in section 2.4 a tangible interface naturally affords problem solving tasks, therefore the environment is designed to encompass aspects of TUI which supports problem solving such as, epistemic actions, physical constraints and pyfos. The environment is designed for projection on a physical surface which limits any visual representation of elements to two dimensions, including the task, maze etc.

Levels

In order to create the interaction with the environment and the collaboration with the agent, the environment maze is separated over several levels changing the difficulty each time. The levels are designed to be simplistic and understandable for the user with no prior experience. The levels introduce new elements individually to ensure a continuous exploration and understanding of the environment elements.

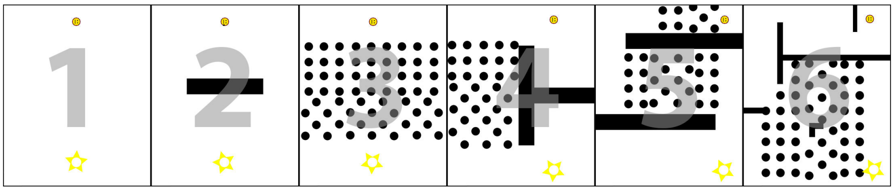
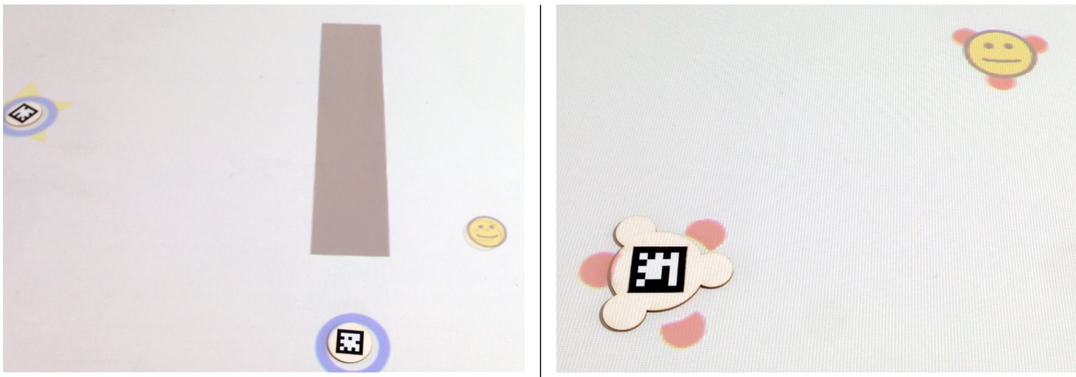


FIGURE 4.2: Level 1-6 - Continuously changing mazes with increasing difficulty

Waypoints

The user interaction consist in making the path for the agent through the maze, for which there is different waypoints, which the agent will use for guidance. The waypoints are physical objects which can be placed on the physical surface according to the virtual environment.

Throughout the levels the participant would be presented with two different obstacles: *Walls* and *Holes*. To navigate around these obstacles the waypoints would apply different abilities to the virtual agent. One would be a simple seek behaviour, getting the virtual agent to move straight to the waypoint allowing it to navigate around walls. The other would apply an avoidance behaviour, allowing it to move around holes. The objective for the participant is to correctly guide the agent to the goal. The path created by the user, is inspired by constructive assemblies used in TUI, but instead of putting the elements together in a jigsaw puzzle manner, the modules of the assembly utilizes the spatiality of the surface for the construction of the sequence.



(A) Seek Waypoint

(B) Avoid Waypoint

FIGURE 4.3: Waypoints

4.2.2 Digital Agent

This section will cover the design of a virtual agent which is presented graphically in the TUI to the user. This is the first step of this project towards employing autonomous behaviours in a digital agent. The digital agent system is designed to allow scalability towards a physical robot in the coming development.

The role of the digital agent in this system is to communicate an estimation of the user solution through a set of expressive behaviours. Through the use of AI algorithms the digital agent should be able to understand the environment presented on the TUI. The digital agent can be activated and follow user instructions and use its expressive features to communicate a level of confidence to what degree it assumes the instructions are correct.

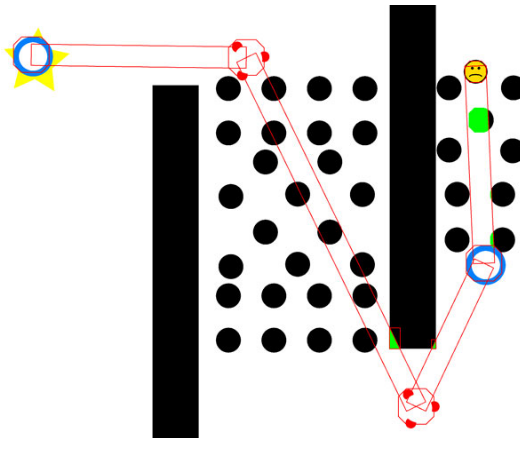


FIGURE 4.4: Confidence estimation visualisation.

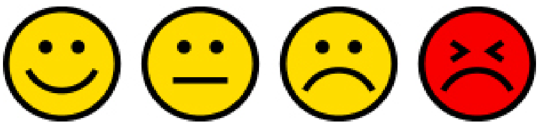


FIGURE 4.5: Smiley expression to estimate a problem solution.

The digital agent will utilise some of the AI behaviours described in section 4.1.1, and lay out a solution estimation through the use of *path finding*, *obstacle collision* and *wall collision* as it can be seen in figure 4.4. The estimation will result in a percentage of confidence that the digital agent on activation would successfully reach its goal. This level of confidence is evaluated each time the user intervenes, thus giving the user a better idea of when to activate the agent.

The main intelligence of the digital agent involves its ability to predict and estimate a solution which it is able to communicate to the user. However it does also have some behaviours as it moves through the environment. These behaviours are *seek* (see figure 4.6 and 4.3A) and *avoidance* (see figure 4.6 and 4.3B) and are triggered by the waypoints the user lays out on the TUI. These behaviours are applied to the digital agent as it moves between the user instructed path. The seek behaviour will instruct the digital agent to move towards the nearest waypoint laid out by the user in a direct line. The avoidance behaviour will allow it to move around obstacles on its way, by detecting objects in front of it and steer clear of them. These behaviours are also included in the solution estimation prior to activating the agent movement.

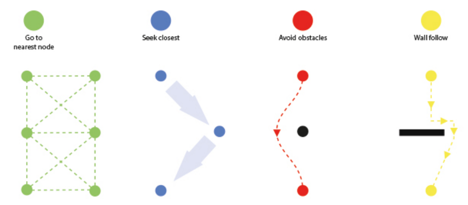


Figure 4.6: Game AI implemented in the digital agent.

4.2.3 Design Overview

In the end we have designed a complete interaction between the physical and virtual domains, where the user can position waypoints across the surface to find a solution to the mazes in coorporation with the digital agent.

As the user creates a path, the agent evaluates the path through the maze and provides and estimate of the solution, this gives the user two possible ways of solving the problem (see figure 4.7). Either they can completely rely on the agent and change the path until the agent is certain of success or they can let the agent run through the path and see where any problems may occur. The two approaches can be defined as either a trial and error approach or a collaboration. It is the goal to determine which approach the user used and whether they collaborated with the agent.

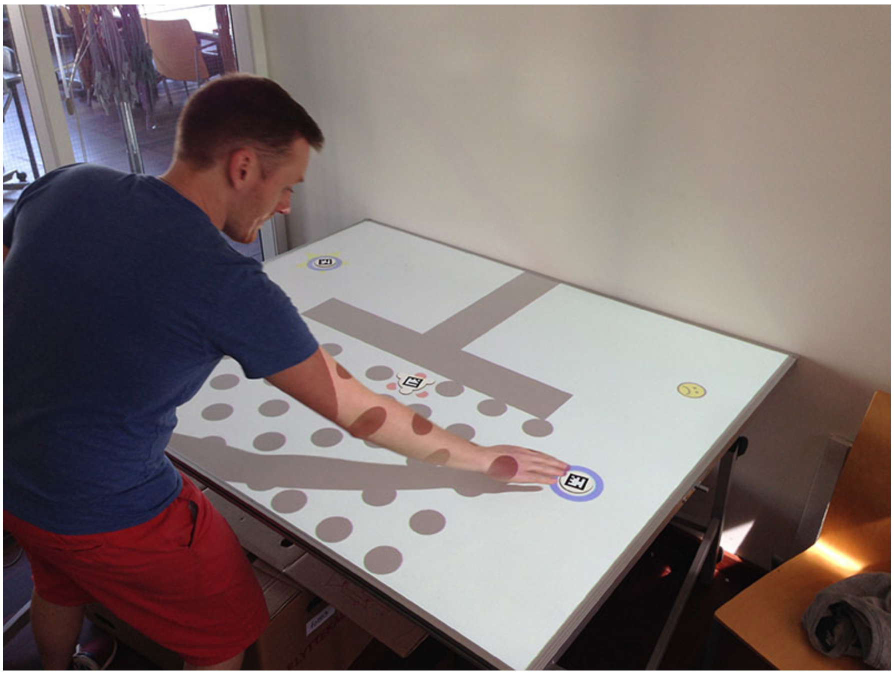


FIGURE 4.7: Final design

4.3 Technical documentation

In this first iteration there are four major technical implementations, a *tracking server*, a *main controller*, *the virtual domain* and *the physical domain*. This section will cover implementations in these areas and describe how they were implemented.

Figure 4.8 tries to outlay the system in a visual manner showing their context and relation. Each element is described in more detail below.

Physical Domain

The physical domain consist of physical objects within the application setup. The surface for the interaction is in this case consisting of flat white surface with the dimensions: 150x122cm.

The tangible objects, designed as way-points for the agents with different shapes according to different attributes. The objects are designed to hold a certain affordance to it along its unique size and shape. The objects are lasercutted in a compressed cardboard material with a white surface.

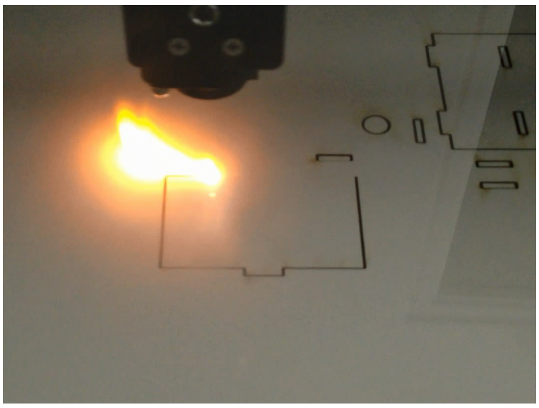


FIGURE 4.9: Laser cutting in progress of a cardboard object

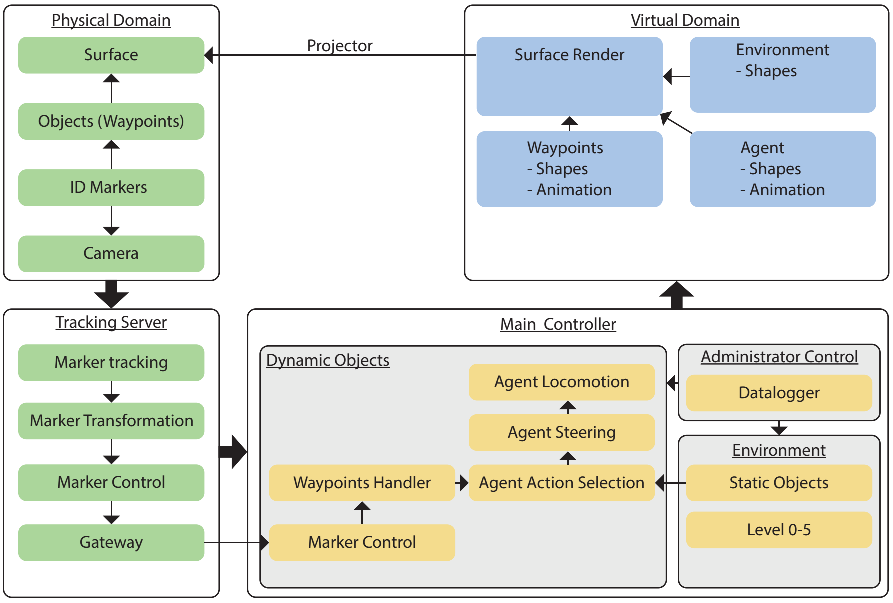


FIGURE 4.8: Diagram of the Framework

Trackable marker tags with a distinct pattern for differentiating between markers. The markers are confined of an outer black border which encloses the data package area within, the data package consists of 4 tracking dots and a set of dots representing a binary system creating a uniqie ID for each marker of up to a 8bit combination - 255 unique ID's. Figure 4.10 shows an example of 3 unique markers.

