Towards and Autonomous Mobile Control Unit of Smart Homes

by

Haidong Wang

Thesis submitted to the
Faculty of Graduate and Postdoctoral Studies
In partial fulfillment of the requirements
For the M.A.Sc degree in
Electrical and Computer Engineering

Ottawa-Carleton Institute for Electrical and Computer Engineering
School of Electrical Engineering and Computer Science
Faculty of Engineering
University of Ottawa

Abstract

Since smart homes will be more and more popular in the future, the needs of finding a friendly, comfortable and smart interface between users and home environments are growing. However, conventional interfaces have reached their limits. So, this paper describes a new concept of interface in a home environment. This work is distinguished by building a separable smart robot. A smart phone is used as a brain, and a robot car is used as a body. With the system, people will be allowed to carry the robot's brain when they are out of their home. In this case, the brain can not only be a normal smart phone, but also can monitor parameters such as the temperature at home. On the other hand, when people are at home, they will be allowed to put their phones on the robot car. By invoking the system, they will have an assist-robot. This assist-robot will act as a new interface in the smart home by following people and recognizing their voice commands in order to take notes, to read notes and to control smart home devices. This thesis also implements face recognition to improve the communication between users and the interface. Test results show that our proposed system is well accepted.

Acknowledgements

I would like to give my sincerest appreciation to my supervisor Prof. Abdulmotaleb El Saddik for his continuous guidance and support not only in academic level but also in my life.

Special and sincere thanks go to Dr. Jamal Saboune for the invaluable assistance, guidance, and feedback he provided throughout my research. I am thankful to Bassem Barake, who developed the Home Automation web service. Also, I would also like to thank all colleagues in DISCOVER and MCRLab for their contributions throughout the research.

I am grateful to my roommates and all my friends for their comments, encouragement and help during this challenging time.

Finally, I would like to express my deepest thanks to my parents and my wife. Their inspiration, understanding and support made this thesis possible. This work is dedicated to them.

Contents

List of Tables

List of Figures

Glossary of Terms

2D Two-dimensional

3D Three-dimensional

API Application Programming Interface

Eigenfaces A set of eigenvectors used in the computer vision problem of human face recognition

HOG Histogram of Oriented Gradient

HSV A color model which uses Hue, Saturation and Value to reproduce a broad array of colors

IP Internet protocol

LCD Liquid Crystal Display

OpenCV Open Source Computer Vision

PC Personal Computer

RGB A color model which uses red, green and blue light to reproduce a broad array of colors

SMS Short Message Service

Socket An endpoint of an inter-process communication flow across a computer network

TCP Transmission Control Protocol

WIFI A popular technology that allows an electronic device to exchange data wirelessly over a computer network.

WLAN wireless local area network

YUV A color space typically used as part of a color image pipeline

Chapter 1

Introduction

1.1 Background and Motivation

The home acts as an important role in people's lives. After a whole day of hard work, people are exhausted when they arrive at home. People may find that they are too tired to move their steps while sitting on sofa or lying in bed. So any small tools that helps people turn lights on or off, or play their favorite music may make their home more comfortable. Moreover, it would be better if everything such as warming bath water and adjusting room temperature was already done before people reach their home. So, when people would arrive home, they would find everything has been adjusted to their suitable preference, and they could take their shower immediately.

Human assistants like housekeepers were a way for millionaires to keep up their homes in the past. Perhaps this is still a solution for the rich in some societies. However, not everyone is wealthy enough to be able to afford a human assistant. Hence, the need for finding an inexpensive and smart assistant for normal families keeps growing.

As today's technology develops, many automatic products and concepts have already come true and can be used at home. Moreover, many of these technologies have the same purpose, which is to reduce people's work and create a smart environment where people are able to have better, easier, and more comfortable life.

According to [30], the smart home environment refers to a home where devices are smart enough to acquire some information from the surroundings in people's lives and work, and use collected information to serve people. There are several core technologies for studying smart homes according to [30] among others:

- Technologies of sensors and actuators.

- Communication technology.
- management technology.
- Interface technology

Nowadays, sensors and actuators are widely accepted in our everyday lives. Some of them are already being used in home environments. For example, auto systems like auto-doors which contain infrared sensors to detect people and actuators to open the door are widely available. Moreover, as they are not expensive, many people may prefer to have one to replace traditional handle doors at home.

Recently, vacuum cleaning robot such as LG Roboking [9] and Samsung NaviBot [16] were introduced into the market. These robot contain sensors to detect places to clean, and by using actuators they can move and do the clean work like an assistant at home.

Figure 1.1: LG Roboking and Samsung Navibot

As smart home environments develop, perhaps in some cases, an actuator is invoked by several sensors in different places. In other cases, actuators need to be controlled by users through interfaces. Therefore, communication need to be set up between sensors and actuators or sensors and interfaces. Communication technologies are aimed to set up a reliable, high speed and low energy consumption networks in smart homes. These communication technologies act as bridges between smart home devices and interfaces. PLC (Power Line Communication) technologies [53] and Low-rate wireless network technologies are the two widely accept networking technologies in smart homes.

The Low-rate wireless network technologies include WLAN, Bluetooth [27], ZigBee [20], Z-wave [19] and others. These wireless technologies are associated with low speed, low power consumption and flexibility in networking of smart homes.

Interfaces and management technologies in smart home environments are designed to help controlling the environment. The management technology manages and keeps the collected information from sensors, and interfaces facilitate these collection of sensory data. In this way, a person can easily understand her living information, and can control the environment conveniently.

Devices such as PDAs and smart phones are used as an interface in a smart home environment.

For example, a PDA was used as the interface in the home of the future by the House_n group [36] at the Massachusetts Institute of Technology (MIT). Information from different sensors was well managed by the PDA, and it also offered some preset options for people to control actuators in the smart home.

Moreover, recent development in information technology, the Internet and web services (clouds) offer a big advantage for the design of Interfaces today. The Internet allows users to remotely monitor their home information and control their home devices through smart phone while away. Web services, which is a cloud computing technology, can be an interface themselves. The advantage of using web services is that they can be remotely controlled by other devices through the Internet.

However, based on the best of my knowledge, the limits of computers or smart phones in smart environment are as follows:

- Computers today are still too big for people to carry everywhere, but people may want to use smart home devices or retrieve home's information at any location and time.
- Smart phones indeed made the interface smaller and more feasible to carry when not at home. However, after arriving at home, it is not comfortable for people to keep carrying the smart phones. Thus, people may put smart phones or tablets anywhere. This means people may need to look for them every time before using them.

Human Assist Robots appear to be an answer to these problems. Long time ago, robots were dreamed to coexist with humans and to help them achieve better lives. The author in [28] provided an idea for a robot in a smart home environment. He suggested that robots should be designed as simple as possible instead of being well programmed to act as humans. For example if a robot wants to turn on a light, it would send a turn on light request to the light actuator, but not use its hand to turn on it like a human [28].

In order to analyze the possibility of using robots as interfaces in smart homes. The authors in [51] evaluated a human robot NAO [14] (Fig 1.2) which ates as an interface and live together with elderlies in their KSERA system. The KSERA system is a smart home, which consists of many smart sensors, a KSERA server and a Robot NAO.

The result from the work shows that the usage of robot NAO is highly rated while living with elderly as an interface. Especially in mindfulness, happiness, friendliness and confidence. The reason as analyzed in [51] is that human robots like NAO will make people feel safe and without fear, it also can communicate with people through voice which make them feel more comfortable. In addition, human robots offers a huge potential to act and think like human in the future.

Figure 1.2: Robot Nao

However, robots today have some limitations:

- Robots assistant are still expensive.
- While using robot assistants as interfaces, the available range of this kind of inter-

faces is limited at home, because it is not convenient for people to bring a robot when they leave their home.

Base on the above discussion of the robot and smart phone technologies, it seems that a smart phone is easy to carry, but it would be better if a smart phone can follow people, and move like a robot, while a robot is too expensive, and seems unfeasible to carry today.

Coincidently, this problem of smart phones and robots is just like laptops and tablets. Tablets such as IPAD, NEXUS 10 are more convenient to carry than laptops. On the other hand, in many situations, tablets still cannot replace computers. One main reason is the feeling of typing on a tablet does not suit some uses. Fortunately, Surface which was recently released by Microsoft offers a solution. Surface [17] is a composite laptop which contains a tablet and a keyboard called type-cover. In this way, it can act as a laptop and without a keyboard it turns to a tablet.

The Surface really gave us the idea, that we can build a brain and body separable robot, which uses a smart phone as its brain, and a mobile WIFI robot as its body.

In addition, Some functions on smart phones could be used in our system. Notepad is one of the most needed functions.

Many things happen in our daily lives and work. In fact, we have to realize that sometimes we do have to record things for future reference, so that we do not risk forgetting something which may be important. These things we should remember may be as small as what we planned to buy on our way home, or as big as an important business meeting which will be held tomorrow. The time range of these things could be from minutes to weeks, and the frequency could be just one time or everyday.

We have been using paper and pens to record our tasks for hundreds of years, and today it is still a good idea to write down what to do next on paper and put these notes in the place we can easily notice. However, we still have chances to forget or to lose these notes. Also, organizing the order of tasks based on importance and deadlines is a time-consuming work. Disarranging tasks and losing notes may lead to forget important tasks when the time has already passed.

Therefore, thanks to the development of the Internet, computers and Smart phones, many reliable notepads were designed for smart phones and computers.

Google Calendar [5] is a good example of today's popular notepads. Google Calendar is a free web application released by Google. As Android OS is widely accepted today, the number of users who are using Google Calendar is increasing. Every note a user takes on the computer or smart phone is synchronized to the appropriate Google Account, so that it can be accessible every where.

1.2 Objective and Contribution

Our objective is to make composite robots act as interfaces in smart home environments by using smart phones and mobile Robots. In fact, a smart phone has good processing power, and containing many sensors such as cameras and microphones. The only problem is that a smart phone lacks self mobility, so that our idea is to improve it by adding the robot to it that is to give it wheels.

The main contributions of this research work is summarized in the following points:

- **Design and development of an potential interface in smart home environments:** As listed before, in smart home environments, there are many limits and problems of smart phones and robots to act as interfaces. Therefore, in order to solve these problems, we developed and designed a composite robot as the potential interface in a smart home. This robot consists of an Android smart phone combined to WIFI robot car. When a person is not at home, she can receive information such as room temperature and controls home devices such as lights through her smart phone. When a person is at home, she can put her phone on a mobile robot car. This robot assistant can understand the user's voice-command to take personal notes, voice control smart home devices, and use the video feed provided by the smart phone's camera to detect and to follow the user. In order to improve the security and personalization of the robot, Eigenface [54] is used in a web service to recognize users. This software/hardware solution would solve the high cost and portability problems of a robot and the lack of self movement problem of a smart phone. Moreover, smart phones are the kind of devices that almost all of us own and are familiar to use.

  As in our system, understanding a user's voice-command, controlling smart home devices and taking a note for the user are using different web services. A traditional way to invoke different web services is to press different buttons in smart phone's application. However, some of these web services is unfeasible to be invoked as traditional method in our system. Therefore, we developed and designed our composite robot to invoke different web services through the user's voice command.

Moreover, smart phones are the kind of devices that we almost all own and are familiar to use, and the robot's body just needs to receive decisions from the smart phone, which can be simply designed. Thus, the cost of the whole system would be inexpensive, and could be mass production. Therefore, in our idea future, people just need to bring their smart phones, and their smart phones can become a robot assistant wherever a body can be found. This is a brand new vision in robotic and smart phones.

- **Designed and development of an algorithm to follow a person visually:** we successfully allow our composite robot to follow a person by using the smart phone's camera. We designed an algorithm to detect the direction that a person is walking according to the person's position in each frame captured by the smart phone's camera. Then, the smart phone controls the robot car to follow the person.
- **Developed the chip of our system's robot car to be a real-time people follower:** We developed the chip of the robot car in order to make it follow a person in real time. After detecting a person and calculating the person's position, the smart phone would send control commands to the chip of the robot car, and the chip would control the wheels of the robot car to follow the person according to the commands.
- **Development of a method to let a smart phone be ware of when the user want to command:** Today, many voice recognition services, such as Google voice search and SIRI, allow smart phones to understand voice-command. However, most of them require people to press a button to initialize the voice recognition service, which it is impractical for our robot. Therefore, we developed a method to let the smart phone be ware of when it should invoke the Google Voice RecognizerIntent [13] and wait for commands.

1.3 Thesis Organization

This Thesis is organized as follows:

- **Chapter 2:** This chapter presents a review of the literature about all the related works. It describes the development of smart home, different kinds of interface and some evaluation in smart homes. It also describes OpenCV, a image processing solution, and some methods for face detection and people detection.

- **Chapter 3:** In this chapter, we discuss our contribution in detail. Details and explanation of the composite robot, method to invoke Google Voice RecognizerIntent and other web services, our proposed smart home interface, and algorithms and methods for our system to follow people can be found.

- **Chapter 4:** In chapter 4, the experimental results are exposed in this chapter. We also describe the experimental hardware and software setting in our composite robot system.

- **Chapter 5:** In the final chapter, we summarize the contributions and progress in this research with a conclusion. We also introduce some future work that can improve this system.

Chapter 2

Literature Review

2.1 Previous Smart Home

Over the past few decades, many research groups and companies have been looking forward to building different smart home environments. They want to create smart home environments that are convenient and where people can feel more comfortable.

In the followings, we will discuss some related smart homes.