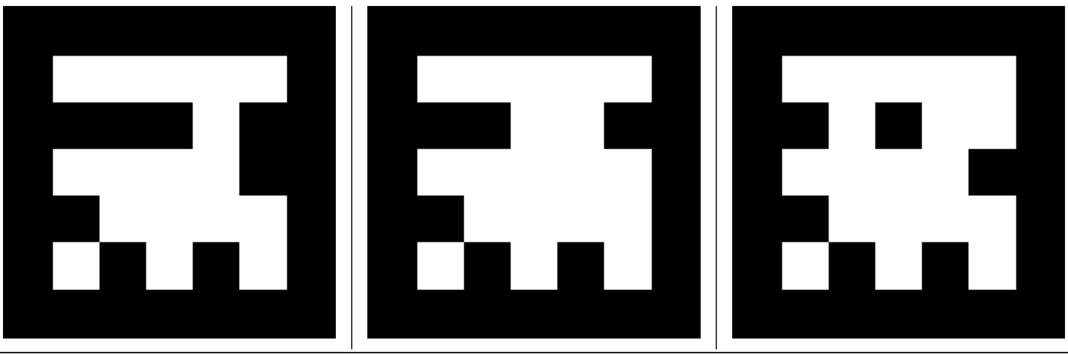


FIGURE 4.10: Sample of ID markers for tracking - The markers are unique in the binary code embeded in the composition of the black squares

A simple webcam with a resolution of 1280x720px, the high resolution is required for a larger distance to tag size ratio.

Tracking Server

The tracking server is a secondary application which handles the camera feed and deduces the tracked markers into surface positions.

The tracking system is based upon the NyARTToolkit[1] Library for Processing[2] which is designed specifically to track squared markers in 3D space. The NyARTToolkit library provides the positions of the four corners of a marker, enough information to extract a precise position and direction of the markers in 3D space.

Marker Transformation

Handles the transformation of markers in 3D space unto the physical surface plane. Initially it was implemented by creating the surface reference plane by 3 calibration markers defining the corners of the surface creating a plane in the same 3D space as the tracked markers. The markers position in 3D space is thereby projected onto the plane providing 2D coordinates within the reference plane of the surface. Due to the process of tracking calibration points on the surface, it created varying results of the projected position up to a difference of approx. 5cm variety and was therefore deemed too imprecise for the application to work.

The second approach uses the camera view pixel coordinates to determine the coordinates mapped on the surface within a predifined area. The predefined area can be seen on 4.11 in the camera view image, where the black border indicates the mapping surface. A code sample can be seen in appendix C.1

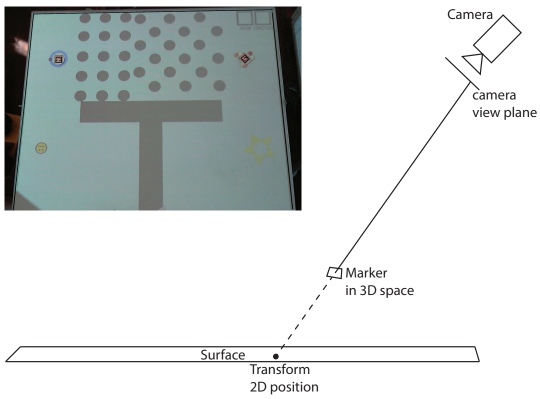


FIGURE 4.11: Camera View and Marker Transformation Illustration

This system stores the tracked markers and checks for any changes within the position or direction of the marker in which case the changes are send through the gateway.

A TCP/IP network protocol, sending information between the two systems. The message system uses JSON objects to send information in the form: {"ID": value, "x":value, "y":value, "rotation":value}. In the event where a marker has been removed from the tracking surface, the marker ID is sent and interpreted as a delete command: {"ID": value}

[1]http://nyatla.jp/nyartoolkit/wp/ by Ryo Iizuka
[2]http://processing.org/ by Ben Fry and Casey Reas.

Main Controller

The main controller implements all logic for of the system. it combines the tracked marker positions, divides it into the corresponding elements of the application, and the environment into a full system with the environment maps and all its objects.

The *marker control* recieves the information from the tracking server through the TCP/IP network protocol. It divides the corresponding markers ID information to the waypoint handler and the necessary information to the Agent.

The *Waypoints Handler* Stores the information about the specific waypoints, their position, direction and corresponding attributes. This information is shared specifically with the Agent Action selection system.

Agent Action Selection

The Agent action selection system implements all logic for the agents, its decisions and outcome.

Figure 4.12A shows in broad terms the agents logical implementation. It consists of three different action states: *idle*, *think* and *move*. During the *think* state it implements a path finding algorithm based on the information from the waypoints handler, where it determines the path by connecting the closest un-used waypoints, this method allows the agents to create a path through the environment based on the users interaction through markers. When the path is determined it applies a collision detection algorithm which checks the path for any obstruction in form of static object in the form of obstacles. Figure 4.12B shows a visualized image of the implemented collision detection, the red lines indicates colliders (boundaries used for detection) and the green area the determined area of collision. The volume of the collided area determines the confidence factor used in determining the reflected confidence of the agent. Success is determined if there is no collision and the agent will achieve its goal. Fail occurs when the agent either directly collides or does not achieve its goal.

Agent Steering

The agents steering behaviour is implemented according to Reynolds [23] where the steering is the resulting direction of a geometrical calculation of all affecting forces in the form of vector calculations. The implemented steering behaviours consists of inertia, seek and avoid. Inertia

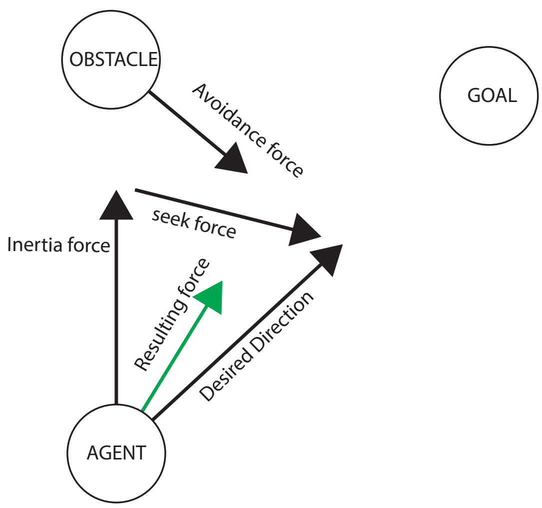
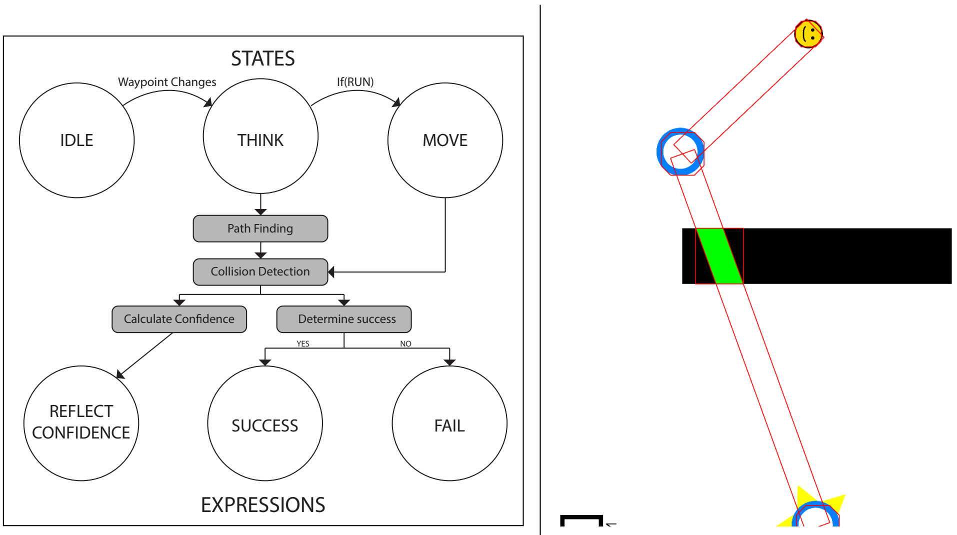


Figure 4.13: Geometrical calculation of steering force



(A) State Machine (B) Visualized collision detection

FIGURE 4.12: Agent Action Selection

is determined by the agents current direction, seek by its desired direction and avoid by a force away from obstacles.

Agent Locomotion

Handles all animation and movements of the agent, in this application it is designed with minimalistic movements following the steering direction. Furthermore it also implements the agents expressions; *confidence level*, *success* and *fail*.

Environment

The environment levels are designed using Scalable Vector Graphics format (.svg) which is a XML based vector image format which allows us to visually design the levels through an editor such as Adobe Illustrator and load it directly into the environment with correpsonding shapes and positions. The static objects of the environment consists of holes and walls, both shown in black colors. The two different static objects affect the agent differently. A total of 6 levels are designed to increasingly explore and teach the user of the new elements and increase the difficulty of the level (See figure 4.2.

Administrator Control Frame

The control frame allows us to control the application directly, hereby starting and stopping the game, choose specifik level etc. Furthermore it includes a datalogger, which stores data from the sessions every 500ms. The output generated includes: *time*, *Level*

Number, *Number of waypoints in use*, *Collision detection*, *The agents state*, *No of fail in current level*, *Agent confidence*, *Moving*. The data is used for statistical evaluation of the session. Additionally the application generates a screen shot when the user sets the application in motion, this shows the current path of the agent, its colliders and possible collision areas which is normally hidden from the view of the application.

The *Virtual Domain* Is the graphical presentation of the application which consists of the shapes and animations form the agent, waypoints and the environment.

4.4 Evaluation

In this iteration it has been the aim to present the user with problem solving tasks. Each task made use of the digital agent, and the user could get help from virtual agent through its expressivity. The virtual agent system was designed to express how confident it was, that the user solution was correct, this was done through the use of a projected smiley face.

The test was designed as an observation study to clarify questions relating to the interaction with virtual agents and the user expectations. The main focus was the relation between the time it took the participant to complete the task and how much they used the agent expression as part of the problem solving.

The test scenario consisted of 6 different levels. In each of these levels the goal was to get the virtual agent from its initial position to a designated goal - marked with a star projected onto the TUI. To do so the participant was given some physical tokens each representing a waypoint which the digital agent could follow. The participants were informed that the agent would always move to the nearest waypoint and continuously do so for each waypoint it passed.

The test participants were encouraged to try as many times as they wanted and were also told that there was no measure of performance.

After the test the participant was presented with a questionnaire (See appendix A) asking questions relating to the user and perception of the agents expression of confidence. Two questions relating to the understanding of the use of the interface and how the behaviours applied to the virtual agent. Lastly there was a question relating to how the user experienced cooperation between the participant and the digital agent.

4.4.1 Data Collection

Throughout each level of the system, both time and number of failed tries is measured to determine the approach the user have. The relation between the time and number of failed tries will provide knowledge of the users error rate, a number representing the number of failed tries per minute. The error rate will provide information about the learning curve throughout the problem solving tasks. The learning curve should be able to show whether or not they were able to perceive information from the digital agent, leading them to the correct solution.

The questionnaire is build up of 5-point likert scale questions, allowing the user to express their use of the virtual agent and how well they understood the aspects of the environment. (See appendix A for full questionnarie)

4.4.2 Results

The test was completed by 6 participants consisting of students from from Aalborg University Copenhagen. Figure 4.14 shows an overview of the results from all the questionnaire questions as a histogram of the 5-point likert scale answers.

Figure 4.14A shows if the expressions of the agent influenced their problem solving approach and whether they used the agent to estimate the solution. The answers were as follows, one participant answered 1, two participants answered 2, one participant answered 3 and lastly two participants answered 4.

Figure 4.14B shows to what extend the expression of the agent met the users expection of the outcome. Four of the test participants answered 3 while the last two participants answered with a score of 5.

Figure 4.14C shows how confident participants were that they understood the physical waypoint they were first introduced to (Seek behaviour). One participant answered the question with a score of 4 and the last five participants answered the question with a maximum of 5.

In figure 4.14D the confidence of how well participants understood the second physical waypoint they were introduced to (Avoidance behaviour), is presented. Two participants answered with a 3, one participant answered with a score of 4 and the last three participants answered with the maximum of 5.

The figure 4.14E shows to what extent participants would describe the problems solving as a cooperation between the participant and the digital agent. The results are almost evenly spread with one participant for each score, except for the two participants who scored it with a 4.

The logged data shows a mean error rate for all subjects of 0.9575 fails/min and standard deviation of 0.1334, the low deviation across subjects indicates that the subjects have approach the problem solving in a common manner. Figure 4.15B shows the individual error rate for all participants as a bar graph.

Figure 4.15A shows the number of fails over time for each subject separated by color, where the error rate is the slope of the line through the individual subjects plot.

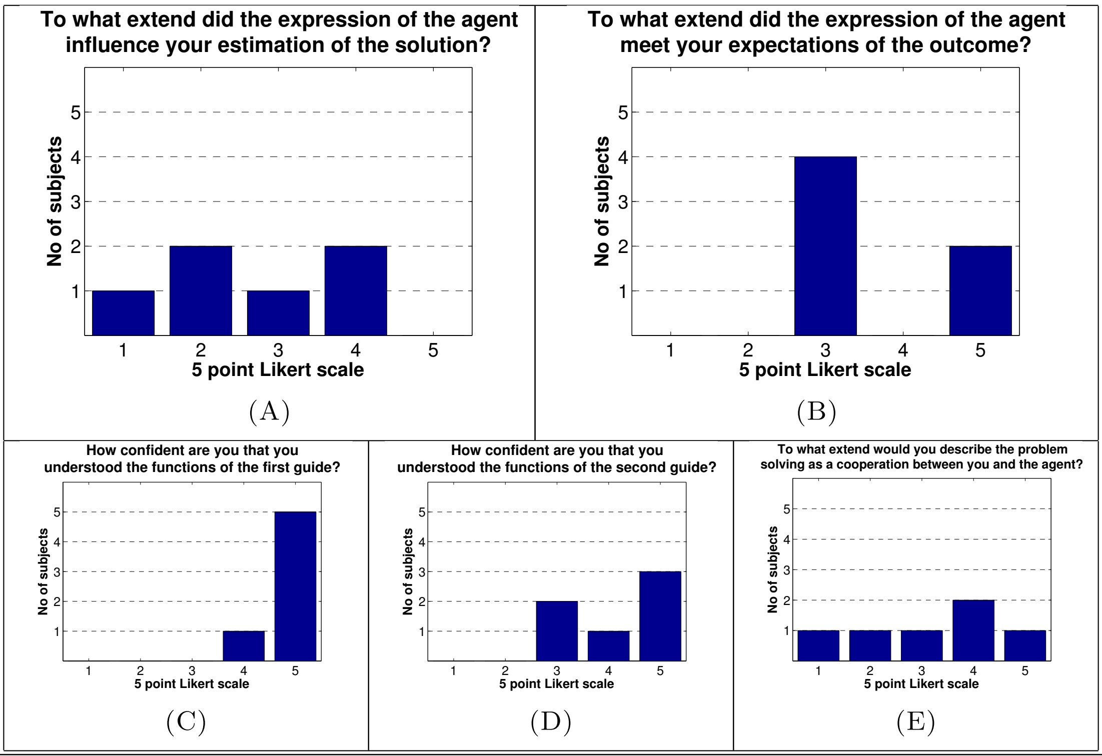
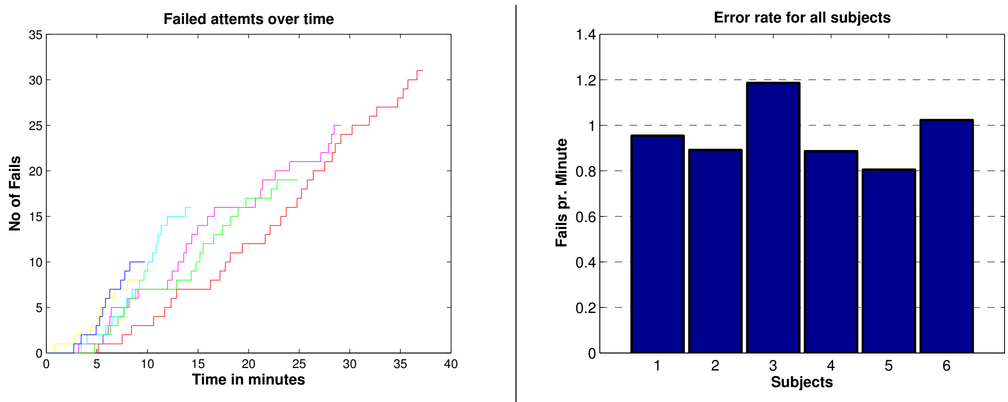


FIGURE 4.14: Histogram of 5 point likert scale questions
Ranging from *Not at all* - *Completely*

From Figure 4.15A is is clear that some subject used less time than others, more specifically the two participants (blue and yellow) finished the 6 levels in less than 10 minutes, where one participant (red) used over 35 minutes to complete the same problem tasks. Though there is a time difference the subjects still have a similar error rate (slope) the one participant simply just had more difficulty solving the tasks.



(A) Plot showing fails over time

(B) Error rate for all subjects

FIGURE 4.15: Error rate results

4.4.3 Findings

As the test consisted of only 6 test participants, no statistical evidence can be provided, it is believed that the results in combination with informal observations throughout the test will give an indication of how this iteration can be evaluated. This section will describe the relation between data collected from the log, questionnaire and observations.

The considered high error rate indicates that the participants used a trial and error approach to the problem solving. This is also apparent from the low and wide spread score in the question of cooperation and the observation during the test which clearly indicated that the participants disregarded the information from the virtual agent. The assumption was that a successful cooperation with the virtual agent would have provided the participants with the knowledge to solve the solution with less failed attempts as they got more familiar with the digital agents evaluation system. Given that users would have understood and used the digital agents expression, this would have resulted in a lower error rate.

The participants did not seem to use the digital agent to evaluate their solution, but they did however appear to understand the expressions of the agent. This could mean several things, as the agent would both show an overall solution estimation prior to trying the solution and would also on activation show the current estimation between two points. It is unclear at what time the users where able to understand the expressions of the agent. One example of this is the agents expression upon failure, where it would show an angry red face. This expression would however only indicate immediate failure, but would not help the user to solve the problem beforehand.

In the question of whether the participants experienced the problem solving task as a cooperation between the participant and the digital agent (see figure 4.14E), the results indicate no tendency. It is not possible to conclude something from these results, which could be due the interpretation of the question. Some feedback from participants afterwards were that users saw the interaction as being complete user control with little or no support from the agent (100% intervention 0% autonomy). Some users even suggested that the agent should be able to solve these tasks completely by itself, and expected more intelligence from a digital agent.

Figure 4.14C and Figure 4.14D shows if the subjects understood the waypoints used for guiding the agent through the maze. The first waypoint was well understood and correctly used by all subjects, which was highly anticipated due to its simplicity. The second waypoint varies a little more which is also indicated by the descriptions provided for each waypoint (See appendix A) where subjects have described it as "Bouncing", "Multi-path guide" and "complicated to understand".

4.5 Discussion

From the results and the observations of this evaluation, we do not believe it is possible to conclude that the implementation of expressivity in a digital agent and the communication to the user was successful, rather the contrary. The aim was to explore the expressivity of an autonomous agent in a problem solving environment, but as it appears that the subjects did not use the virtual agent as cooperative help for solving the task, there is not a lot of data regarding the expressivity of the robot that can be developed upon. Though participants have understood the expressions of the agent they were not put to use. While observing the subjects interacting with the application it became apparent that they tend to disregard the agent and merely estimate the solution themselves. Thus, the subjects did not consider the virtual agent as an intelligent help that could contribute to the problem solution.

One of the key issues with this system, is believed to be an undesirable division between intervention and autonomy. The feedback from users, when evaluating the test through informal talks, indicated that the digital agent was perceived as a tool and not and agent with any autonomous capabilities. This division appeared to affect the user to spend most of their time manipulating the solution and testing it out, leaving little or no attention to the digital agent. This issue of how to design the ratio between intervention and autonomy is going to be a key focus in the future development of this system. One could even argue that the user interaction and expressivity should be treated seperately as a next step.

Chapter 5

Iteration 2

Based on findings from the design presented in iteration 1, this iteration will move towards embodying the digital agent system in a physical mobile robot. The physical presence of the agent will allow us to expand on the expressive features of the agent through movement and lights.

In terms of the balance between autonomy and intervention, much of the feedback from iteration 1 showed a conflict between users actively having to solve a problem solving task and concurrently being attentive of a digital agents expressivity. As our findings in iteration 1 indicate, the solution estimation from the digital agent did not seem to significantly affect how users approached the problem solving. This iteration will therefore be focusing mainly on the expressivity of an autonomous robot with little to no intervention from the user. Some of the existing research of implementing personality in autonomous robots will be presented and used in the development of this iteration.

The system will be based on the same system as seen in iteration 1, but will generally be aiming towards replacing graphical representations of objects with physical objects. The system is able to load arbitrary level designs and create a layer of virtual colliders and paths on top of the TUI platform. This allows the system to scale to wide variety of scenarios and put autonomous robots and their expressivity in a context. The use of adding a context to the autonomous robot will also be described and explored in this iteration.

5.1 Analysis

As it has been briefly described above, this iteration will be about investigating behaviours in a digital agent; in this iteration embodied in an autonomous robot. As it was also described in the preliminary research section 2.5 Meerbeek et al. presents a list of desirable personality features in robots[16]. These will be part of understanding how to approach behaviours and expressivity in robots. The approach to behaviours will also be inspired by related research in movement and point light expressivity to design and develop the physical expression of these personality traits.

5.1.1 Point light source expressivity

As technological advances are making their way into our everyday lives through computers and other devices - robots inclusive - we also see an increase in ways of notifying user of changes in the state of these devices. From mobile phones we know lights, sound and vibration can be used to inform a user of calendar notification, calls, emails or messages. The iRobot Roomba also utilises sound through little melodies and led light notifications in different colours and patterns to notify the user of battery levels and cleaning modes.

According to Harrison et al. the design space of point lights can be rich when making full use of the change of intensity over time. However current use of point lights is thought to be narrow and unimaginative[24]. The work with point lights and the proof-of-concept behaviours proposed by Harrison et al. is with a single white point light. The guidelines and proposed behaviours is an interesting starting point for working with slighty more complex behaviour with additional point lights and colours to choose from.