2.1.1 Aware Home

In 1999, the Aware Home [39] was built at Georgia Institute of Technology. This experimental home is a smart room filled with context-aware devices. For example, the researchers built vision-based sensors that can track people, as well as a smart floor to track them in larger spaces. A wearable computer was used to manage data in this smart home. In brief, they were trying to make it possible for the home to understand information from its inhabitants and contextual information from itself. Depending on the information collected, intelligent tasks would be activate automatically, creating a comfortable living area.

This home also functions as a living laboratory for ubiquitous computing research.

The computer used as the interface in the Aware Home can manage data, however, this kind of traditional computer is not convenient to carry when leaving the home.

2.1.2 MavHome

Stepping into 21 century, the University of Texas at Arlington conducted MavHome project (Managing an Intelligent Versatile Home) [32]. This project aims to create a smart environment that acts as an intelligent home agent.

The MavHome would perceive states of the home through its environment sensors, and intelligently activate the following responses through device controllers. For example, at 6:45 a.m. every morning, the MavHome increase the heat to optimal temperature for the occupants to wake up. It can predict the chance of rain by using its sensor and inform people before they leave. It can also close the windows automatically through its actuators.

Moreover, in order to meet goals of efficiency, the MavHome could also predict the inhabitant's future actions based on prediction algorithms. The author presents the Active-Lezi algorithm in order to do this. Moreover, the prediction algorithm also provides information to the decision-making component of the MavHome. The goal of the decision-making component is to enable the home to automatically control some basic functions itself.

The architecture of MavHome is illustrated in Figure 2.1.

The problem with the MavHome is that although the home can be an intelligent agent by using a prediction algorithm, it does not have a good interface to let a person control devices and read information directly.

2.1.3 Ubiquitous Home

Tatsuya Yamazaki [52] built a Ubiquitous Home equipped with sensors everywhere. Cameras, microphones and speakers were added to the ceilings of every room (Figure 2.2-left), which can be used interact with a computer server. Pressure sensors are embedded under the floor (Figure 2.2-right), using binary detection units to track residents and help detect furniture positions. Infrared sensors are placed on top of doors in order to detect human movement and determine which room they are in. RFID sensors are used to check forgotten property when people leave the home. All the sensors and actuators are connected to a network.

By using these technologies, the author introduced three kinds of context-aware services in the Ubiquitous smart home: TV programming recommendations, recipe showing, and forgotten property check.

Besides that, the author introduced Phyno, a dialogue-based interface robot with

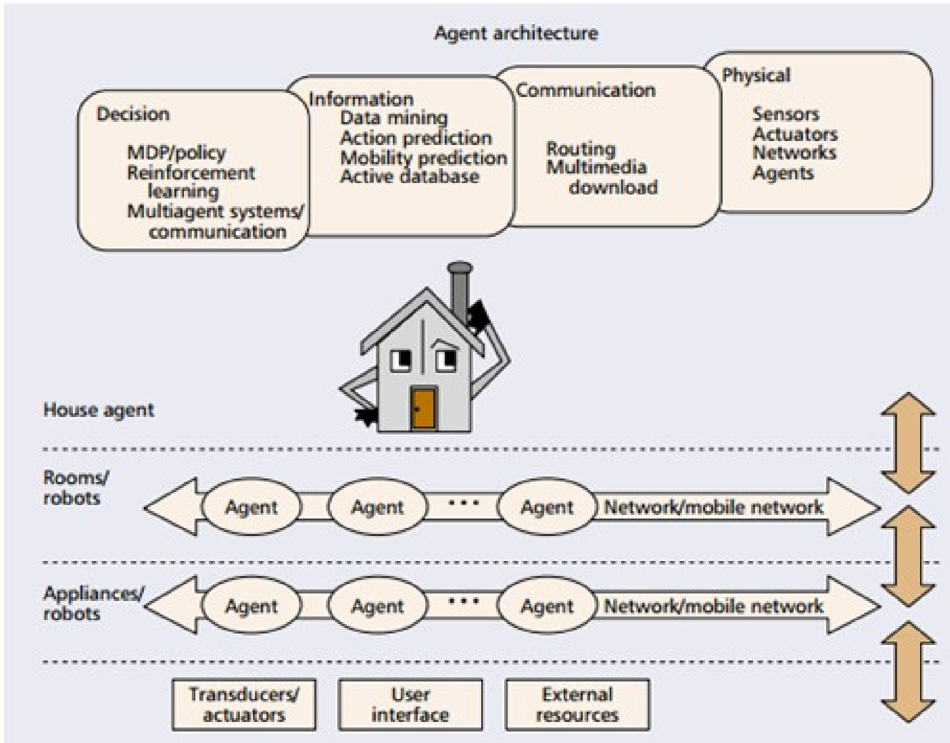


Figure 2.1: The Mavhome

face recognition functions as the interface. However, it was just a concept in the author's mind and did not, in reality, control a smart home.

2.1.4 MIT Home of the Future

At the Massachusetts Institute of Technology (MIT), the House_n group [36] was trying to design a home of the future. They believe that the home of the future would help people adjust the environment instead of automatically controlling devices. This means the home just needs to collect the environmental information from its sensors, and use its programmed algorithms to provide suggestions in order to help people control smart home devices through its interfaces.

In this project, different kinds of networked sensors and actuators were integrated in different places. And a wearable PDA was used as the interface. Figure 2.3 provides examples of how this home works. From the figure, we know that the home can collect information from its temperature sensors and wind strength sensors, and run the

Figure 2.2: The Ubiquitous Home

programmed algorithms to make a forecast. Thus, suggestions like "open the window now?"can be made by the home itself and sent to the user's PDA's screen to help her make the most appropriate decisions.

A PDA improves the interface by making it smaller and portable for when a person leaves her home. However, it is not comfortable and convenient for a person to hold when she is at home.

2.1.5 Voice-based Smart Homes

David Liu and Dao Xian discussed a home environmental control system based on X-10 devices, self-designed and developed software, and tablet PC in [41]. X-10 [18] is a protocol in PLC technology and act as a communication technology in smart home. Software as a management technology was developed under visual studio.net and could be installed and run in any PC with a Windows OS. Then, a tablet PC is a friendly user interface in this smart home. Figure 2.5 is the diagram of the Home environment control system. Devices such as table lamps, a radio and a TV in room 1, as well as devices such as water heater and air conditioner in room 2 can be controlled through the PC and tablet PC interfaces.

Moreover, it is embedded a voice recognizer by using Dragon NaturallySpeaking 8.0 [12], so that users can control their home environments including lights, TV, radio and

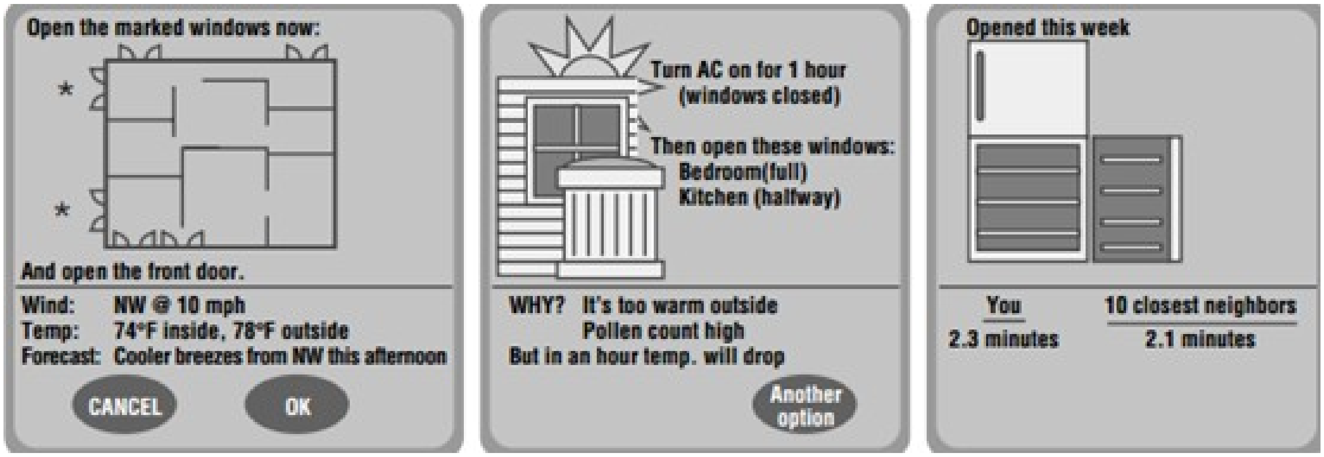


Figure 2.3: PDA as an interface in the home of future

VCD/DVD players by using standard mouse or their voice. Figure 2.4 illustrates the operation flow chart.

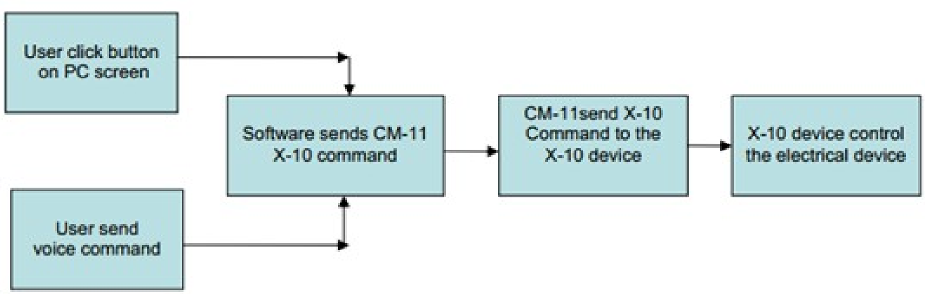


Figure 2.4: The flow chart of the system

2.1.6 Web Service based Smart Homes

The researchers at the University of Ottawa designed and developed a web service based smart home environment in [45]. Web services is a cloud computing technology, which was claimed to be a central role in future smart homes in [21]. In the web service based smart home environment, different devices are allowed to remote access the web service and control the smart home.

Figure 2.6 shows some actions and events provided by the web service.

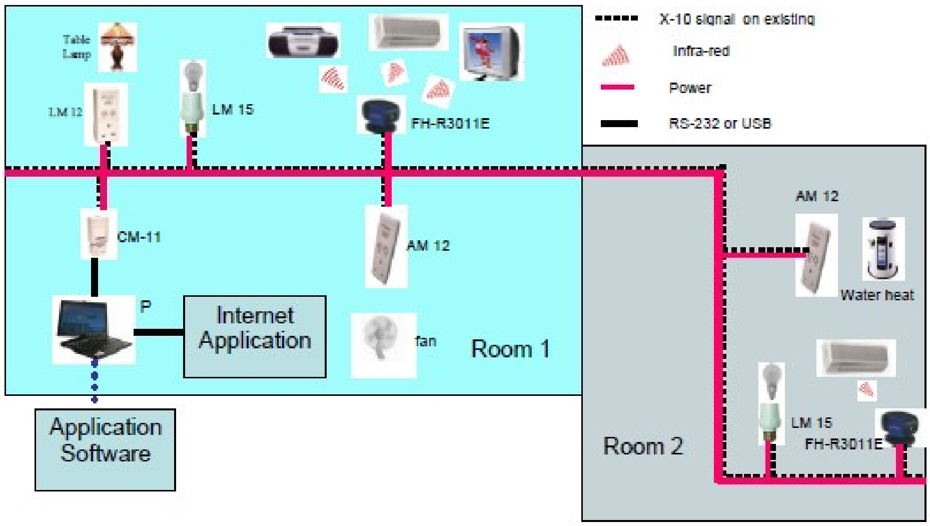


Figure 2.5: Home environment control system

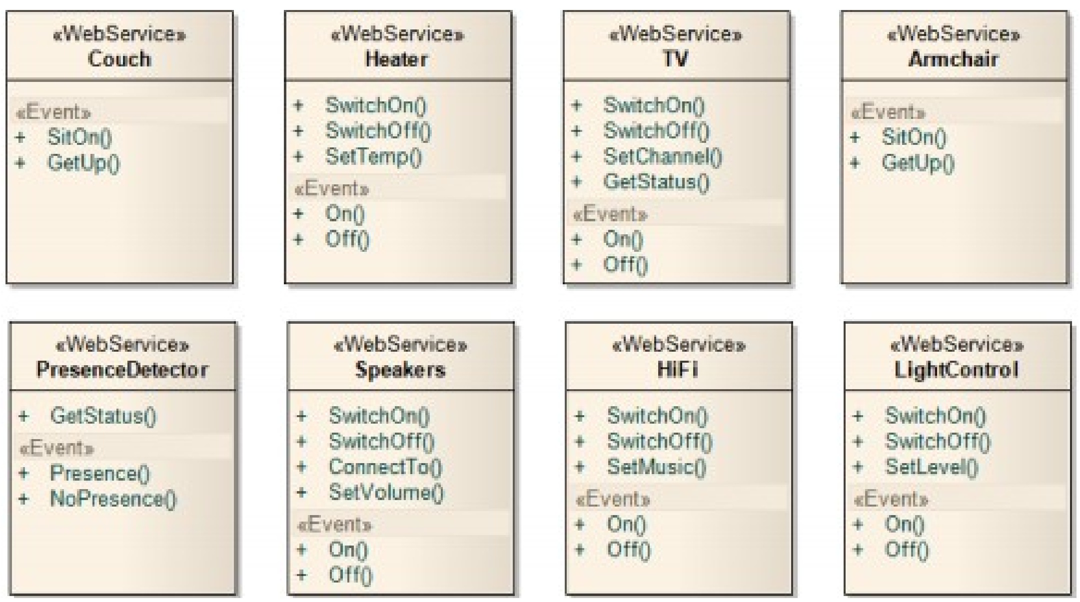


Figure 2.6: Actions and events of the web service

2.2 Integrating Robots into Smart Environments

In many science fictions, there are always huge paragraphs that describe how smart an assistive robot is, how friendly an assistive robot is, and how convenient people's lives can be when robot assists them. Assistive robots able to coexist with humans and help them achieve better lives have been considered to appear in the future scenario of our everyday lives for a long time.