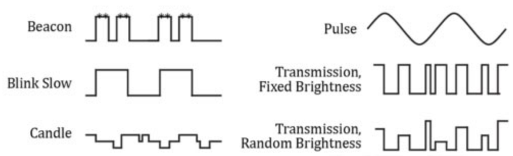


FIGURE 5.1: 6 of the 24 proof-of-concept light behaviours used in this project[24].

By adding additional light and colour features to light expressivity in a robot, it is believed that it is possible to add colour associations and spatially (right and left) to the expressive behaviours. These are little expansions to the research in light behaviours by Harrison et al.[24]. But would allow for a greater variety of behaviours to explore in the physical space.

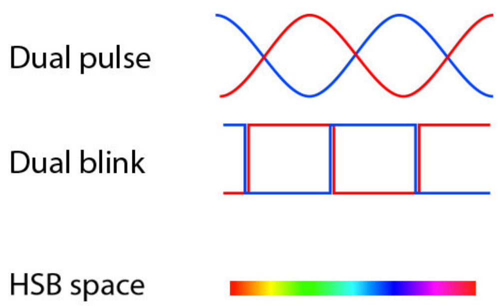


FIGURE 5.2: Expanding on light behaviours with one additional point light and Hue-, Saturation- and Brightness features.

5.1.2 Movement and Physical Presence

One of the observations from iteration 1 indicate lack of attention to the digital agent. One of the reasons for this lack of attention to- and use of the digital agent as a supportive tool to solve the task could as described be that participants were preoccupied with the given problem solving task. Another reason could be the limited physical presence of the agent; being only a projected image on the TUI surface.

A widely recognised argument in the importance of interaction with physical object and spatial presence of embodied systems is that we ourselves exist in the physical world. We as humans have have sensory experiences with the world and we perceive and understand the environment around us in terms of how our sensory systems would expect objects to feel, be manipulated and its affordances[8]. By embodying a digital agent into a physical robot we draw on human experience and knowlegde we have gained through a lifetime of interacting with materials, for example temperature, surface quality, softness and weight[8].

It is in this physical space we can experiment with movement and presence of a robot and how this can be utilised in expressing different internal states of the system. One study shows that personality can be inferred by a number of cues such as physical presence, language and gestures. This study is based on human-human interaction. Among non-verbal cues *gestures*, *body movement* and *facial expression* are found to be the most reliable cues to determine personality from among humans.[25]

Other than communicating personality more reliably, the physical space also allows for placing a user in a specific context. Since a TUI platform is a physical space with all the benefits it may entail from a HCI perspective, it is still a simulation of the real world. Some companies use TUI's to train warehouse apprentices through a project called TinkerSheets and TinkerTable[13]. TinkerSheets and TinkerTable uses a similar setup to this project, to simulate a warehouse with physical shelves and virtual forklifts and security zones projected on the table. This miniature simulation of a context can be presented through recognisable objects from the real environment to induce a better relation to the environment and context the interaction is presented in.

By working towards an embodied agent in the physical space that can express its internal state through its body movement and gestures, the expressive features should gain effect to the user.

5.1.3 Interpreting Actions and Intentions

Expressivity in robots can be used depending on the purpose of the robot and a users expectations to its abilities. In social robotics robots expressivity is often about being able to interpret human interaction and act in a social context where focus is on facial expression, showing emotions, speech and generally imitate the human ability to express itself. In domestic robots expression can be said to be of a less complex nature. Domestic robots have a purpose and function, this function is oftentimes designed to relieve the human of these tasks by either doing it better or more convenient.

From iteration 1 some observations indicated that users did not attribute the digital agent any significant intelligence, function or personality as it did not exhibit a particular intelligent or autonomous behaviour while moving around in its environment. All of the autonomy while moving was user instructed. This also meant users were not able to read any personality or other intentions from the digital agent.

To be able to interpret actions or intentions of a robot, there must exist and underlying system, containing a model of what the robot is about to do, how it is going to do it. If this information can be communicated effectively, users are more likely be able to interpret abilities of the robot and act upon those assumptions.

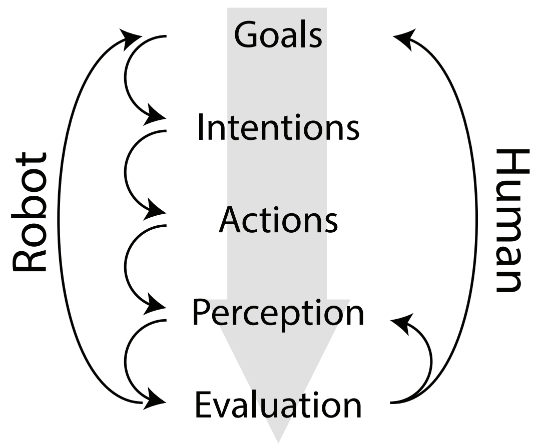


Figure 5.3: Illustration of the implemented gestures

In figure 5.3 the model shows the use of Donald Normanns 7 steps of actions [5] used in a HRI context. This model is also used by Scholtz to describe the supervisor role of humans when interacting with robots that possess action planning systems, such as autonomous robots[2]. The model shows how the 7 steps of actions can be used to illustrate a robot action selection systems as well as the human approach to interactions in general. On the left the robot action selection system is illustrated as a continuous loop from goal to evaluation of its actions. In this process the robotic system is designed to form *intentions* and *actions* that allow it to complete its goal. It must be clarified here, that in robot systems, intentions and actions are individual systems and sub-systems of low-level locomotive and planning algorithms, that over time make out the overall intention to and action of solving a goal. When this project aim to express intentions and actions of a robots it is in reference to the high-level intentions and actions. On the right side of the model the human role in the interaction with the autonomous robot is illustrated. The human can by continuously evaluating the robot performance do two things. If the robot is performing as expected, the human supervisor does not need to take any action. However if the evaluation does not meet the users expectations of the robot performance, the robot goal selection can be altered.

5.2 Design

Initially the idea was to design the same scenarios as the previous iteration, with changes according to the findings, in order to evaluate the robots expressions through a scenario. The first iteration clearly showed some major drawbacks when interacting with an autonomous agent and the intervention therein. The ratio between intervention and autonomy had to be changed to ensure a higher level of autonomy and less intervention for the user to perceive the robot and understand it expressions. It is therefore concluded that the environment should provide the possibility for the robot to express itself in a predefined scenario represented by physical objects. The robot should be able to autonomously navigate and perceive this environment to solve dynamically generated goals.

5.2.1 Application specific domain design

The focus of the project is to explore expressivity in simple domestic robots. The context in which the robot expresses itself affects how the user perceives its expressions. This contextual expressivity also affects the evaluation procedure and the subjects' understanding of the robots expressions. Therefore the context is designed to simulate a simple utility robot in a supermarket scenario, because it is highly feasible and assumed to be understandable for any evaluation subjects. It is important to note that the system is being designed to be scalable to any context for which robot expressions are desirable.

Supermarket

The supermarket design is an environment in which the robot can perform routine tasks among a set of shelves and express its behaviour along the way. The supermarket allows for designing a set of routine tasks which in this case is decided to be *Washing*, *Vacumming* and *Restocking*, each routine tasks is chosen for its easily perceived relation to the context of the supermarket and both simplistic action and more complex such as restock.

5.2.2 Intervention

The context of a supermarket allows the robot to autonomously perform routine tasks, initially we designed a shared control system for intervening with the robots routine tasks. By adding special markers for the user to interact with the robots routine, the user could select specific tasks which should be prioritized by the robot. This should encourage a form of collaboration between the user and the robot. Through pilot testing of the system it became clear that the interaction with the system moved the subjects attention from perceiving the robots expression towards controlling the robot. Therefore, because this is an exploration of the expressivity of the robot, the user interaction was minimized in order to keep the focus on perceiving the expressions of the robot. The new minimized intervention system consisted of the user being able to prioritize an area of the supermarket for the robot to perform the tasks, but even so the interaction intervened with the main focus and was also removed, leaving the scenario to be completely autonomous with no interaction, to assure no complications with the subjects' perception of the robot's expressions.

5.2.3 Designing Robot Expressions

Physical Gestures

The three routine tasks' physical gestures are designed according to the presumed association of: *Vacuuming* expressed as the regular motion of moving the vacuum cleaner back and forth across a surface, *Washing* expressed as a circular motion of rubbing or polishing the floor, as you would with a mob or cloth and *Restocking* expressed as a three-point back and forth movement along the shelf, as if you were filling up the shelf from one end to the other.

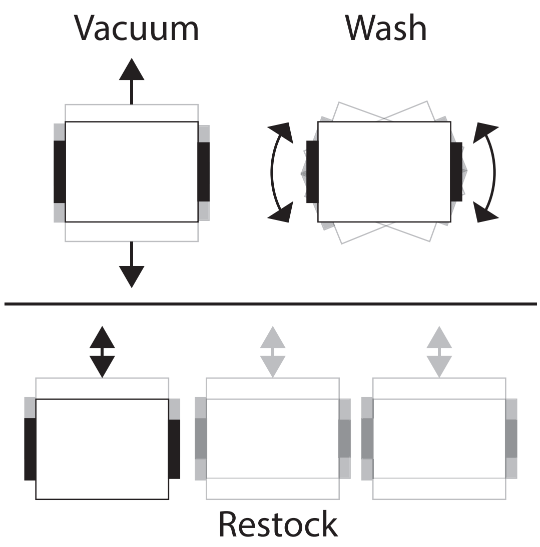


FIGURE 5.4: Illustration of the implemented gestures

Light expressions

According to Harrison [24] a blinking behaviour is considered working, therefore we have created a dual blink behaviour with different colours each represtenting a task: *Vacuum* with a white neutral color, *Washing* with a blue color which could be perceived as the color of water, *Restock* with a yellow color seen in construction vehicles.

Furthermore as the robot is moving from one task to another, there is designed a dual pulse light behaviour with the same colors for the users to percieve the intentions of the robot.

5.3 Technical documentation

The implemented framework for iteration 2 is an expansion of the first iteration where all technical documentation also applies to this iteration.

Based on the same framework illustration as the first interface with some modifications. The main changes lies within the robot locomotion, but also changes to the environment and agent action selection. The logic for the robot is expanded to fit a more autonomous robot without the need for user supervision. Most importantly it implements a Dijkstra path finding algorithm from a network of node in the environment and not the same system for waypoints as the previous iteration, a good sample of the dijkstra path finding algorithm can be found in appendix C.4

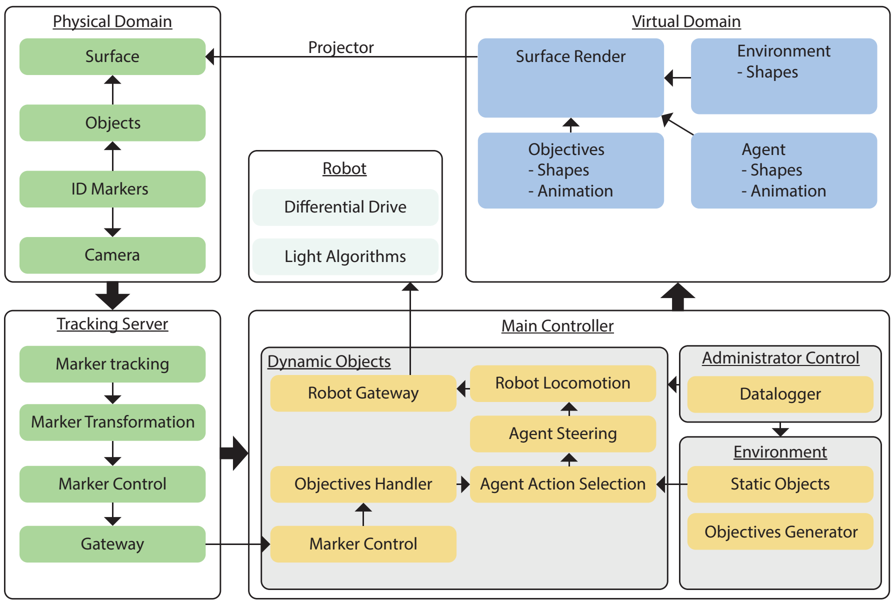


Figure 5.5: Diagram of the Framework for iteration 2

5.3.1 Robot Locomotion

The robot locomotion uses the steering direction derived from the agent steering component similar to iteration 1. By applying a unit circle rotated by 45° according to the current direction the differential motor forces are calculated by the sin and cosin of the angle a (see figure 5.6). The current design does not employ the use of the robots reversed driving mechanism, therefore when the difference angle between the current direction and the steering direction exceeds a limit of 1.5 radians it overrides with a u-turn movement applying opposite forces to the differential motors causing a complete turn.

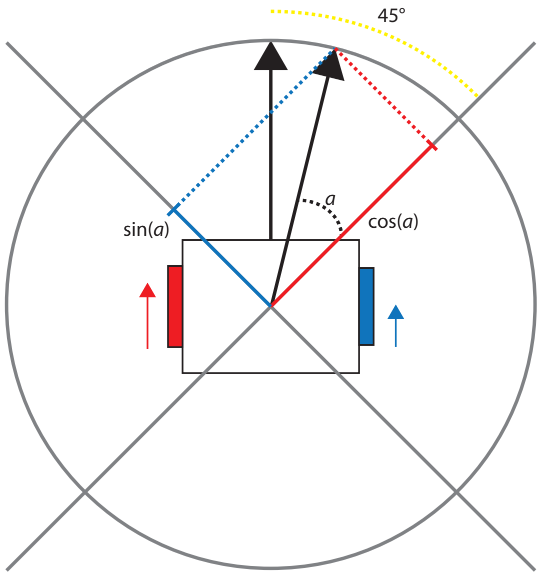


FIGURE 5.6: Illustration of the geometrical calculation used for calculating the differential motor forces (red & blue)

Special movements like the gestures performed are programmed time dependent sequences. Figure 5.4 shows the three different gestures, where e.g. *vacuum* is a sequence of forward motion for 500ms and reversed motion for another 500ms, repeated within a time frame of 3000ms. *Restock* is a more complex motion but it uses allocated points in the environment to navigate around the shelves. These extra points are used to approach the shelf at three different positions along the shelf (see videos for all behaviours in appendix D.3).

Robot Gateway

The robot gateway is a TCP/IP network protocol sending packages from the main controller to the robot through WIFI connection. The package encoding is similar to that used for JSON object but simplified due to the microcontrollers limited memory. Package format: {identifier<int>, input-1<float>,...,input-n<float>} depending on the identifier the robot reads a specific number of input arguments passed along. Motor values are send every 100ms and light values every 500ms. See appendix C.3 for code sample.

Robot

The physical body of robot was designed and made in compressed cardboards material and cut in a lasercutter which provided both speed and precision in the process compared to the initial 3D printed design. The robot consists of the elements in figure 5.7. All design files for the robot's exterior can be seen in appendix D.4 and the finished robot can be seen in figure 5.8.

The micro-controller is an ATmega644PA chip on a Microduino board[1] which is comparable to the popular Arduino Mega2560[2] but the size of a quarter.

- 2 Servos - hacked for continues rotation
- 2 wheels - PMMA clear plastic with o-ring seal for traction
- Microduino core+
- Microduino serial connection
- Microduino Wifi shield
- 2 RGB light emitting diodes
- 3.7V Battery
- ON-OFF toggle

FIGURE 5.7: Robot content

Light Algorithms

The light algorithms are programmed on the robots microcontroller and not in the main controller for a seemless control of light sources. The algorithm are implemented as states which can encompases any given color saturation. Some of the implemented and used algorithms are, *Work*, *Pulse* and *Beacon* which can be found as code in appendix C.5

Objectives Generator

For the evaluation purpose, an automatic generator of objectives consisting of *Washing*, *Vacummin* and *Restocking* is implemented by randomly generating an even distribution of objectives for the robot. The washing and vacumming objectives are randomly placed within the 2D space of the surface without interfering with static objects. Restock is

[1] http://www.microduino.cc/Modules/MicrodoinoCoreModules/Microdoino-Core-Plus(28644p)
[2] http://arduino.cc/en/Main/ArduinoBoardMega2560

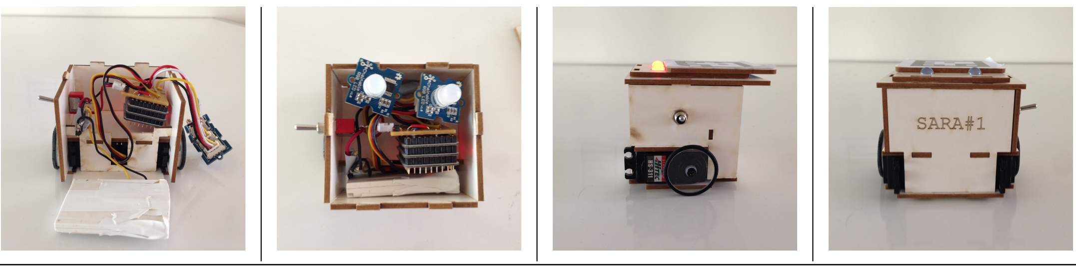


FIGURE 5.8: Robot - SARA#1

generated on boths sides of the 12 shelves in the environment as this objective is dependent on the static ojects. Objective generator code sample can be found in appendix C.2

5.4 Evaluation procedure

We will be evaluating three different methods for a robot to express its actions. The methods of expressivity has been chosen from current uses in simple robots where motor gestures and point lights are already being utilised. These two methods will be evaluated as separate cases, to see how they each perform. Additionally we will combined the expressivity of the two methods to see if there are benefits from this combined method.

- Gestures only
- Light behaviours only
- Light and gestures combined

The hypothesis of this test is presented as a null-hypothesis and tested against an alternative hypothesis. The null-hypothesis of this experiment is:

"H0: There is no significant difference between using lights and gestures to express robot behaviours."

In order to be able to compare the performance of each participant under all three conditions the test is designed to be within group. The three different scenarios will be presented to each test participant. To be able to account for any learning effects that might occur during the test, the order in which they are presented will change according to test participants. Furthermore with each participant being presented with all conditions, the aim is to keep the test to a maximum of 20 minutes to account for any fatigue and annoyance which could impact the performance negatively.

The test participant will be presented with a TUI representing a small supermarket. The TUI consists of 12 shelves making up the isles of the supermarket. In this supermarket the semi-autonomous robotic agent (SARA) is presented as being an autonomous multipurpose robot. SARA is able to complete three tasks. These are:

1. Clean up water spills.
2. Vacuum dust.
3. Restock the shelves.

Each of the tasks are projected as red dots on the TUI table and over time more tasks will be added. These red dots will be representing one of the three tasks to be performed in the supermarket. As SARA moves through the supermarket doing its job it will express its actions and intended actions through behaviours that are either light behaviour, gestural behaviours or a combination of both light and gestural behaviours.

The observational part of the test will run over the course of 15 minutes divided into three sessions of each 5 minutes. Through each session the subject is asked to answer which of the three different tasks it is performing, and if confident which tasks it is about to perform, which will be measured as correct or incorrect answers. For each of the five minutes SARA will express its actions with different expressive methods.

FIGURE 5.9: Test Scenario

After each of these 5 minute sessions a questionnaire will ask questions relating to the robot's behaviour to see if the participant is able to correctly understand its actions and intentions. After the test we will be presenting the participant with questions related to the overall experience of these behaviours and how well they are able to recollect some of the specific behaviours they experienced.

5.4.1 Data Collection

Throughout each sessions, the subjects responses to the robots expressions of the behaviours will be recorded along with the correct answer in the quantitative data log.

The questionnaire is build up of 5-point likert scale questions, allowing the user to express their confidence for each sessions for how well they understood the expressions and a set of end questions where they shall describe the perceived behaviours in text form. (See appendix B for full questionnaire)

5.5 Results

This section will show the results from the evaluation of iteration 2. It will present the data from both the datalog files and the questionnaire. Througout the section the different sessions (*Gesture*, *Light*, *Combined*) will be denoted as conditions and the robot action types (*Wash*, *Vacuum*, *Restock*) will be denoted as types. The answers are divided between actions (answers while the robot is performing a type) and Intentions (which type the robot is about to do).

The test was completed by a total of 13 subjects consisting of student from Aalborg University Copenhagen with a median age of 25 years spanning from 23-42 years of age and 2 female and 11 male subjects.

5.5.1 Data log

This section covers the data from the data log providing information about the robots actions and intentions according to the subjects answers.

Through the test sessions the subjects were asked to provides an answer each time the robot was performing an action. It was only if test subject were confident they understood the intentions, they would provide answers to the robots intentions.

Therefore the number of answers relating to robot actions directly reflects the number of the action related expressions presented to the user, whereas the number of intention related answers are lower as this was not required of the subject, if they were not confident they could provide an answer. Table 5.1 shows the answers related to the robot actions, where the sum in each test conditions provides validity that the number of actions presented for the subjects is evenly distributed for each condition. The sum of each type of expressivity presented shows that there is an uneven distribution between the presented expressive types which may affect the answers.

Actions				
No Of Answers	Wash	Vacuum	Restock	Sum
Gesture	34	10	77	121
Light	38	17	95	150
Combined	39	16	73	128
Sum	111	43	245	

Table 5.1: No of answer for each type and condition for robot actions - Total of 399 answers

Table 5.2 shows the answers for the robots intentions. These numbers a significantly lower and does not represent the number of expressions the subjects were presented with, but is rather a result of the subjects confidence that they understood the robots intentions.

Intentions				
No Of Answers	Wash	Vacuum	Restock	Sum
Gesture	1	0	4	5
Light	6	3	17	26
Combined	8	6	18	32
Sum	15	9	39	

TABLE 5.2: No of answer for each type and condition for robot intentions - Total of 63 answers

Hypothesis testing

To test for the difference between the different conditions, the results of the action related answers will be used.

C1: *Gesture*

C2: *Light*

C3: *Combined*

A paired-sampled t-test was performed with a degree of freedom of 12 (n-1):

C1-C2: $t(12) = 4.2742$, $p = 0.0011 < 0.05$

C1-C3: $t(12) = -0.0976$, $p = 0.9239 > 0.05$

C2-C3: $t(12) = -3.3790$, $p = 0.0055 < 0.05$

There is a significant difference between condition 1 and 2 (*Gestures* and *Light*) rejecting the null hypothesis:

H0: *There is no significant difference in using light behaviours and gestures to express robot behaviours.*

Furthermore there is no significant difference between condition 1 and condition 3 (*Gestures* and *Combined*).

There is also a significant difference between condition 2 and condition 3 (*Light* and *Combined*).

Correct Answers

This section presents the answers retrieved from the subjects as a percentage of correct answer of the actions performed. Figure 5.10A shows the mean correct answers for each condition with a standard deviation across subjects individual mean value in each condition. The total mean of all subjects' answers for all conditions are: mean = 65.23% with a standard deviation of 16.17% across subjects. For gestures condition only the mean correct answers are: mean = 76.26% with a standard deviation of 20.26% across subjects. For light condition only the mean correct answers are: mean = 42.39% with a standard deviation of 25.50% across subjects. For the combined condition the mean correct answers are: mean = 77.04% with a standard deviation of 20.83% across subjects.

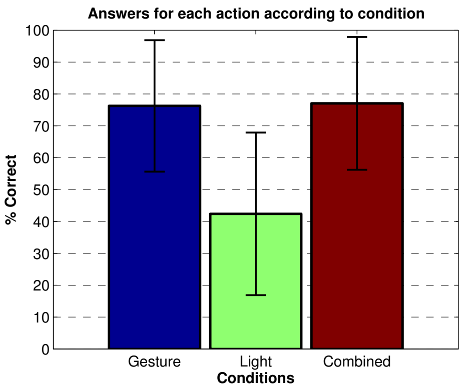
Figure 5.10B presents the same data divided into the three different types of actions. The reults for each type under each condition will be presented as mean percent correct and standard deviation across subjects.