Many research groups are keen on developing robots, and many of them are trying to integrate robots into smart home environments.

The followings are some previous works on the subject:

2.2.1 PEIS Ecology

Mathias Broxvall, Marco Gritti and Alessandro Saffiotti in [28] introduce the concept of the Ecology of Physically Embedded Intelligent Systems, or PEIS-Ecology. This is a work that explores a new road to integrating robot devices into smart environments. The main approach of the PEIS-Ecology is that the PEIS-Ecology can be used to overcome many hard problems of autonomous robotics, and it allows robots to be integrated in current smart home technologies.

The concept of PEIS-Ecology is as follows. Firstly, any robot in the smart environment can be seen as a PEIS. Secondly, all PEIS are connected and can cooperate by a uniform communication model. The paper also provided an example to illustrate this conception. Figure 2.7 from [28] is the example of cooperation in a simple PEIS-Ecology. A simple autonomous vacuum-cleaning robot, which can be seen as a PEIS, includes obstacle detection and various cleaning functions. However, the vacuum-cleaning robot cannot detect position itself, so it cannot execute a complex cleaning strategy. Fortunately, the home is equipped with a set of cameras, which can monitor and estimate the position of the vacuum cleaner. Then, by combining this set of cameras with the vacuum cleaner, a simple PEIS-Ecology is completed, where the cameras provide the vacuum-cleaning robot a localization function. Therefore, the vacuum-cleaner can use this localization function to accomplish a complex cleaning strategy.

Figure 2.8 from [28] illustrates some views of the PEIS-Home. The author did realize some concepts of the PEIS-Ecology.

2.2.2 Network Robot Systems

Similar to the PEIS-Ecology, Akimoto, T. and Hagita, N. in [22] and Alberto Sanfeliu, Norihiro Hagita and Alessandro Saffiotti in [46] introduced another type of robotics technology called "network robot system". This Network robot system integrates robots into a smart environment containing ubiquitous sensors, PCs, and cellular phones. The

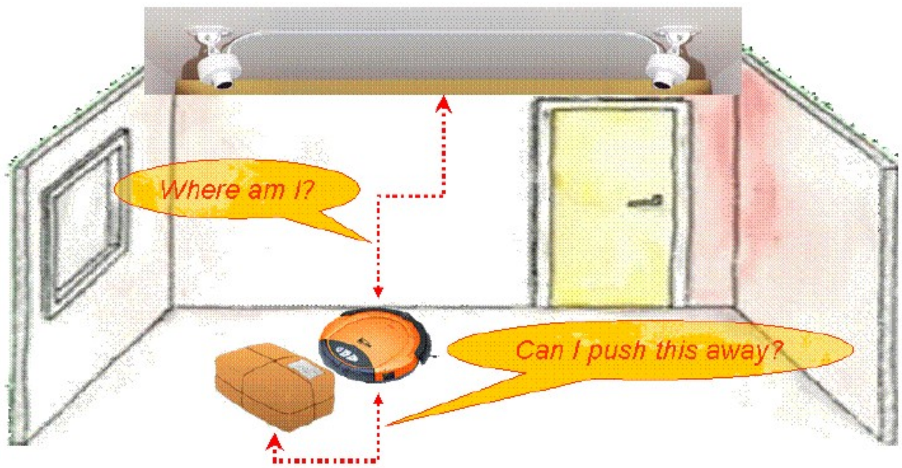


Figure 2.7: An example of cooperation in a simple PEIS-Ecology

information collected by the ubiquitous sensors and PCs in the smart environment will help robots provide robot-based services.

According to [46], Network Robot Systems (NRS) includes the following elements: physical embodiment, autonomous capabilities, network-based cooperation, environment sensors and actuators, and human-robot interaction. This means any NRS has to have at least one physical robot which has autonomous capabilities. This robot is considered as a basic element of a NRS. In addition, the environment must include other sensors such as vision cameras, and actuators such as speakers. These sensors, actuators and the robots must be able to cooperate through a network. Moreover, the NRS must have a human-robot related activity.

Figure 2.9 illustrates the concept of the Network Robot System. In the scenario of the image, an elderly person and a child ask a question to a humanoid robot, and the robot tries to reply to the question. In this case, the robot locates the old person and the child by using RFID tags, which are attached on the elderly person and the child's clothes, and answers the question by using information provided by other robots and sensors in the NRS.

Figure 2.8: Images of the PEIS-Home

2.2.3 Robot Nao

Based on the concept of the Network Robot System and the PEIS-Ecology introduced above, the authors in [42] presented a case study, which integrated a humanoid robotic platform Nao 1.2 into a smart home environment. The task of the robot Nao in this case study is to bring a can of soda from a fridge to a human user.

According to the PEIS-Ecology, the robot Nao is the basic component of the PESI-home. In addition, there are several other PEIS components that provide information to the robot Nao in this PEIS home: the PEIS-Cam provide images of the living space; the PEIS-PersonTracking which is connected to a set of stereo camera mounted on the ceiling, provide the person's location information; the PEIS-Fridge is a small sized refrigerator consisting of an actuator to open the door, a robotic arm with a griper to bring the drink outside and a camera with clustering algorithms to direct the gripper towards a can of soda. Under the assistance of the PESI-camera, the robot Nao was able to avoid

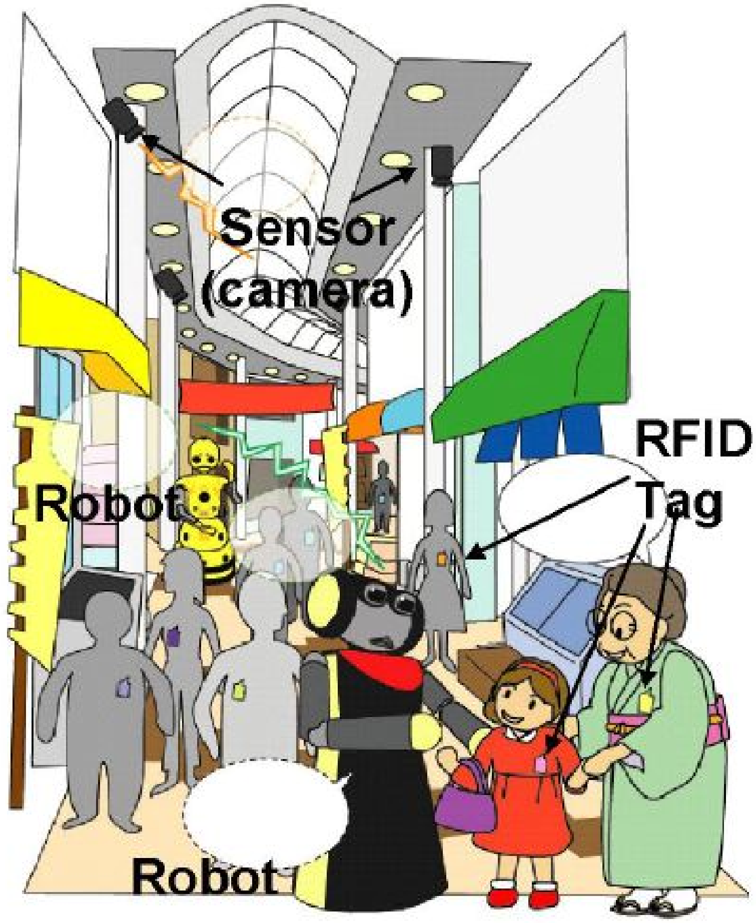


Figure 2.9: An example of the Network Robot System

obstacles and walk to the PESI-Fridge. Under the assistance of the PSI-Fridge, the robot Nao sent a request to the fridge, and a can of soda was brought out. After grabbing the soda can, the robot Nao successfully found the person by using the information from the PESI-PersonTracking. Therefore, "the bring soda"task was successfully completed.

The authors also evaluated how the performance of the robot Nao can be affected if the robot wants to locate itself without being assisted by any other PESI components. The authors used pre-defined marks attached on the ground to guide the robot Nao, but the results were poor.

Figure 2.10 illustrates the PESI-Fridge, and how a PESI-Fridge can assist the robot Nao to bring a can of soda.

Figure 2.10: PESI-Fridge

2.2.4 RoboMaidHome

Seung-Ho Baeg, Jae-Han Park, Jaehan Koh, Kyung-Wook Park and Moon-Hong Baeg initiated a smart home environment project, RoboMaidHome in [24] and [25], which integrated a light-weight service robot to provide reliable services by interacting with the smart environment through wireless sensor networks.

The environment which was built for the service robots consisted of three main components: smart objects with radio frequency identification (RFID) tags and with sensors and actuators; the home server which was used to communicate with smart objects and to manage information for reliable services; the service robot that acted as an interface in collaboration with the smart home environment. In the three components above, the service robot aims to provide directly services to humans, while the smart objects with RFID sensors or actuators are used to provide environmental information for the home server and to be controlled by the home server. The home server makes decisions based on the collected data, and controls the smart objects to assist the service robot in providing reliable services.

Figure 2.11 illustrates these components in the RoboMaidHome. Moreover, Figure 2.12 from [24] shows the structure of the RoboMaidHome in reality.

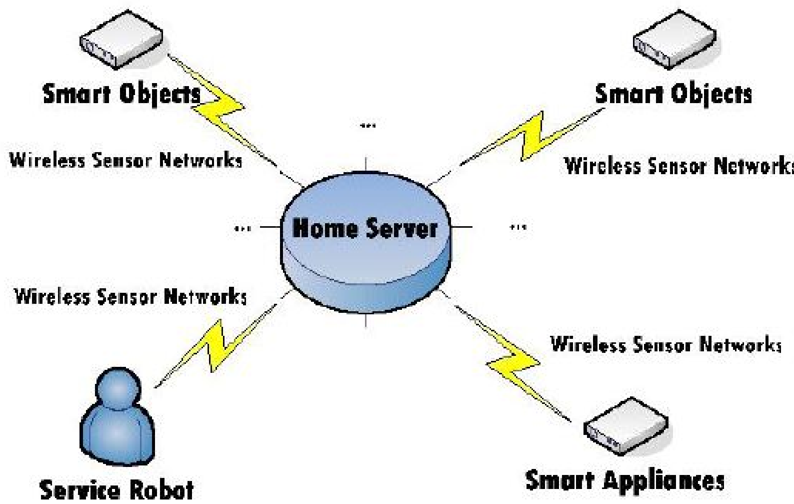


Figure 2.11: The components of the RoboMaidHome

2.2.5 Voice-based Robots in Smart Home

The authors in [29] also proposed the use of a social robot as an interface between the user and the smart environment. However, the difference is that the authors considered speech as the method of communication between robots and people.

The main contribution of the paper is that the authors discussed the importance of recognizing both the linguistic content and the tone of the spoken sentence, which would be used to understand the user's intent during the communication.

Speech is indeed a natural and efficient way for human to interact with robots. Figure 2.13 illustrates the diagram of the intention recognition schema from [29] by using the speech-based robot.

2.2.6 WABOT-HOUSE

Sugano, S. and Shirai, Y. [47] mainly introduced the structure of the WABOT-HOUSE (Waseda Robot-house) project which integrated robot technology into smart environments. Figure 2.14 illustrates this structure.

According to the concept of the WABOT-HOUSE, the robot should recognize its own indoor position by using the information provided by cameras, and its outdoor position by using GPS. Also, the robot needs to communicate with a home server, which manages

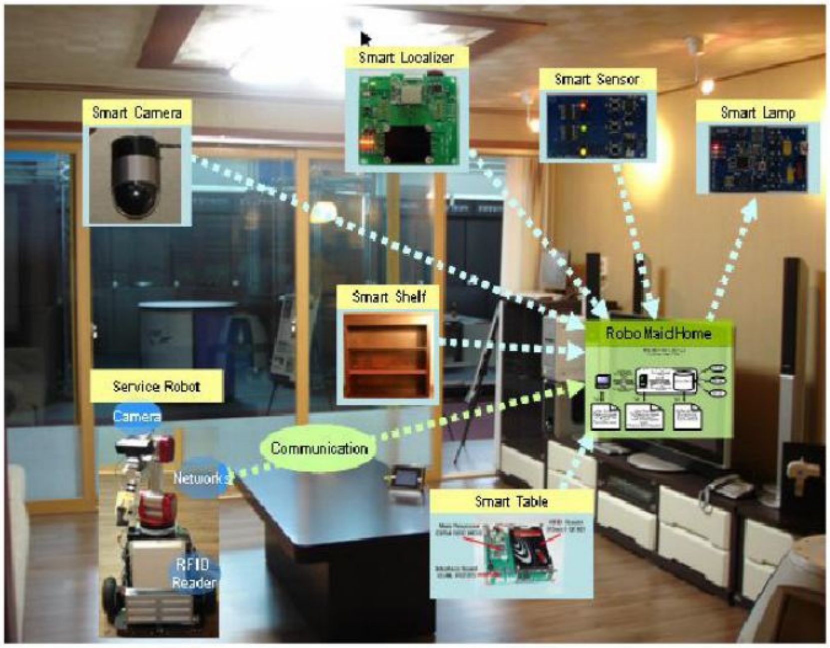


Figure 2.12: The physical structure of the RoboMaidHome

the smart home information and makes decisions to control the smart home devices.

2.2.7 Robots in ZigBee Based Smart Home

Fei Lu, Guo-Hui Tian [43] proposed another concept to integrate robots in a Zigbee-based smart home.