When looking at the gesture condition the subjects where able to correctly identify the washing behaviour with a mean of 94.44% and a standard deviation of 12.00%, for the vacuuming behaviour the mean was 13.89% with a standard deviation of 22.15%, finally the restocking was correctly identified with a mean of 77.14% and a standard deviation of 23.22%.

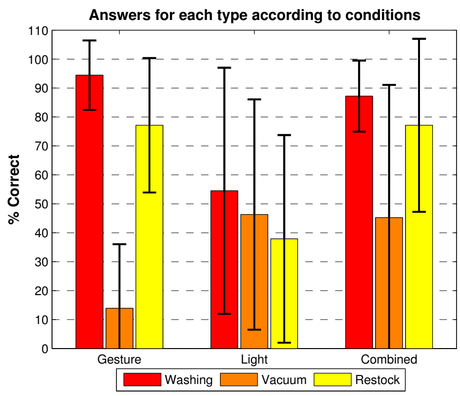
The results when looking at the light condition show that subjects where able to correctly identify the washing behaviour with a mean of 54.49% and a standard deviation of 42.54%, the vacuuming behaviour was correctly identified with a mean of 46.30% with a standard deviation of 39.77% and finally the restocking was correctly identified by subjects with a mean of 37.89% and a standard deviation of 35.88%.

When combining the behaviours in the last condition, subjects where able to correctly identify the washing behaviour with a mean of 87.22% and a standard deviation of 12.28%, the vacuuming behaviour was correctly identified with a mean of 45.24% with a standard deviation of 45.86% and finally the restocking was correctly identified by subjects with a mean of 77.13% and a standard deviation of 29.88%.

Figure 5.11 shows the individual subjects answers for each action.



(A) Mean correct answer for each condition for robot actions - error bars: ± 1 standard deviation across subjects



(B) Mean correct answer for each type according to conditions for robot actions - error bars: ± 1 standard deviation across subjects

FIGURE 5.10: Bar graphs of answers

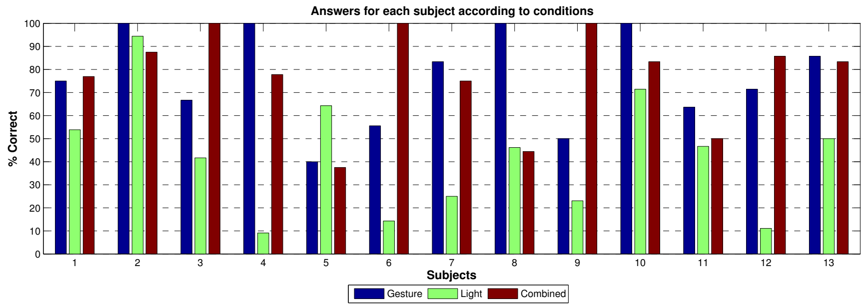


FIGURE 5.11: Answer for each subject according to conditions for robot actions

Correct Answers for Intentions

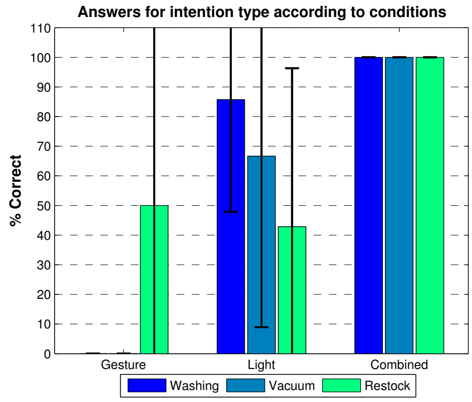


FIGURE 5.12: Mean correct answer for each type according to conditions for robot intentions - error bars: ± 1 standard deviation across subjects

Figure 5.12 shows the results from the answers of the robots intentions, the answer was encouraged if they felt confident they knew what the robot was about to do. It shows the total of 63 answers, see table 5.2 to compared with the amount of answers. Subjects only felt confident enough to answer questions related to the robots intention 15.79% of time. The numbers show that for the gesture condition the subjects answered correctly with a mean of 33.33% with a standard deviation of 57.74%. For lights the mean correct answers were 65.83% with a standard deviation of 42.39%. Lastly users were able to identify the robots intentions through a combination of light and gestures with a mean of 100% and a standard deviation of 0%.

Sessions order

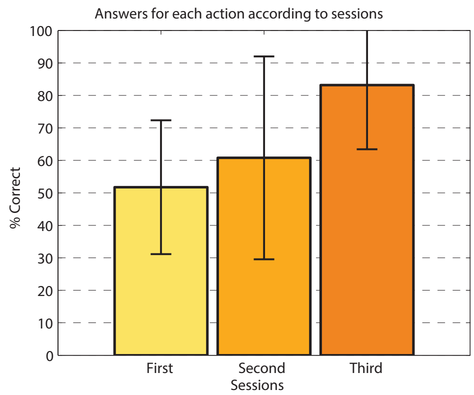


FIGURE 5.13: Mean correct answer for each session for robot actions - error bars: ± 1 standard deviation across subjects

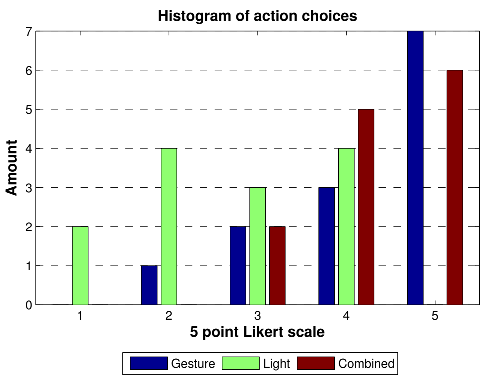
The three different conditions are presented through 3 different sessions, as explained in the procedure (see section 5.4), in a period of 5min. To avoid any learning effect during the three sessions the order of presentation of the conditions is changed according to the subject number. Figure 5.13 shows the mean correct answers for the subject in order of appearence.

For all the conditions when presented in
the first session the subjects scored a mean of 51.73% correct answers with a standard deviation of 20.61% across subject. In the second session the mean was 60.78% with a standard deviation of 31.23% and in the last session users answered correctly with a mean of 83.18% and a standard deviation of 19.75%.

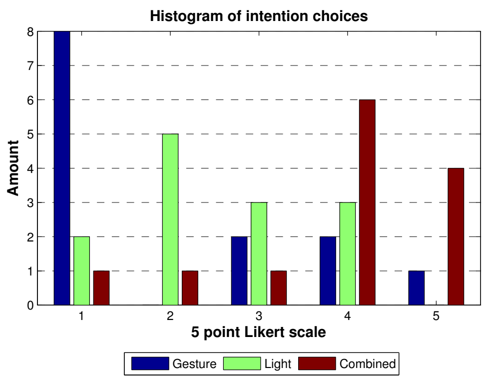
5.5.2 Questionnaire

For each sessions of 5min, the subject filled in two 5 point likert scale describing how confident the subject where between each session of the robots actions and intentions. Figure 5.14A shows the total score for all subjects as a histogram for the actions of the robot for each of the three sessions where the participants in the gesture condition scored it with; 1 subject answered with a score of 2, 2 subjects scored it with 3, 4 subjects with 4 and 7 subjects with a score of 5. For the light condition, 2 subjects scored it with 1, 4 subjects with 2, 3 subjects with 3 and 4 subjects with 4. For the combined condition, the subjects scored it with, 2 subject of 3, 5 subjects of 4 and 6 subjects of 5.

Figure 5.14B shows a histogram of the answers to how confidence they were toward it's intentions for each condition. Where the subjects answered to the gesture condition with; 8 scored it with 1, 2 scored it with 3, 2 with 4 and 1 subject scored it with 5. For the light conditions the subjects scored it with; 2 subjects with 1, 5 subjects with 2, 3 subjects with 3 and 3 with 4. For the combined condition the subjects scored it with; 1 subject with 1, 1 subject with 2, 1 with 3, 6 with 4 and 4 subjects scored it with 5.



(A) Histogram of level of confidence for actions ranging from, *Not at all* - *Completely*



(B) Histogram of level of confidence for intentions ranging from, *Not at all* - *Completely*

FIGURE 5.14: Histograms showing results from 5 point likert scale

Behaviour Descriptions

After the last sessions the subjects where asked to describe the 3 behaviour in text form. A few noteworthy subjects answers are shown here, for a full list see appendix B.

TABLE 5.3: Small sample of behaviour descriptions, full table: B.1.

Subject	**Wash**	**Vacuum**	**Restock**
6	Doing little random jumps - also orange lights.	it was very close to washing - but it didn't have those little jumps - it was more "clean"	It turned towards the obstecales and looked a lot like it was stocking up. think it was orange lights
10	It was scrubbing the floor.	Moving back and forward like you do with the vacuum nozzle.	It put things on different shelves by moving around and back and forth.
13	The first session was with lights only and I thought that blue light would equal water/washing. Later when I had seen movements, I felt confident that the twisting motion was a washing action - and it matched with blue lights :)	It was hard to tel just from the lights color, but when I saw the forward/backward motion it reminded me of vaccuming.	From the lights I had no solid idea, but the motion of driving up to the shelves, move to the side and drive up to the shelves again - well that's restocking!

5.6 Findings

This section will describe how the results of the quantitative data along with the questionnaire and observations evaluates this iteration.

The statistical t-test shows that the null hypotheses can be rejected indicating that there is a difference between using light and gestures in expressing robot behaviours. Furthermore the test between the conditions also shows that there is a statistical difference between using light alone and lights combined with gestures, this result relate to the findings that the gestures alone had no statistical difference with the combination of light and gestures, indicating that the light source, which is only difference within gestures and combination, is not an affecting factor of expressing robot behaviours.

The low mean correct answer percentage for the light condition only clearly shows that this condition was difficult for users to perceive as the only expression type which may be the reason that it didn't affect the combined condition, because the subjects may have relied their answers for the combined condition on the gestures alone resulting in the same mean results for gestures and combined.

Figure 5.10B which shows the individual task types according to the conditions shows a very small mean value for *Vacuum* in the gesture only condition of 13.89%, which shows the subjects had difficulty perceiving the vacuuming tasks through gestures alone. With the opposite high mean value for *Washing* it may indicate that the subject might have been confused between the two tasks and simply answered washing when both where presented. This theory is supported by a answer provided by the subject 6's description of the vacuum tasks: "it was very close to washing - but it didn't have those little jumps - it was more clean". Furthermore the *Vacuum* tasks also has a low mean in the combination condition providing indication of the same evidence.

The light condition only shows that all three tasks has closely the same mean value, shown in figure 5.10B, with a high standard deviation. This results indicates that in the light only condition it may have been difficult to distinguish between the tasks, which is why the answers appear to be random.

The figure 5.13 shows the affect of the order the conditions are presented in. The first session clearly shows a lower mean answer percentage indicating a learning effect throughout the evaluation. This supports how important it is to change the presentation of condition when performing within group tests.

From the questionnaire there is shown a tendency for users to be more confident of the robots expressions in the condition of gesture and combined, and light only scores low results shown in 5.14A.

Figure 5.14B shows a clear tendency of people not being confident about the robots intentions in the condition with gestures only, this is corresponds with the expected results as the robot technically doesn't have any indication of its intention using gestures, only through light. However the light condition was expected to have a higher score for showing the intentions but because the subjects had difficulty perceiving the light it also affected the perception of the robots intention, since it is only expressed through lights. Though observation showed that the subjects tried to analyse a distinct movement behaviour while traveling between tasks, though there where none.

5.7 Discussion

In this project it has been the aim to uncover the use of expressions in semi-autonomous robots. With the use of simple locomotive abilities and two LED-lights it is possible to design a wide variety of movements and expressive light behaviours. Within the right context we believe this combination can be used to design a rich expressive communicative tool between humans and robots.

The platform proposed in this project provides a tool that allows research of the relations between situational context and expressivity in robots more closely. A tangible user interface is a strong tool, that can be used to place a robot in context that users are able to understand and can relate to. By using knowledge of the world through the physicality of objects it is possible to situate a robot that can be associated to real world situations. Miniature models as utilised in this project has been an efficient way of creating a context for users.

This project presented users with three behaviours in the context of a super market. From this it is hard to define specific guidelines for how to design robot behaviours overall but we believe that further research in how users perceive specific movements and light expressions, could open up to a better understanding of how robots can be used to express the internal states of a complex autonomous system. By using expressivity to add a sense of personality to robots, it would be possible for users to form a mental model of autonomous systems, that might prove to be a useful way to communicate the abilities and affordances of an autonomous robot. Since robotic systems are inherently complex systems, it can not be assumed that consumers would be able to comprehend the actual abilities of such a system, and a mental model would improve our opinion and perception of robotic systems.

Many of the robots we will experience entering our homes in the near future, through domestic appliances, toys or educational applications, will pressumably be equipped with some motor abilities and LED-lights as part of their purpose or to be able to notify a user of some internal states such as energy levels. If designed well, these could be used beyond their functionality and act as part of the embodiment of a more user centered design approach.

To more clearly be able to assess this notion of an extended communicative language between humans and robot, there are still room for much more work to be done. This project is proof that expressivity in robots is a complex field that should employ knowledge from several areas of research where cognition, psychology, spatiality, autonomy levels and the use of visual cues all could play a large part of a more general apporoach to effective expressivity in robots.

Chapter 6

Conclusion

The findings of this project shows, that when treating expressivity as lights and gesture separately, we can conclude that gestures are more effective than light, with a statistical significant difference in how well user were able to identify the behaviours proposed in this project.

The combination of the two expressive methods did not show any decrease in effectiveness compared to gestures alone. Some of the feedback from users indicated that they were more confident of what the robot intention with the combination of lights and gestures.

The percentage of correct answers when identifying behaviours through expressitivty we would consider relatively high with above 70%. When going through the feedback from test participants most people were also able to describe the behaviours correctly, according to have the behaviours were designed.

Bibliography

[1] C Breazeal. Social interactions in HRI: the robot view. *Systems, Man, and Cybernetics, Part C: ...*, 34(2):181–186, 2004. URL `http://ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=1291665`.

[2] Jean Scholtz. Theory and evaluation of human robot interactions. *System Sciences, 2003. Proceedings of the 36th ...*, 2003. URL `http://ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=1174284`.

[3] HA Yanco and JL Drury. A taxonomy for human-robot interaction. *...AAAI Fall Symposium on Human-Robot Interaction*, 2002. URL `http://www.aaai.org/Papers/Symposia/Fall/2002/FS-02-03/FS02-03-015.pdf`.

[4] Yvonne Rogers, Helen Sharp, and Jenny Preece. *Interaction design: beyond human-computer interaction.* John Wiley & Sons, 2011.

[5] Donald A Norman. *The design of everyday things.* Basic books, 2002.

[6] Aaron Powers. Feature what robotics can learn from hci. *interactions*, 15(2):67–69, 2008.

[7] Sara Kiesler and Pamela Hinds. Introduction to this special issue on human-robot interaction. *Human–Computer Interaction*, 19(1-2):1–8, 2004.

[8] Eva Hornecker. The role of physicality in tangible and embodied interactions. *Interactions*, 18(2):19, March 2011. ISSN 10725520. doi: 10.1145/1925820.1925826. URL `http://portal.acm.org/citation.cfm?doid=1925820.1925826`.

[9] Orit Shaer and Eva Hornecker. Tangible user interfaces: Past, present, and future directions. *Found. Trends Hum.-Comput. Interact.*, 3(1–2):1–137, January 2010. ISSN 1551-3955. doi: 10.1561/1100000026. URL `http://dx.doi.org.zorac.aub.aau.dk/10.1561/1100000026`.

[10] GW Fitzmaurice. Graspable user interfaces. 1996. URL `http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.120.2035&rep=rep1&type=pdf`.

[11] Orit Shaer. Tangible User Interfaces: Past, Present, and Future Directions. *Foundations and Trends® in Human–Computer Interaction*, 3(1-2):1–137, 2009. ISSN 1551-3955. doi: 10.1561/1100000026. URL http://www.nowpublishers.com/product.aspx?product=HCI&doi=1100000026.

[12] H Ishii. The tangible user interface and its evolution. *Communications of the ACM*, 51(6), 2008. URL http://dl.acm.org/citation.cfm?id=1349034.

[13] Guillaume Zufferey and Patrick Jermann. TinkerSheets: using paper forms to control and visualize tangible simulations. *...Conference on Tangible ...*, pages 377–384, 2009. URL http://dl.acm.org/citation.cfm?id=1517740.

[14] Katrin Wolf and Peter Bennett. Feelable User Interfaces : An Exploration of Non-Visual Tangible User Interfaces.

[15] Peta Wyeth and Helen C. Purchase. Tangible programming elements for young children. *CHI '02 extended abstracts on Human factors in computer systems - CHI '02*, page 774, 2002. doi: 10.1145/506558.506591. URL http://portal.acm.org/citation.cfm?doid=506443.506591.

[16] Bernt Meerbeek, Martin Saerbeck, and Christoph Bartneck. Iterative design process for robots with personality. *...on New Frontiers in Human-Robot ...*, 2009. URL http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.159.8815&rep=rep1&type=pdf.

[17] Bram Hendriks, Bernt Meerbeek, Stella Boess, Steffen Pauws, and Marieke Sonneveld. Robot Vacuum Cleaner Personality and Behavior. *International Journal of Social Robotics*, 3(2):187–195, December 2010. ISSN 1875-4791. doi: 10.1007/s12369-010-0084-5. URL http://link.springer.com/10.1007/s12369-010-0084-5.

[18] Erin Walker and Winslow Burleson. User-Centered design of a teachable robot. *Intelligent Tutoring Systems*, 2012. URL http://link.springer.com/chapter/10.1007/978-3-642-30950-2_30.

[19] Michael S Horn, Erin Treacy Solovey, and Robert JK Jacob. Tangible programming and informal science learning: making tuis work for museums. In *Proceedings of the 7th international conference on Interaction design and children*, pages 194–201. ACM, 2008.

[20] Randy Goebel. *Lecture Notes in Artificial Intelligence Subseries of Lecture Notes in Computer Science LNAI Series Editors*. ISBN 9783642255038.

[21] J Siekmann and W Wahlster. *Lecture Notes in Artificial Intelligence*. ISBN 9783642172472.

[22] Jacob Nielsen. Why you only need to test with 5 users, 2000. URL http://www.nngroup.com/articles/why-you-only-need-to-test-with-5-users/.

[23] CW Reynolds. Steering behaviors for autonomous characters. *Game developers conference*, 1999. URL http://www.red3d.com/cwr/steer/gdc99/.

[24] Chris Harrison, John Horstman, Gary Hsieh, and Scott E Hudson. Unlocking the Expressivity of Point Lights. 2012.

[25] Kwan Min Lee, Wei Peng, Seung-A Jin, and Chang Yan. Can Robots Manifest Personality?: An Empirical Test of Personality Recognition, Social Responses, and Social Presence in Human?Robot Interaction. *Journal of Communication*, 56(4): 754–772, December 2006. ISSN 0021-9916. doi: 10.1111/j.1460-2466.2006.00318.x. URL http://doi.wiley.com/10.1111/j.1460-2466.2006.00318.x.

List of Figures

List of Tables

Appendix A

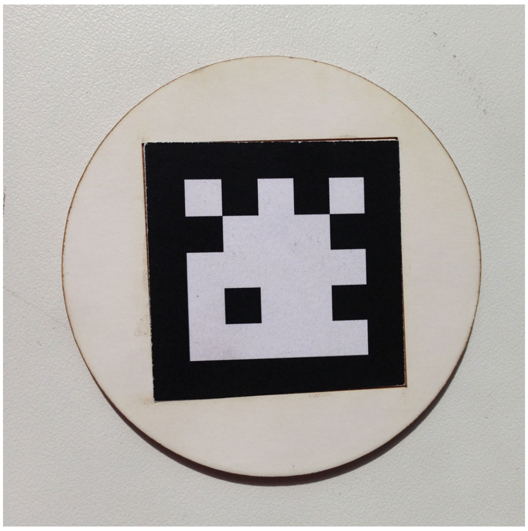
Questionnaire Iteration 1

To what extend did the expression of the agent influence your estimation of the solution?

	1	2	3	4	5	
Not at all	○	○	○	○	○	Completely

To what extend did the expression of the agent meet your expectations of the outcome?

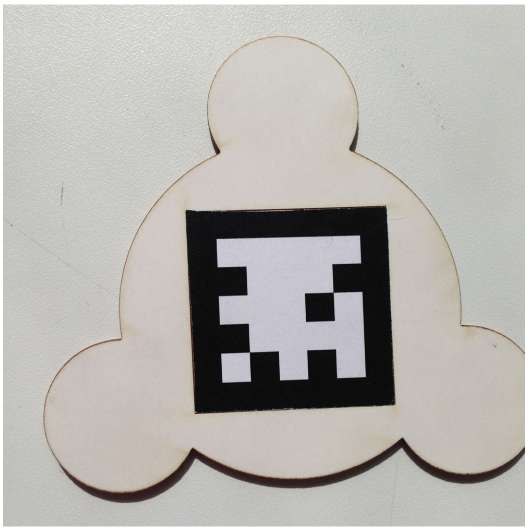
	1	2	3	4	5	
Not at all	○	○	○	○	○	Completely



How confident are you that you understood the functions of the first guide?

	1	2	3	4	5	
Not at all	○	○	○	○	○	Completely

How would you describe its function?



How confident are you that you understood the functions of the second guide?

	1	2	3	4	5	
Not at all	○	○	○	○	○	Completely

How would you describe its function?

To what extend would you describe the problem solving as a cooperation between you and the agent?

	1	2	3	4	5	
Not at all	○	○	○	○	○	Completely

Questionnarie Descriptions of the Guides

Table A.1: Tables of subjects description of the guides for iteration 1.

Subject	Describtion of first guide	Describtion of second guide
1	Reach the objektive in a straight Line	Reach the objective avoiding obstacles
2	Easy quite straight forward	Complicated to understand at first
3	Liniear	Bouncing
4	Attractor	Multi-path guide
5	Connects the smiley with other guide directly	Does the same as the first agent plus passes the obstacles
6	Go to brik	Pathfinding brik

Appendix B

Questionnaire Iteration 2

Demographic Questions

Age:

Gender:

- ○ Male
- ○ Female

Sessions Questions

How confident are you that you understood the intentions of the robot? (What it was about to do)

	1	2	3	4	5	
Not at all	○	○	○	○	○	Completely

How confident are you that you understood what it was doing? (What action it was performing)

	1	2	3	4	5	
Not at all	○	○	○	○	○	Completely

repeated for all three sessions.

End Questions

Below please describe how you interpreted the actions of the robot (the actions it was performing)

Please describe for washing

Please describe for vacuuming

Please describe for restocking

Questionnarie Descriptions of the behvaviours

Table B.1: Tables of subjects description of the behvaiours for iteration 2 evaluation.