In the paper, robots were used as interfaces to communicate with a home server through WIFI (802.11g). The home server managed information collected by sensors and provided services for people through actuators in the smart home. For example, it can get location information from mobile phones or PDAs' GPRS Module. The protocol of the communication between sensors, devices, and the home server is the ZigBee protocol

Figure 2.15 illustrates the basic structure of this smart home.

2.3 Smart Phones

Today, mobile phones have never been more popular, and the functions of mobile phones are far greater in number and complexity than just making a call. Stylish and versatile

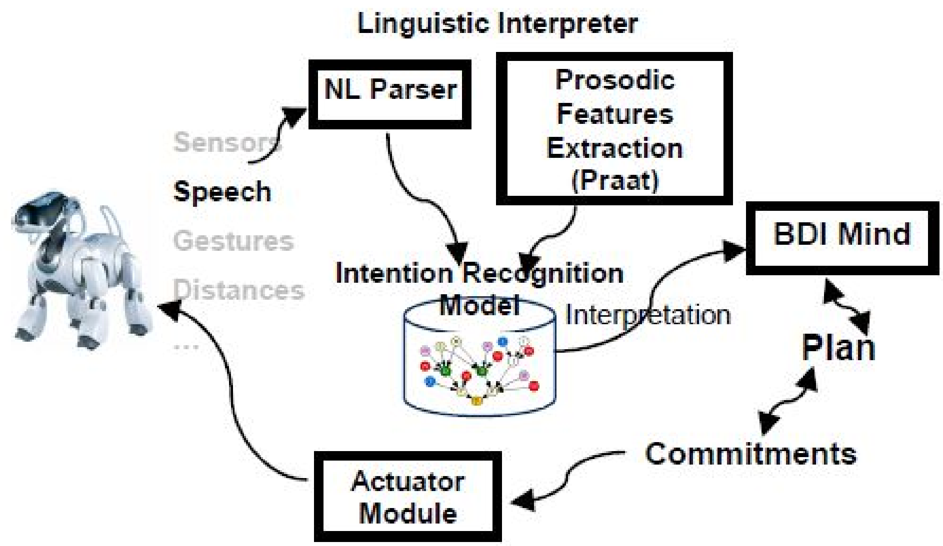


Figure 2.13: The Intention recognition schema

phones with hardware features like GPS, accelerometers, and big touch screens are more accepted in this era. IOS and Android OS are two popular operation systems on our smart phones, and they both provide a good platform for developers to use the hardware features to create innovative mobile applications.

Therefore, many developers integrate smart phones in different systems in order to improve the experience of these systems. The following are several examples of previous works.

2.3.1 Integrating Mobile Phones in a Smart Home

In 2008, the authors in [23] used a phone, a laptop (Home Server) and Bluetooth communication technology to present an End-to-End wireless solution in a smart home.

In the system, a Java-enabled mobile phone communicates with the home server via Short Message Service installed on the mobile phone, and the home server communicates with the home devices via the Bluetooth connection.

Figure 2.16 from [23] illustrates an example of the controlling process. The mobile phone send SMS messages with device name and control command. After receiving the SMS messages from the mobile phone, the Home Server sets up communication with the device, and then controls device's status.

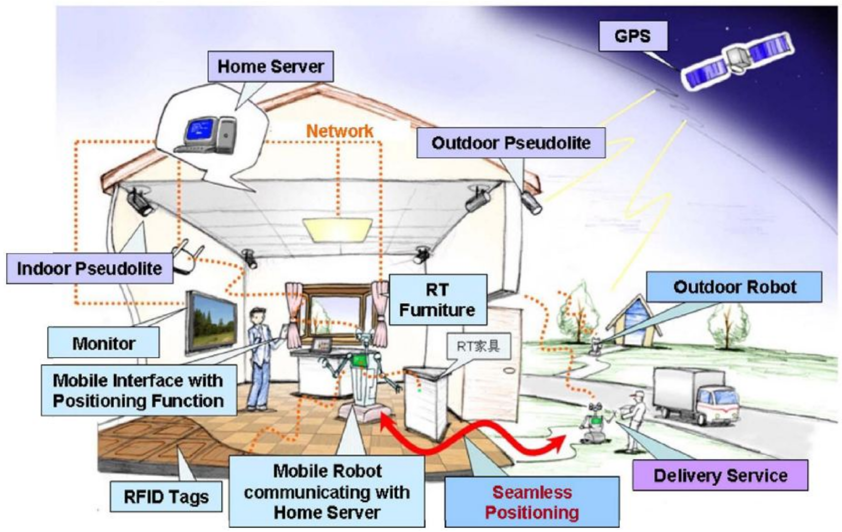


Figure 2.14: The Concept of the WABOT-HOUSE

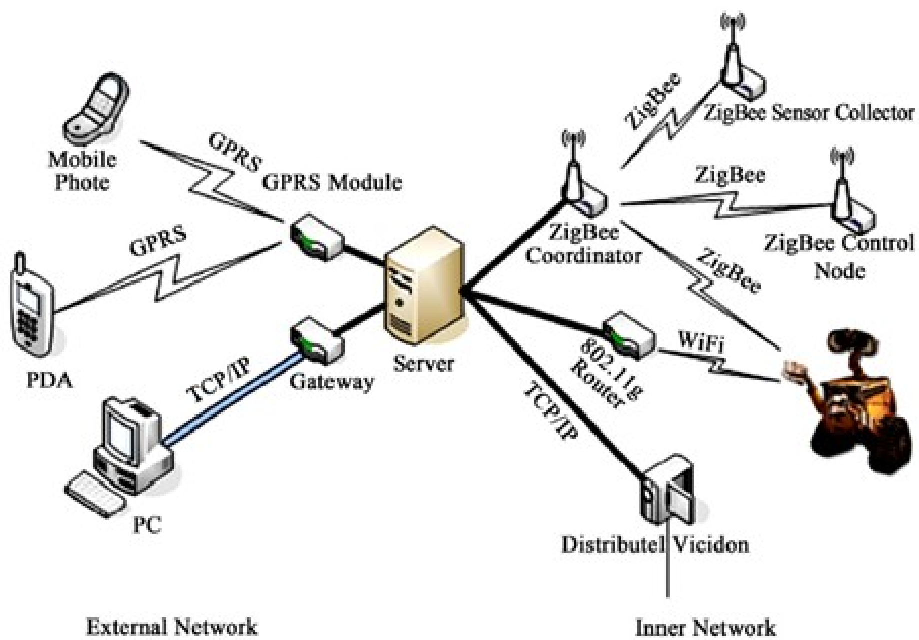


Figure 2.15: Concept of robots in Zigbee-based smart homes

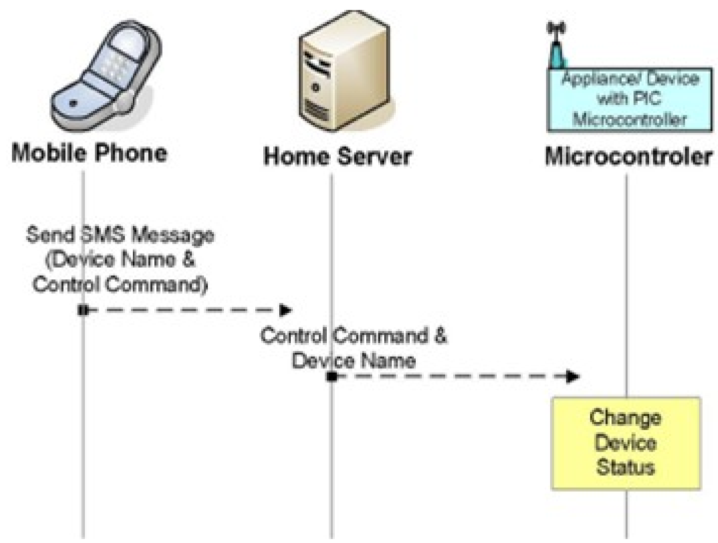


Figure 2.16: A smart phone as an interface in a smart home

2.4 Following People with a Mobile Robot

Controlling a robot to follow is an important function of our composite robot. Also, in the real word, if a robot wants to coexist with people, the ability to follow people would be considered fundamental. Figure 2.17 from [35] illustrates the concept of a person following robot.

There are two methods that can be considered to allow a robot to follow a person. One way is to install cameras in the surroundings, and use these cameras to detect a walking person and to control a mobile robot to follow that person. The author in [44] proposed a human following system by using this method. The other way is that the robot itself can detect and track a person. The latter way are more accepted, because if a robot does not depend on devices in its surroundings, it can be used in anywhere instead of being limited to some special areas. Moreover, the ability of a robot to follow people by themselves makes them more human, which is a goal of robot development.

Some related works which allows a robot to follow a person are introduced as follows:

2.4.1 Pattern on Clothes

The author in [35] used a pattern recognition system in [40] to learn and to match a human back template, which is actually a unique pattern on the person's clothes, and a shoulder template in each image captured by a camera on the robot. A computer

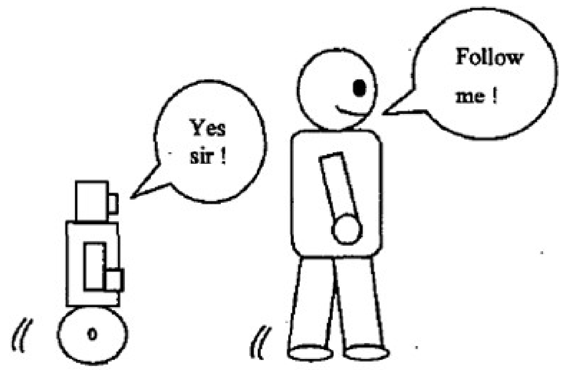


Figure 2.17: The concept of robots

calculates the position and the size of the pattern and the shoulder in each image, and then controls the robot to follow the person. Figure 2.18 shows how the system worked at that time.

However, this system is limited by the cloth, as a highly recognizable pattern is required in order for the system to detect it in a complex background.

2.4.2 People Detection and Laser Range Sensor

The author in [26] presented a multi-sensor system to track people with a mobile robot. A laser range sensor was used to detect legs, and a PTZ camera was used to detect faces. These two sensors were put on a robot to provide information for tracking people. From the x-y position of the detected face on the image, bearing and elevation relative to the robot were easily computed. Then, by combining the range and position information provided by the laser range sensor, the robot was able to follow that person.

However, the problem is that if it requires a face detection system for the robot to follow people, then these people have to walk backwards, which is obviously not convenient in the real world.

Therefore, the author in [55] improved the robot by applying a swarm intelligence based optimization algorithm, Particle Swarm Optimization (PSO) [38] [33], for people tracking instead of tracking people according to their front face. A similar laser range finder, which provided the range information for the robot to follow people, is also used.

In the research, when a person was detected, it started its people tracking system to calculate the (x, y) position of the central point of the person in each image. At the

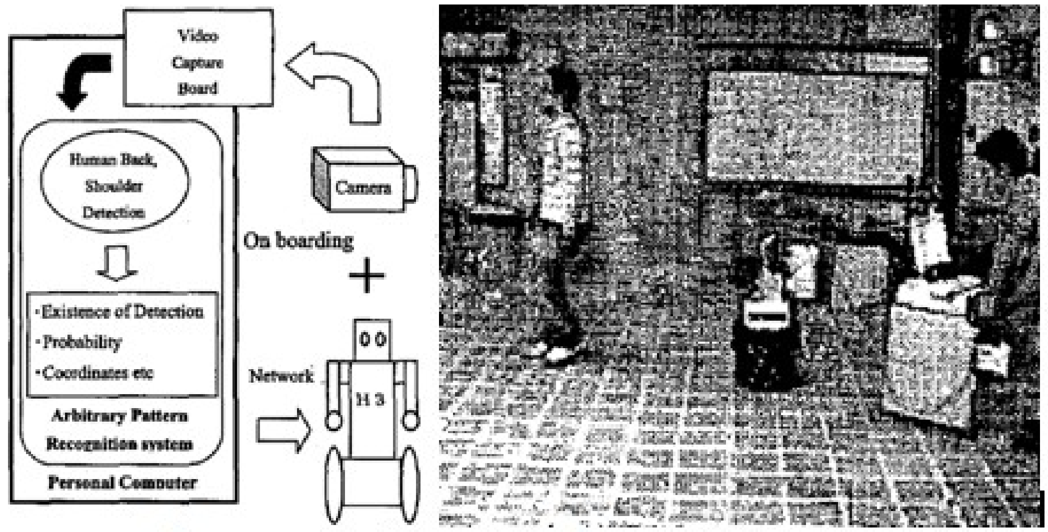


Figure 2.18: A robot follow the pattern on the cloth

same time, the distance between the robot and the person was provided by the laser range finder. By combining this information, a path was calculated inside the robot to control the robot's movements. The left image of Figure 2.19 shows the people detection and tracking result by using robot's camera, and the right image shows the result of following. Figure 2.20 shows the robot used in the project.

2.4.3 Infrared LEDs

In 2012, the author in [50] presented another design and implementation of a mobile robot from [1] for people following. Figure of 2.22 illustrates the robot. Two infrared LEDs were tied on the back of people. A robot with three wheels and controlled by a notebook was able to recognize the LEDs by using the camera on its top. After "seeing"the LEDs, the notebook computed the position of the person and controlled the robot to follow the person.

In the project, each image was filtered by a color range, so the two infrared LEDs could be easily recognized as the left image of figure 2.21 shows. Then, the information of the person's position and the distance could be calculated. The right image of figure 2.21 shows the infrared LEDs tiled in the person's back.

However, it is not a good idea for a robot to have to depend of extra devices to follow people. Moreover, in the real world, it is not convenient and comfortable for people

Figure 2.19: Using people detection and laser range sensor to follow a person

to wear these devices. Therefore, I strongly agree with those who use people detection methods to provide the position information of the person to the robots.

In this chapter, we introduced some previous smart homes which used computers, smart phones or robots as an interface. We also introduced some previously tested methods for robots to follow a person. From these previous works, I noticed that computers, smart phones, and even robots cannot meet the needs of today's smart home interface. Therefore, we propose a composite robot as the interface in a smart home, and we make it more human by using voice command and by making it follow a user. Our composite robot will be mainly introduced in the next chapter.

Figure 2.20: Using people detection and laser range sensor to follow a person

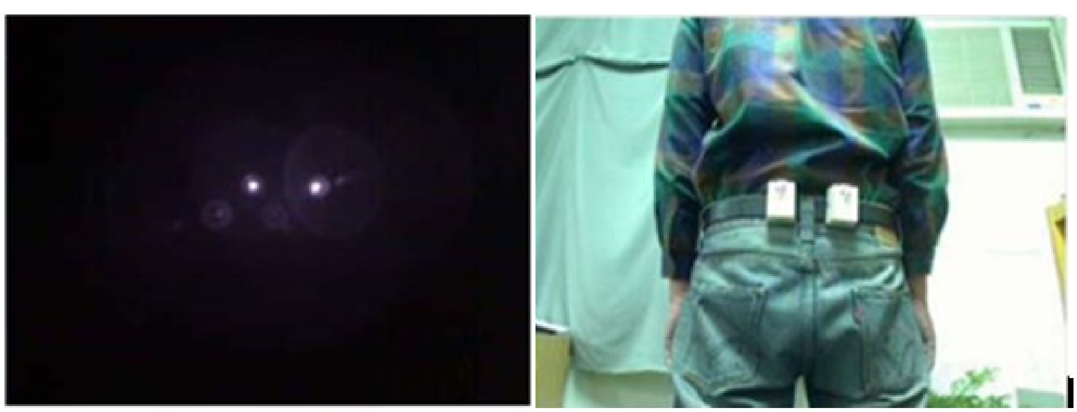


Figure 2.21: Infrared LEDs and a filtered image

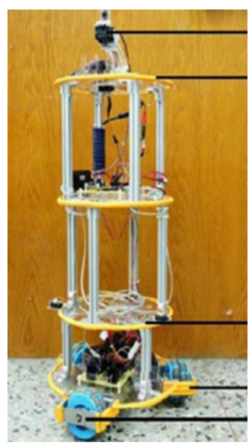


Figure 2.22: Using infrared LEDs to follow a person

Chapter 3

The Proposed Separable Robot Interface

In this chapter, a detailed description of how the proposed composite robot interface works in our smart home is given. We start by introducing the high-level design to describe the whole idea of our system, in which, we will introduce the two modes of our system: the Portable Mode and the Robot Mode. These are designed for a person to use at home and when away from home. The Robot Mode is our main approach, and will be explained in other sections. Details of the algorithm used for following a person visually will be introduced. Also, the method we proposed for invoking different web services, the position of the "brain", the approach we used to detect a person and the algorithm we developed for the chip of our system's robot car to be a real-time person follower under the Robot Mode will be explained.

3.1 High-level Design

Figure 3.1 illustrates the high-level design of our system, and also provides the main idea of how our system works.

We developed an application for an Android phone, which is the fundamental of our system. In the application, there are two modes: the Portable Mode and the Robot Mode.

The Portable Mode: If a person is not at home, her Android OS phones can be manually connected to the Google Calendar or a Home Automation web service. The home automation web service we implemented was developed by Bassam Barake from

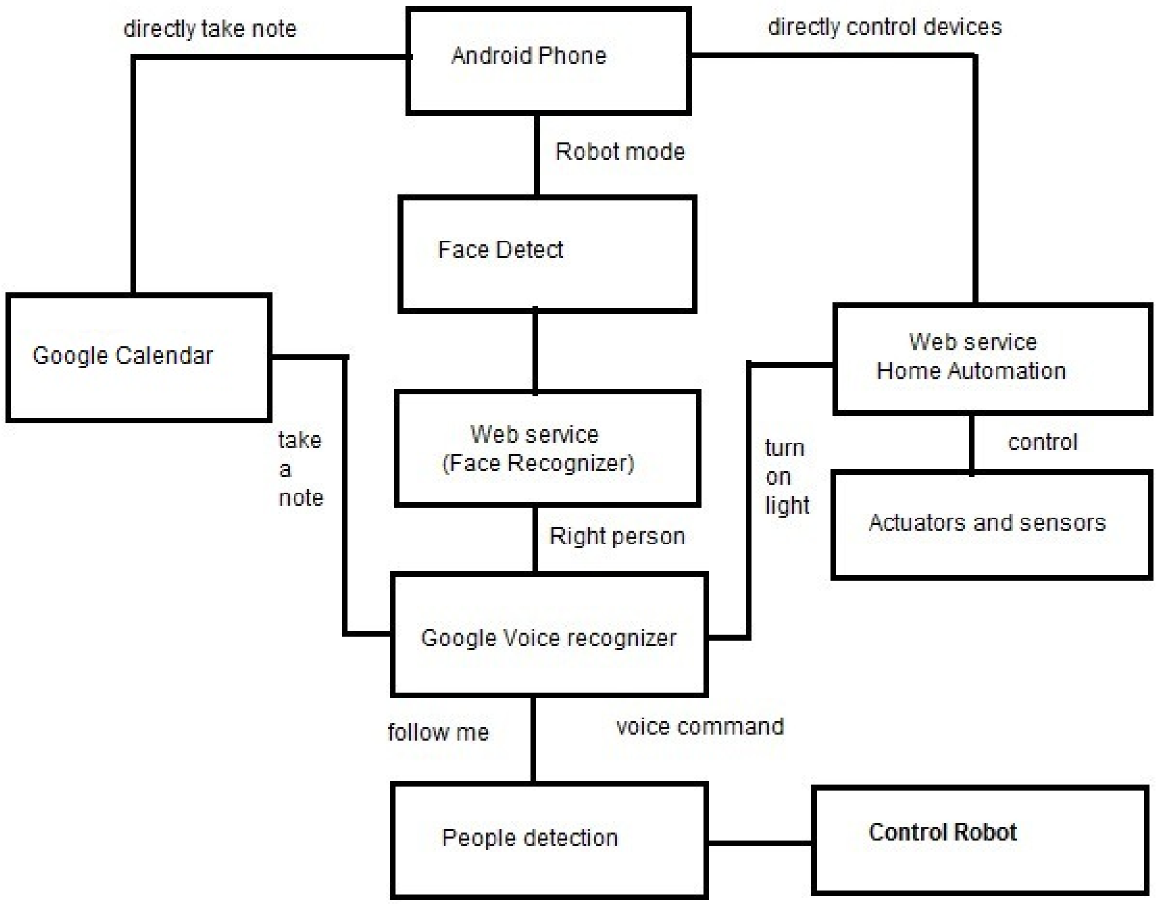
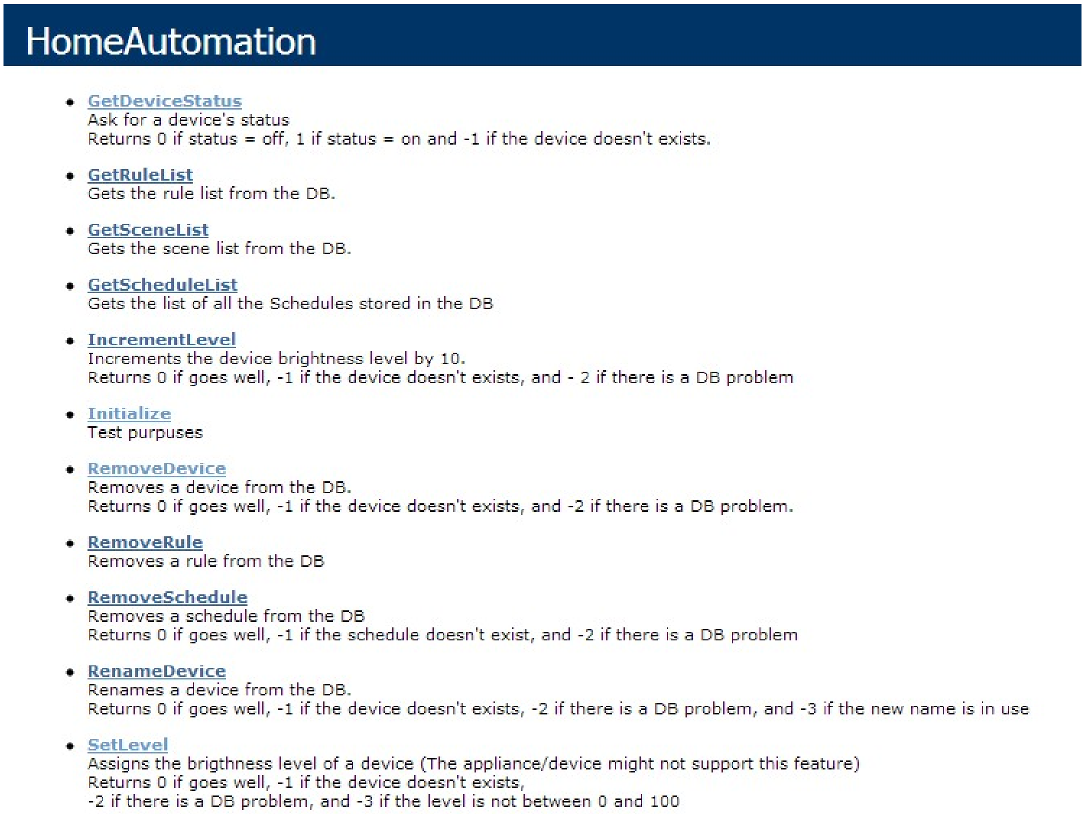


Figure 3.1: The high-level design

the MCR lab of the University of Otmawa. This web service allows users to monitor their room information and to control home devices. Figure 3.2 is a part of this web service retrieved by a computer. Similarly, the Google Calendar is a web service developed for people to take notes, which was introduced in Chap.1. In this case, the Android phone would be acting as a traditional phone interface in a smart home.

The Robot Mode: When a person is at home, she can select the Robot Mode in her phone and put her Android phone on the robot car. Then, a smart robot would be set up and act as the interface in the smart home. It can start a conversation with the user by recognizing the user's face and can invoke different web services by understanding the


HomeAutomation

- GetDeviceStatus
  Ask for a device's status
  Returns 0 if status = off, 1 if status = on and -1 if the device doesn't exists.
- GetRuleList
  Gets the rule list from the DB.
- GetSceneList
  Gets the scene list from the DB.
- GetScheduleList
  Gets the list of all the Schedules stored in the DB
- IncrementLevel
  Increments the device brightness level by 10.
  Returns 0 if goes well, -1 if the device doesn't exists, and - 2 if there is a DB problem
- Initialize
  Test purpuses
- RemoveDevice
  Removes a device from the DB.
  Returns 0 if goes well, -1 if the device doesn't exists, and -2 if there is a DB problem.
- RemoveRule
  Removes a rule from the DB
- RemoveSchedule
  Removes a schedule from the DB
  Returns 0 if goes well, -1 if the schedule doesn't exist, and -2 if there is a DB problem
- RenameDevice
  Renames a device from the DB.
  Returns 0 if goes well, -1 if the device doesn't exists, -2 if there is a DB problem, and -3 if the new name is in use
- SetLevel
  Assigns the brigthness level of a device (The appliance/device might not support this feature)
  Returns 0 if goes well, -1 if the device doesn't exists,
  -2 if there is a DB problem, and -3 if the level is not between 0 and 100

Figure 3.2: The home automation web service

user's voice commands. Moreover, it is able to follow the user if that is what the user wants. We will explain the detail of the Robot Mode in the following sections.

3.2 Invoking Different Web Services Through Voice

As explained in the High-level design, a smart phone application is required to connect with the Google Calendar, in order to take or read user's notes, and to connect with the Home Automation Web service to control smart home devices. In the Portable Mode, it is easy to invoke these web services by pressing the buttons in our smart phone application. In the Robot Mode, we tried to invoke these web services by using voice commands. The voice command is recognized by Google voice, which is initialized in a ubiquitous method.

The details of invoking different web services are as follows:

3.2.1 Activating Google Voice Recognition

Understanding voice-command is not difficult for smart phones today. Google provide the Google Voice search [13] for their Android phones to recognize voice, and for IOS system, they have SIRI. However, the problem is that these two recognition services both require a person to press a button in the application to initialize the voice recognition, while in our system's Robot Mode, it is impractical to do this.

Under the Robot Mode, the smart phone is attached to a robot car, and the user could be several meters away from the robot. Therefore, it is not feasible for the user to press a button on the phone every time he or she wants to activate voice recognition to make a command.

Therefore, in the Robot Mode of our system, the smart phone needs to be aware of when it should use its voice recognition itself. People usually notice that someone wants to speak to them when that person is looking at them. Based on this concept, we developed and designed a method consisting of image processing for Android phones to let if be aware of when to activate its voice recognition. Figure 3.3 shows the flow chart of our proposed method.