Subject	Wash	Vacuum	Restock
1	If the robot was lighting blue it was washing, if light was lighting white it was vaccuming, and if the robot was lighting yellow it was restocking	moving in a straight direction and then back and forth	moving in a straight direction and then moving towards the shelf and back and then again towards the shelf and back
2	side to side movement and blue lights	back and forth movement and purple lights	orange light and driving back and forth to different places on the shelfs
3	Rubbing. Circular motion.	Back-forth. like a normal vacuum cleaner.	Starts at one end of the shelf and woks its way through.
4	Forward, backward movements with different vector angle	Forward, backward movements within the same vector	Movements describing moving object from one place to another, also helped the fact that the robot was facing the shelfs
5	rotation , blue lights	movements back and forth,	complex movement covering a bigger surface than the other actions, orange lights
6	Doing little random jumps - also orange lights.	it was very close to washing - but it didn't have those little jumps - it was more "clean"	It turned towards the obstecales and looked a lot like it was stocking up. think it was orange lights

Continued on next page

Table B.1 – continued from previous page

Subject	Wash	Vacuum	Restock
7	washing was represented by a rotating movement. short and quick. pivotal movement, same axis	vacuuming was short and repetitive back and forth, slight rotation after each pair of forward/backward pattern	the robot rotated to face the shelf, it would move forward to get closer to it and back then move to a contiguous section and repeat. Usually three times.
8	small semi circular movements on the spot	going back and forward on the spot tipping	starting at a corner of the shelf and then move in rectangular movments along the shelf aprox 3 times
9	når den lyst blåt og den "twistede" var det tydeligt at den ville vaske gulv.	Når den så lyste hvidt og lige køre lidt frem og tilbage, ville den støvsuge.	Når den lyste gul og kørte ind mod hylderne, ville den sætte på plads.
10	It was scrubbing the floor.	Moving back and forward like you do with the vacuum nozzle.	It put things on different shelves by moving around and back and forth.
11	rotating around it's axis a bit	going back and forth a few times	moving 3 times at the same shelve: right, middle, left
12	blue light and back and forth movement.	not sure. different light and forward and back movement.	yellow light and moving to and from the shelves.
13	The first session was with lights only and I thought that blue light would equal water/washing. Later when I had seen movements, I felt confident that the twisting motion was a washing action - and it matched with blue lights :)	It was hard to tel just from the lights color, but when I saw the forward/backward motion it reminded me of vaccuming.	From the lights I had no solid idea, but the motion of driving up to the shelves, move to the side and drive up to the shelves again - well that's restocking!

Test advertising

The advertising was done through social media and contained the following message and a doodle sign-up form.

```
SARA #1 Testing - Robot Expressivity
https://www.youtube.com/watch?v=xMY7hCp7q3c
Our Robot SARA has been told to clean the Supermarket
but she needs your help completing the task!! - MAX 20min
There are no requirements for participating and any help
is much appreciated. So please feel the urge to come help us
with our Master Thesis in Interaction Design
Location: FKJ12 - Ground floor, E-lab room 013

Best
Niels Peter Rasmussen & Jannik Jepsen
jjepse09@gmail.com
```

Appendix C

Code samples

This section shows sample of the programming code used for implementing the system. The full source code material for both iterations can be found on the CD, see Appendix D.1

C.1 Marker Transformation Sample

```
//Function for converting tracked corners of the marker to a centered position on the surface
void SetPosition(PVector[] v){
  println("Detected New Position for marker ID: " + ID);
  Vertex = v;
  PVector ScreenPos = new PVector();
  //Findin the average position of the 4 corners, center of the marker
  for(int j = 0; j < 4; j++){
     ScreenPos.add(Vertex[j]);
  }
  ScreenPos.mult(0.25);

  //Transforming the position to the surface
  keyPos = TrackingSurface.getTransformedCursor((int)ScreenPos.x,(int)ScreenPos.y);
  PVector corner1 = TrackingSurface.getTransformedCursor((int)Vertex[0].x,(int)Vertex[0].y);
  PVector corner2 = TrackingSurface.getTransformedCursor((int)Vertex[1].x,(int)Vertex[1].y);
  PVector dir = PVector.sub(corner2,corner1);
  Rot = dir.heading();
  WritePosition(); //Sends the position through the gateway to the main controller
}
```

C.2 Objectives Generator Algorithm

```
//Function for generating objectives for the second iteration
void GenerateObjectives() {
  if (Pause) //if the program is paused discontinue
    return;

  if (millis()-ObjTimer < 15000) //wait more appr. 15 seconds +- random 3 sec
    return;

  if (CurrentBoard != null && CurrentBoard.Rosa != null) {
    float r = random(100); //random value for selecting objectives
    if (r > 80) {
      //load restock position from the map, necessary for stocking the shelves in the correct maner
      println("No Of STOCKPOINTS: " + CurrentBoard.Map.getChild("STOCKPOINTS").getChildCount());
      int rS = (int)random(0, CurrentBoard.Map.getChild("STOCKPOINTS").getChildCount());
      println("Random Stock: " + rS);
      ArrayList<PVector> temp = new ArrayList<PVector>();
      println("No of child Stock points: " + CurrentBoard.Map.getChild("STOCKPOINTS").getChild(rS).getChildCount());
      for (int i = 0; i < CurrentBoard.Map.getChild("STOCKPOINTS").getChild(rS).getChildCount(); i++) {
        temp.add(CurrentBoard.Map.getChild("STOCKPOINTS").getChild(rS).getChild(i).getChild(1).getVertex(0));
        temp.add(CurrentBoard.Map.getChild("STOCKPOINTS").getChild(rS).getChild(i).getChild(0).getVertex(0));
      }
      CurrentBoard.Rosa.NewObjective(new Objective(temp)); //Add the loaded restock objective to the board
    }
    else {
      PVector pos = new PVector(random(1220), random(1500)); //random position on the surface
      for (Obstacle ob : CurrentBoard.Obstacles) { // check for any collision with the static objects, as no objectives should be placed on the shelves
        Rectangle2D bounds = new Rectangle2D.Float(pos.x-50, pos.y-50, 100, 100);
        if (ob.Collider.intersects(bounds))
          return;
      }
      if (r > 33)
```

```
        CurrentBoard.Rosa.NewObjective(new Objective(pos, 0.0, 19, ObjectiveType.WATER)); //Create water objective on the random position not colliding with the shelves
      else
        CurrentBoard.Rosa.NewObjective(new Objective(pos, 0.0, 19, ObjectiveType.VACUUM));//Create Vacuum objective
    }
    ObjTimer = millis()+random(3000); //Set the timer between the generation of objectives with a deviation of 3 seconds
  }
}
```

C.3 Robot Gateway

```
//Function for sending motor values over network to the robot
void WriteMotorValues(float[] Power) {
  if (millis()-SendMotorDelay < 100) //delay of 100ms
    return;
  SendMotorDelay = millis(); //reset delay
  Stopped = false; //ensure that the robot functionalities are not stopped
  if (RClient != null && RClient.active()) { //check for connection with the robot
    if(NetDebug)
      println("Writing Motor: 0," + Power[0] + "," + Power[1] + ","); //debug
    RClient.write("0," + Power[0] + "," + Power[1] + ","); //write motor values to the robot client
  }
  else{
    if(NetDebug)
      println("No connection to the robot"); //debug
  }
}

//function for writing color values to the robot including light state to be used, hue of each diode and saturation if needed
void WriteColorStateHueAndSat(int state, int hue1, int hue2, int sat) {
   if (millis()-SendColorDelay < 500 || TestState == 0) //delay and check of evaluation test is condition without light then break.
    return;
  SendColorDelay = millis(); //reset delay timer
  if (RClient != null && RClient.active()) { //check for connection
    if(NetDebug)
      println("Writing: 4," + state + "," + hue1 + "," + hue2 + "," + sat + ","); //debug
```

```
    RClient.write("4," + state + "," + hue1 + "," + hue2 + "," + sat + ","); //write values to the robot client
  }
  else{
    if(NetDebug)
      println("No connection to the robot"); //debug
  }
}
```

C.4 Dijkstra Path finding algorithm

```
//Dijkstra Pathfinding Algorithm
ArrayList<Node> open = new ArrayList<Node>(); //set for all non checked node in the map connected to the path
ArrayList<Node> closed = new ArrayList<Node>(); //set for all the checked nodes which should no longer be considered

open.add(StartNode);
Node current = StartNode;
while (true) { //Continuesly check nodes until the goal has been found
  //Find the node with the lowest possible accumulated distance and check that first
  min = Float.POSITIVE_INFINITY;
  Node next = new Node();
  for (int i = 0; i < open.size() ;i++) {
    Node temp = open.get(i);
    if (current.AccuDist < min) {
      min = current.AccuDist;
      next = temp;
    }
  }
  current = next;

  if (current == EndNode) //The current node being checked is the end goal, finnish!
    break;

  //Collect all connected nodes which is not already checked (closed) and put them in the open list and calculate accumulated distance along with memorizing the connections previous for backtracking
  for (int i = 0; i < current.Connections.size(); i++) {
    Node temp =  current.Connections.get(i);
    if (closed.contains(temp))
      continue;

    float accuDist = current.AccuDist+current.Pos.dist(temp.Pos);
```

```
    if (temp.AccuDist > accuDist) { //Check if the accumulated distance for the node is not already smaller than the current, meaning if it has been checked before and found closer
      temp.AccuDist = accuDist;
      temp.Previous = current;
      open.add(temp); //add to the open list for checking
    }
  }

  closed.add(current);
  open.remove(current);
}
//backtracking the found path by connecting the end nodes previous in a continues loop
Node BackTrack = EndNode;
while (true) {
  if (BackTrack == StartNode)
    break;

  Node temp = BackTrack.Previous;
  temp.Next = BackTrack; //Set the nodes Next path connection as the backtracked node
  BackTrack = temp;
}
}
```

C.5 Light Algorithms

```
//Work light Algorithm
void Work(){
  if(millis() > StateTimer+200){//200ms
    Colors[2] = 0; //Sets brightness value of first Diode
    Colors[5] = 0.5; //Sets brightness value of second Diode
  }
  if(millis() > StateTimer+400){//for the next 200ms up to 400
    Colors[2] = 0.5; //Sets brightness value of first Diode
    Colors[5] = 0; //Sets brightness value of second Diode
    StateTimer = millis();
  }
}

//Pulse Light Algorithm
void Pulse(){
  Count += 0.1; //increase value
  float brightness = (1+sin(Count))/4; //Calculate brightness by an offset sinuswave
```

```
  //Apply brightness to both Diodes
  Colors[2] = brightness;
  Colors[5] = brightness;
}

//Beacon light algorithm
void Beacon(){
  if(millis() > StateTimer+100){//on for 100ms
    Colors[2] = 0.5;
    Colors[5] = 0.5;
  }
  if(millis() > StateTimer+200){//of for 100ms
    Colors[2] = 0;
    Colors[5] = 0;
  }
  if(millis() > StateTimer+300){//on for 100ms
    Colors[2] = 0.5;
    Colors[5] = 0.5;
  }
  if(millis() > StateTimer+400){//of for 100ms
    Colors[2] = 0;
    Colors[5] = 0;
  }
  if(millis() > StateTimer+1000){//of for 500ms
    StateTimer = millis();
  }
}
```

Appendix D

CD Content

D.1 Source Code

All source code be located on the CD in folder "\SourceCode" All source code is prorammed using Processing IDE, it is therefore recommend using the same editor but any text editor will show the content. In total there is approximatly 3000 lines of code for the second iteration

D.2 Matlab

All data processing is handle in Matlab where source code can be located along with necessary log files on the CD in folder "\Matlab"

D.3 Video Material

A selection of videos of the robot sara is included in the CD folder "\Video"

File name	Description
restocking.MOV	A clip showing the restocking behaviour *restocking*
vacuumin.MOV	A clip showing the restocking behaviour *vacuuming*
washing.MOV	A clip showing the restocking behaviour *washing*

D.4 Design Files

All design files and illustrations used in the system can be found in folder "\DesignFiles"
Folder Content:

- SARA.ai - designfile for robot
- Hylder.ai - markers and shelves