As illustrated in the figure, when the Robot Mode is activated, the camera of the Android phone is opened and looks for faces first. We implement the Haar-cascade model in OpenCV [10] to detect faces. Haar Like Cascade presented by Viola and Jones in [49] is an object detection application based on a boosted cascade of simple features. Figure 3.4 from [49] illustrates the selection of first and second features when Haar-like features are used to detect a face. Through Figure 3.4, we can find that the main facial features used in Haar-cascade face detection are the two eyes of a person. Therefore, a person's face is only detected if he or she faces the camera, and this is similar to when a person wants to start a conversation. Figure 3.5(b) shows that a face is detected when a person is facing the camera, while Figure 3.5(a) shows no face result from face detection when a side-face is shown to the camera.

Moreover, in order to improve the security and personalization of our system, before activating the Voice Recognition, an identification of the person is required. Therefore, we developed a web service by using the Eigenfaces algorithm, which was also used by Turk, M.A. and Pentland, A.P. to recognize faces in [48]. The Eigenfaces algorithm we implemented is also from OpenCv. In our system, the face recognition is processed through our web service, so that the face information of different smart home users can be stored in a same server and can be managed conveniently by the users themselves.

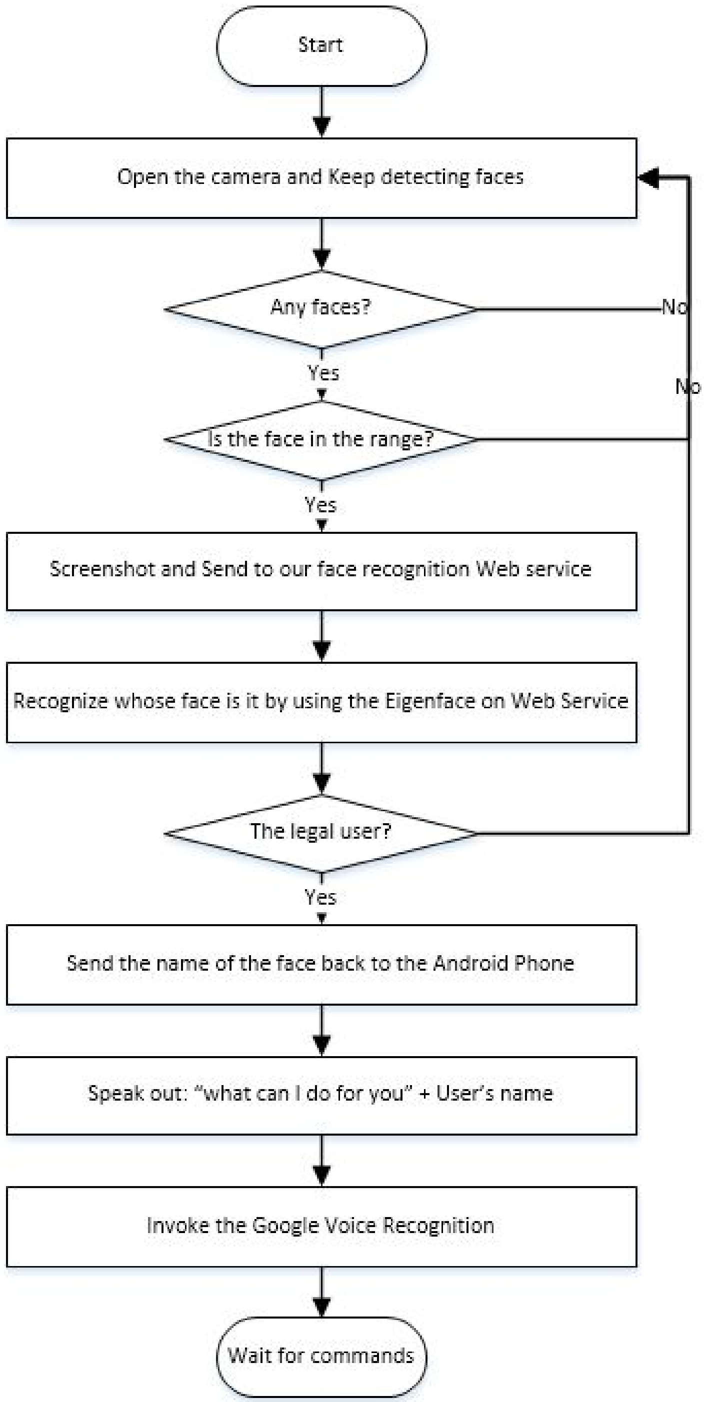


Figure 3.3: The flow chart of initializing voice recognizer

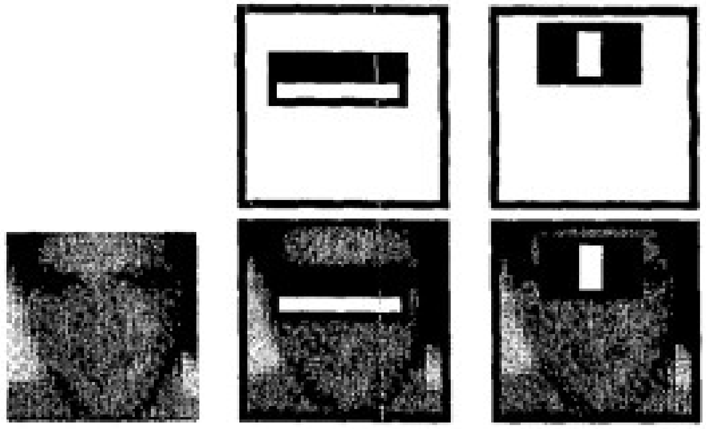
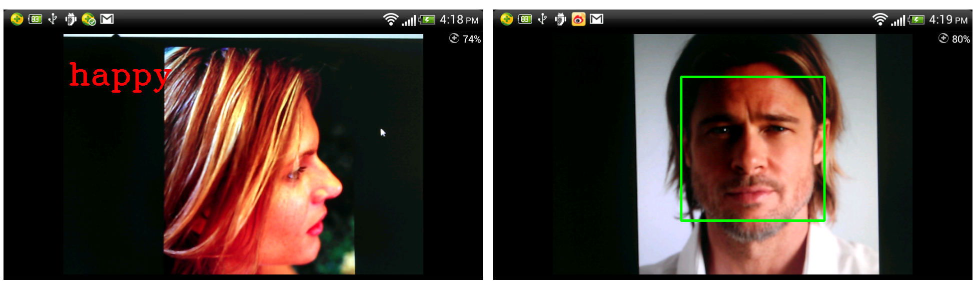


Figure 3.4: Haar: The first and second features



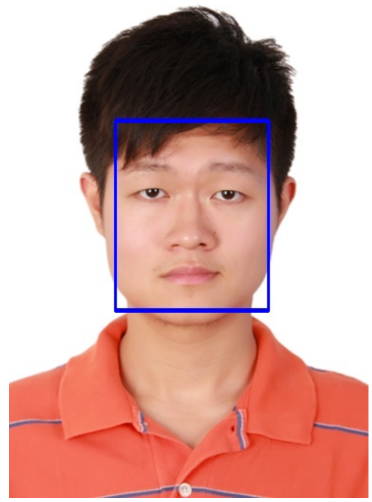
(a) Side-facing result (b) Front-facing result

Figure 3.5: Examples of Haar-cascade face detection result

Therefore, under the Robot Mode, if a face is detected, a screenshot will be automatically and immediately sent to the face recognition web service. The face in the screenshot is compared with the information saved in the web service. After the face recognition, only the legal user whose face information matches the face information stored in the web service can activate the voice recognition.

Before using the face recognition web service, we need to train the recognizer first. Therefore, the face information of a user is extracted from photos uploaded by the user. The face in each photo is detected by Haar-cascade and marked by a fixed size (260*320 pixels) square. Then we change these pixels from RGB to Grayscale by using the function in OpenCv, and re-save as another photo. This re-saved gray image will be used to train the recognizer and be a reference when recognizing faces. Figure 3.6(b) is an uploaded photo, and Figure 3.6(a) is the extracted gray image.

(a) Extracted face



(b) Original photo

Figure 3.6: Extracted face information from an uploaded photo

3.2.2 Activating Google Calendar and the Home Automation Web Service

Google Calendar and the Home Automation web service are connected according to voice-commands from users. After activating the Google voice recognizer, a user has the following voice-command options:

- **"Take a note":** This voice command allows a user to retrieve their Google Calender to take a note. After connecting the user's Google Calender, the smart phone will ask several questions to help the user finish the note through voice. The questions are: "note title", "note detail", and "note location". Google Voice RecognizerIntent is also used to translate the user's spoken words into text. In this way, a note is taken through a conversation. Figure 3.7 is a related example.

- **"Read schedule":** As a robot, reminding users about their schedule is an important function. Therefore, under the Robot Mode, a user can know her schedule. By saying the words "read schedule", the smart phone will retrieve the user's Google Calendar, looking for today's notes, which were taken beforehand, and will speak out the notes.

- **"Turn on light":** This voice command allows the user to control their smart home devices. By recognizing these words, the smart phone sends a request to the

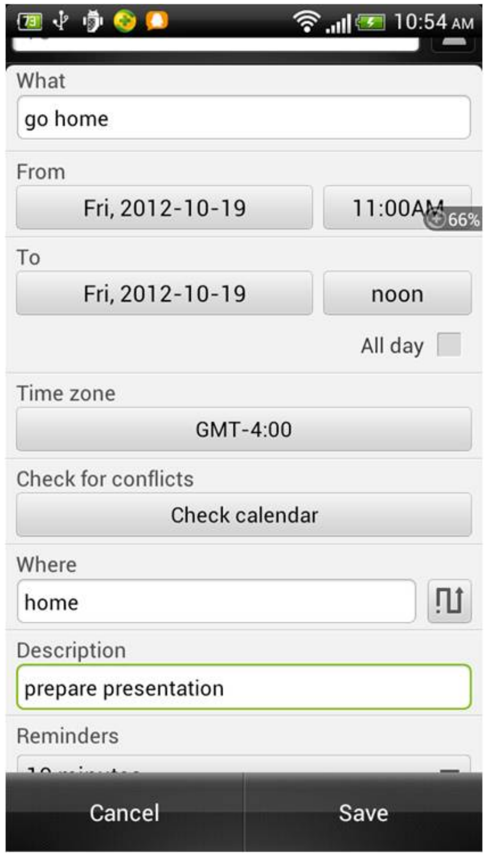


Figure 3.7: Take a note

Home Automation web service, and the web service commands the light actuator to turn on the light. Figure 3.8 is a related example.

3.3 Conception of a New Inexpensive Robot

The Robot Mode is aimed to be used when a person is at home. Following a person is the most important function for the Robot Mode.

In our system, the composite robot needs to follow the person by using the smart phone's camera and the robot car's wheels. Therefore, a person needs to be tracked visually. However, tracking a person visually is not easy, because the image taken by a camera can only display a 2D world while the real word is 3D. Furthermore, in real time, it is even harder to follow a person accurately in this 2D image world when the person is moving forward, left-forward, and right-forward.

Figure 3.8: Turn on light

We therefore, developed a method for following a person visually. In our proposed method, we try to take advantage of the mobility of the robot car to follow a person instead of calculating the particular position of the person in the real world visually and then controlling the robot to follow the person. The basic concept of our method is that the person's position is compared with two thresholds in each frame taken by the smart phone's camera. The threshold in our system is a value in pixels, beyond which we consider that the person moved. Then, the robot car moves forward in order to keep a constant distance to the person, and it turns left or right in order to keep the person in the middle of each frame.

In this section, we will first introduce the algorithm we implemented to detect a person. Then the details of our method for following a person visually and the role of the smart phone in our system will be explained. We use the Android OS smart phone, HTC EVO 3D [7], as an example to explain these ideas and to show the models for calculating the thresholds. At last, the robot car and how it is controlled in order to follow the person will be introduced.

Figure 3.9 illustrates the flow chart of how our Robot Mode works in order to follow a person.

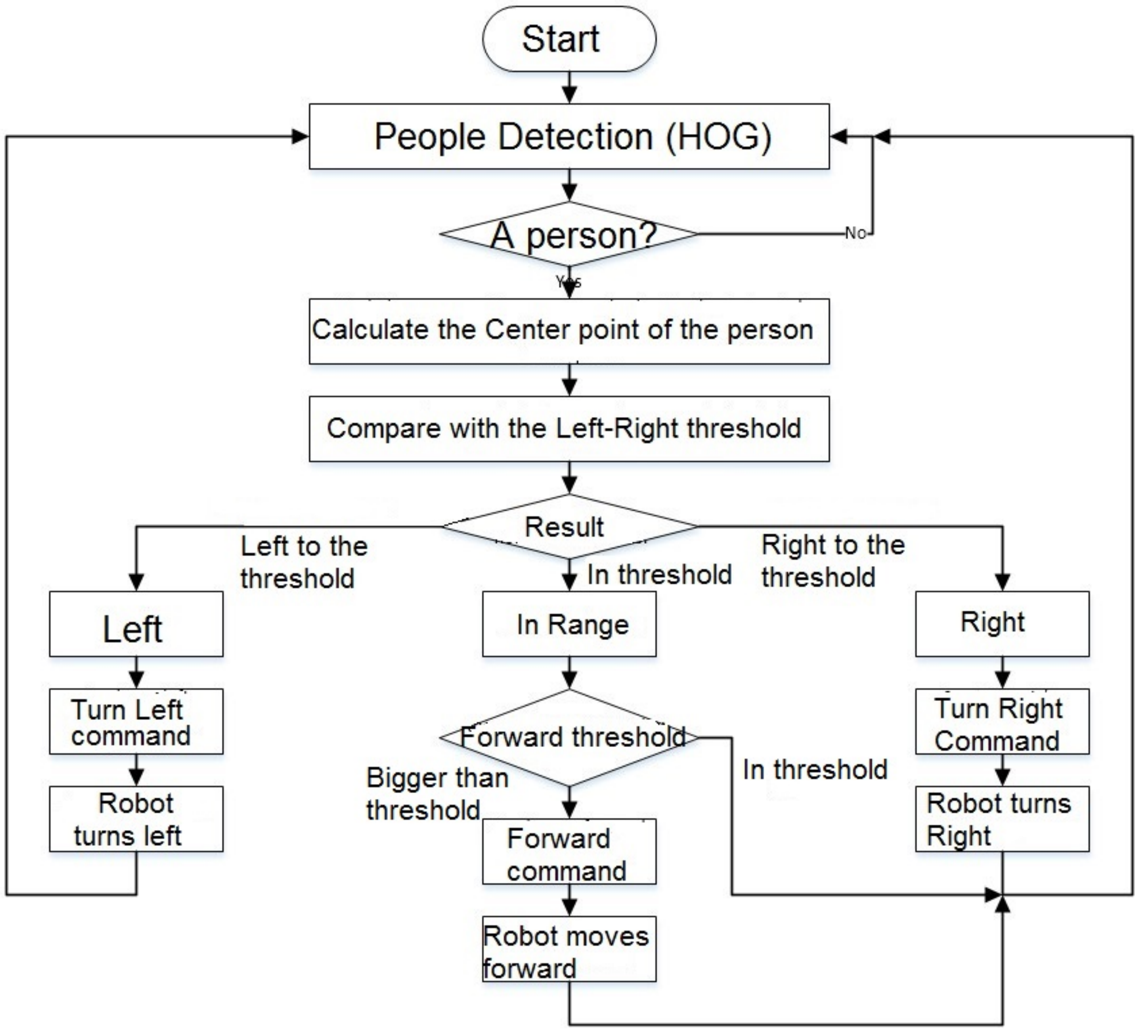


Figure 3.9: The flow chart of following a person

3.3.1 The People Detection Method

In our system, before the phone controls our robot car to follow a person, the phone requires the position of the person. As mentioned in the literature review, we prefer to use person's detection methods for calculating a person's position. Detecting humans from images is a challenging task, because of a person's variable appearance. Therefore, a fixed feature set which could be recognized even in a cluttered background or under dim lighting is necessary. There is extensive literature on finding this fixed feature set to detect a person from images. [34] is a survey that contains many early efforts of person's detection. In our system, the Histogram of Oriented Gradient (HOG) descriptors are implemented to detect a person.

In 2005, Dalal, N. and Triggs, B. [31] found that locally normalized Histogram of Oriented Gradient (HOG) descriptors are excellent at describing feature sets of humans relative to other feature sets. Therefore, the authors presented a high performance people detector by using HOG descriptors. The Linear Support Vector Machine (SVM) was also used as a baseline classifier.

According to [31], in order to build a HOG descriptor, the image window needs to be divided into a grid of cells, and a histogram of gradient directions is computed in each cell. For better consistency in illumination and shadowing, a measure of local histogram "energy"over larger spatial regions called blocks is computed, and these results are used to normalize all of the cells in the block. The normalized descriptor blocks are Histogram of Oriented Gradient descriptors.

Figure 3.10 from [31] illustrates how the whole system works. The detector window consists of a grid of overlapping blocks, in which Histogram of Oriented Gradient feature vectors are extracted. Then, a linear Support Vector Machine (SVM) classifies the resulting vectors into person or non-person. The detection window is scanned across the image at all positions and scales.

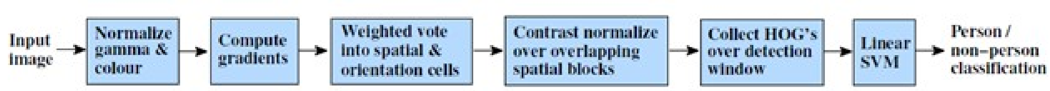


Figure 3.10: The flow chart for HOG

This HOG method for human detection has proven to be particularly successful, which is why we implemented it in our system.

We keep the parameters which are proposed by the authors in [31] for the default properties of the detector: 64*128 detection window; 16*16 pixel blocks; 8*8 pixel cells; linear gradient voting into 9 orientation bins; Gaussian spatial window with 8 pixel. We also keep using the whole body template for the person's detection.

However, the resolution of images which are taken by a smart phone's camera also influences the speed and the accuracy of the person's detection in our system. When implementing the HOG descriptors in my Android phone (The HTC EVO 3D [7]), the resolution was initially set to 800*640 pixels, but the speed of detecting a person was very slow (about 5 seconds per frame). In order to make it work in real-time and ensure the accuracy ofa person's detection, the resolution of frames taken by the smart phone's camera is fixed to 240*160 pixels. Figure 3.11 illustrates an example of the person's detection results when the resolution is set to 240*160 pixels. Moreover, we set the maximum height of a person in each frame to 140 pixels, otherwise, the person cannot be detected effectively.

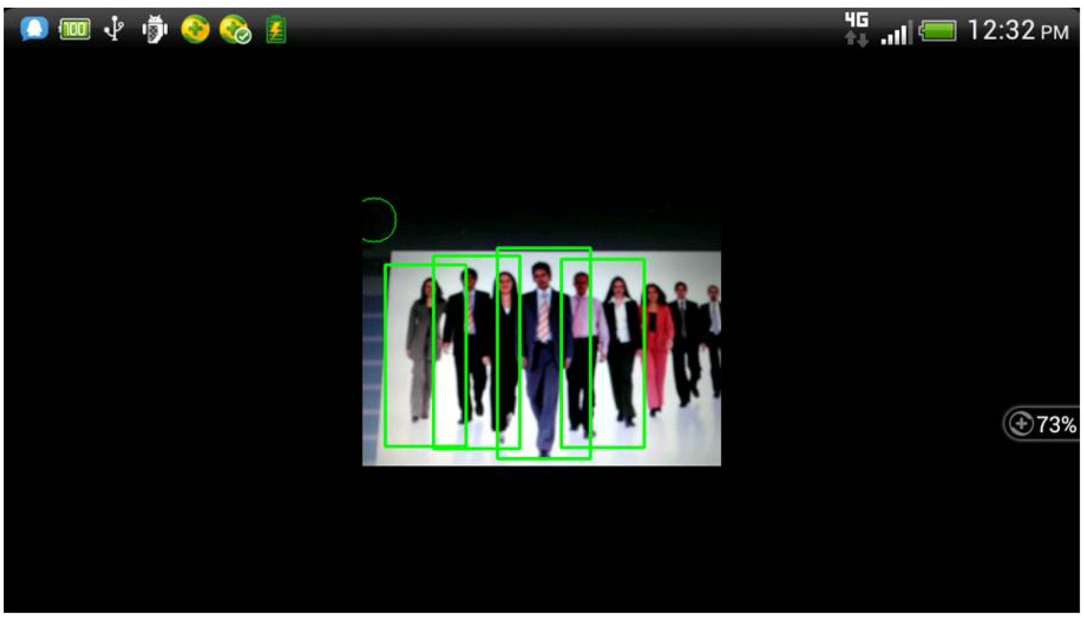


Figure 3.11: A person's detection example under 240*160 pixels

3.3.2 Important Parameters

In this subsection, some important parameters of our system, which influence the algorithm for following a person visually will be explained. We use an Android OS smart phone, HTC EVO 3D [7], as an example to explain the algorithm that was used to calculate these important parameters. These algorithms and methods can be reused in any smart phone by changing several smart phone parameters in the algorithms. Firstly, we propose a position for putting the smart phone on the robot car to connect. Secondly, we implement an algorithm to calculate the angle of view of the HTC EVO 3D smart phone camera. Lastly, we use the angle of view to calculate the minimum following distance as an example in our system when our proposed robot is following a 1.8m (5.9 feet) tall person.

The Position of The "Brain"

The focal length of a smart phone camera, the template for a person's detection, and the position of the camera when putting the smart phone on the robot car will influence the accuracy of the algorithm used to calculate the person's position.

In our system, the template for a person's detection is the whole body of the person, as mentioned above, and the focal length of a smart phone camera depends on the manufacturer of the user's smart phone. Therefore, we propose a potential position for the smart phone when it is connected to the robot car. Figure 3.12 illustrates our proposal. We recommend attaching the smart phone to the end of the robot car.

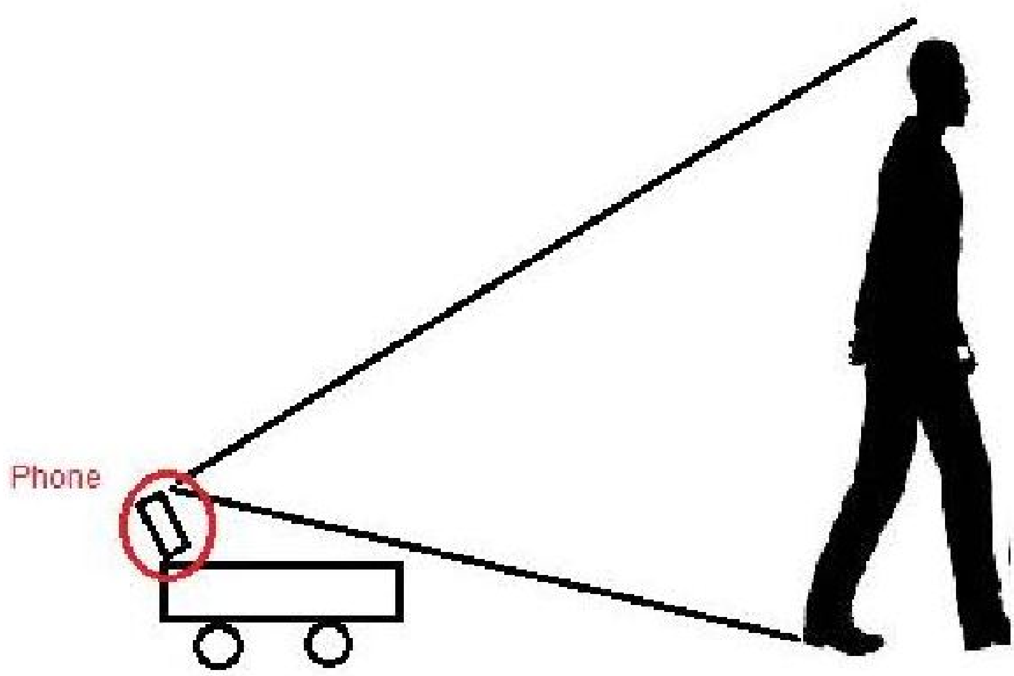


Figure 3.12: The smart phone position

The Angle of View

The smart phone in Figure 3.12 is the HTC EVO 3D [7]. Its camera is a pinhole camera, which is a simple camera without a lens and with a single small aperture.

Figure 3.13 from WIKIPEDIA [11] is an example of a pinhole camera when it is taking a picture. Figure 3.14 shows the geometry of a pinhole camera seen from an axis.

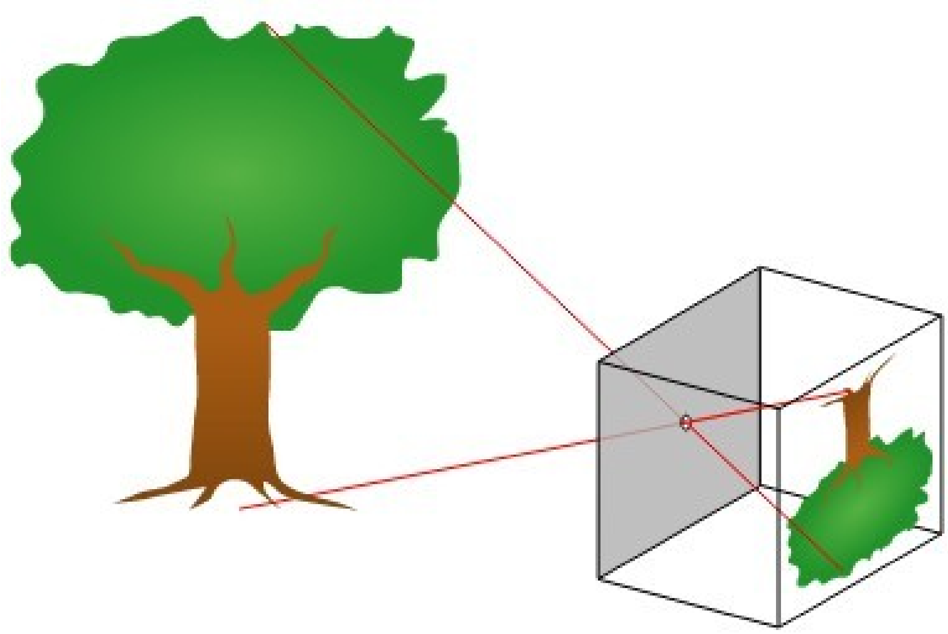


Figure 3.13: Pinhole camera

Morever, the focal length of the HTC EVO 3D camera is 35mm when it is considered as a 35mm film camera [4]. The frame size used in a 35mm film camera is 36mm*24mm [2]. Therefore, we can calculate the camera's angle of view according to Equation 3.1 from [3], where ‘d ’represents the size of the frame of a 35mm film camera, and ‘f ’represents the focal length. The angle of view describes the height and width of a given scene that is imaged by a camera.

$$a = 2 \arctan \frac{d}{2f} \tag{3.1}$$

$$a(h) = 2 \arctan \frac{d}{2f} = 2 \arctan \frac{36}{2 * 35} = 54.4 \tag{3.2}$$

$$a(v) = 2 \arctan \frac{d}{2f} = 2 \arctan \frac{24}{2 * 35} = 37.8 \tag{3.3}$$

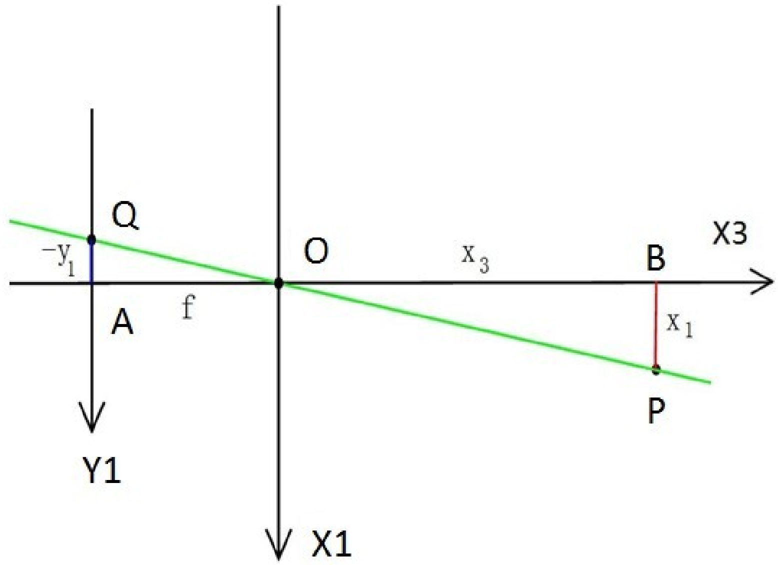


Figure 3.14: the geometry of a pinhole camera

A camera's angle of view can be measured horizontally, vertically or diagonally, as shown in figure 3.15. Therefore, with Equation 3.1, we calculate that the horizontal angle of view of a picture taken by the EVO 3D is 54.4 degrees (3.2), and the vertical angle of view is 37.8 degrees (3.3).

However, as the maximum pixels for detecting a person are only 140 in height, the real vertical angle of view of a scene that is imaged by the smart phone camera is 33.4 degrees (Equation 3.4).

$$a(v) = 2 \arctan \frac{d'}{2f} = 2 \arctan \frac{24 * 140/160}{2 * 35} = 33.4 \tag{3.4}$$

The Following Distance

By combining the position of the smart phone on the robot car and the vertical angle of view of the camera, we can calculate the distance between the smart phone and the person when the smart phone is used to detect the person and to control the robot car to follow the person.

Here we give an example by calculating a minimum following distance when our robot

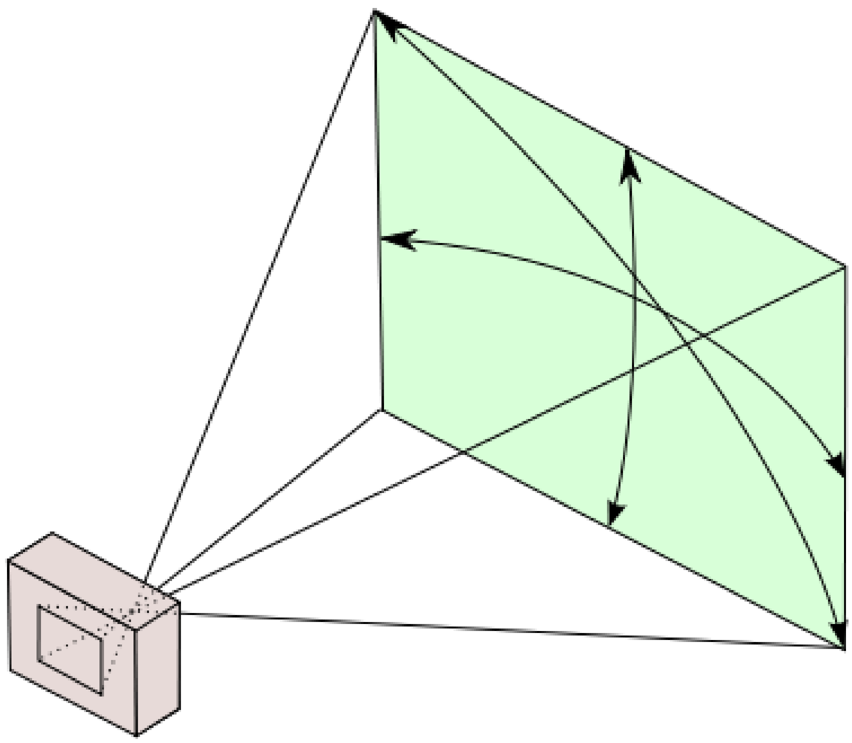


Figure 3.15: A camera's angle of view

is following a 1.8m person.

Figure 3.16 shows an ideal situation, in which the height of the person's whole body is coincidently 140 pixels in the image taken by the smart phone's camera. Assuming the person is 1.80m tall, the point O is the optical center of the smart phone camera, and it is about 0.2m from the ground. The oC is the distance between the smart phone and the person. Therefore, we can have Equations: 3.5, 3.6, and 3.7. Then, the minimum distance X is 2.9m in this case. Moreover, in this ideal situation, the angle between the smart phone and the vertical line is 10 degrees when the smart phone is connected to the robot car as the robot interface in the smart home.

$$\angle AoC + \angle BoC = 33.4 \tag{3.5}$$

$$\tan \angle AoC = \frac{AC}{oC} = \frac{0.2}{X} \tag{3.6}$$

$$\tan \angle BoC = \frac{BC}{oC} = \frac{1.6}{X} \tag{3.7}$$

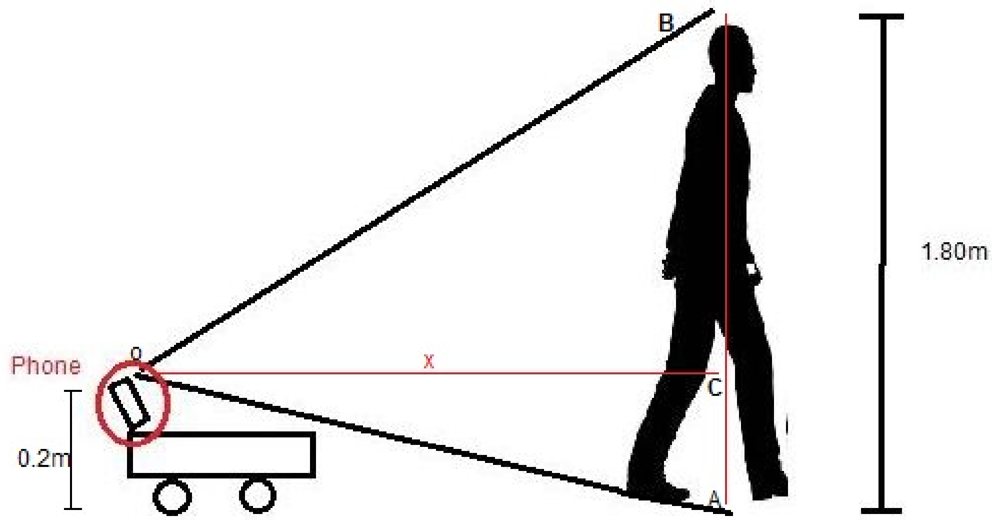


Figure 3.16: Calculating the minimum distance

3.3.3 Finding the Direction in which a Person is Moving

In this subsection, we will introduce the algorithm we developed in the smart phone. The algorithm allows our system to determine the direction in which a person is moving by using the frames taken by the smart phone's camera.

As mentioned above, the basic concept of our method for following a person is to compare the person's position with two thresholds in each frame taken by the smart phone's camera. Therefore, in this subsection, we also present two examples to explain how to calculate the thresholds in our system.

Moving Left and Moving Right

To find out whether a person is moving horizontally (left or right) in frames taken by the smart phone's camera, we require information about the horizontal image resolution taken, the horizontal center point of the person when he or she is detected, and the horizontal angle of view of the camera.

For example, when we use the HTC EVO 3D in our system, as mentioned above, the image resolution for a person's detection is 240*160 pixels, where it is 240 pixels in width. The horizontal angle of view of the HTC EVO 3D's camera is 54.4 degrees, which was calculated above.

According to the pinhole camera model, Figure 3.17 and Figure 3.18 show a real scene and the related mathematical model for calculating whether the person is moving left or right. In the model:

- The point o is the optical center of the smart phone's camera. The point P is the center of the person's horizontal position when he or she is detected in the real world, while the point C is the person's horizontal position displayed on a frame taken by the camera, and the point E is the central point of the frame.

- The line AB stands for the width of each frame taken by the smart phone's camera, and the value is 240 pixels. The line CE is the horizontal value (pixels) between the center point of the detected person and the center point of the frame. The line OD is the vertical distance (meters) between the camera and the person in the real world. The line DP stands for the distance the person has moved from the center line ED.

- The value of angle AOB is the same as the horizontal angle of view of the camera, which is 54.4 degrees.

We compare the value of the line CE in pixels with a threshold to figure out whether the person is moving left or right in the real world. As the value of the angle DOP is the same as the value of the angle COE, the value of CE can be calculated by using Equation 3.9, where the value of the line OE is 233.5 (pixels) which can be calculated with Equation 3.10.

$$\frac{DP}{OD} = \frac{CE}{OE} \tag{3.8}$$

$$CE = \frac{DP * OE}{OD} \tag{3.9}$$

$$OE = \frac{AE}{\tan \angle AOE} = \frac{120(pixels)}{\tan(27.2)} = 233.5(pixels) \tag{3.10}$$

Now we need to find a proper threshold, which will be compared to the value of CE in each frame. If the value of CE is bigger than this threshold, we consider that the person is moving left or right.

Here is an example of how to calculate the threshold for detecting whether a person is moving left or right. According to [37], the average step length of an adult is

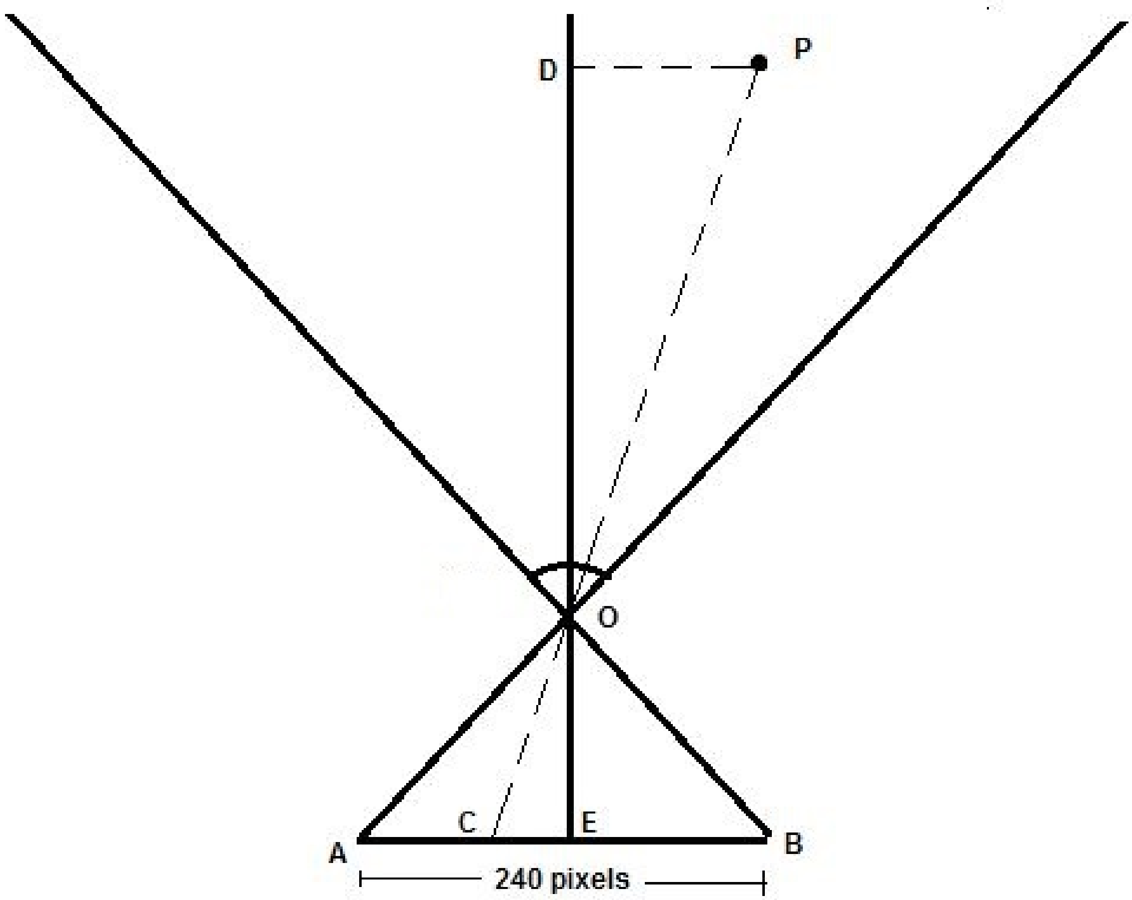


Figure 3.17: Mathematical model for moving left or right

(0.74m+0.04/leg length). Therefore, if a person is stepping to the left or the right, his or her body's central point position will be about 0.37m away from the original position. This means that the general value of DP is 0.37m if a person is moving left or right. Then, for example, if we assume that a user is 1.8m tall, the value of OD is 2.9m as calculated above, and we obtain the following:

$$CE = \frac{DP * OE}{OD} = \frac{0.37 * 233.5}{2.9} = 30(pixels) \tag{3.11}$$

According to Equations 3.11, the general CE is equal to 30 pixels. Therefore, the threshold we used to determine whether a person is moving to the left or to the right is set to 30 pixels, and the whole method is explained as follows:

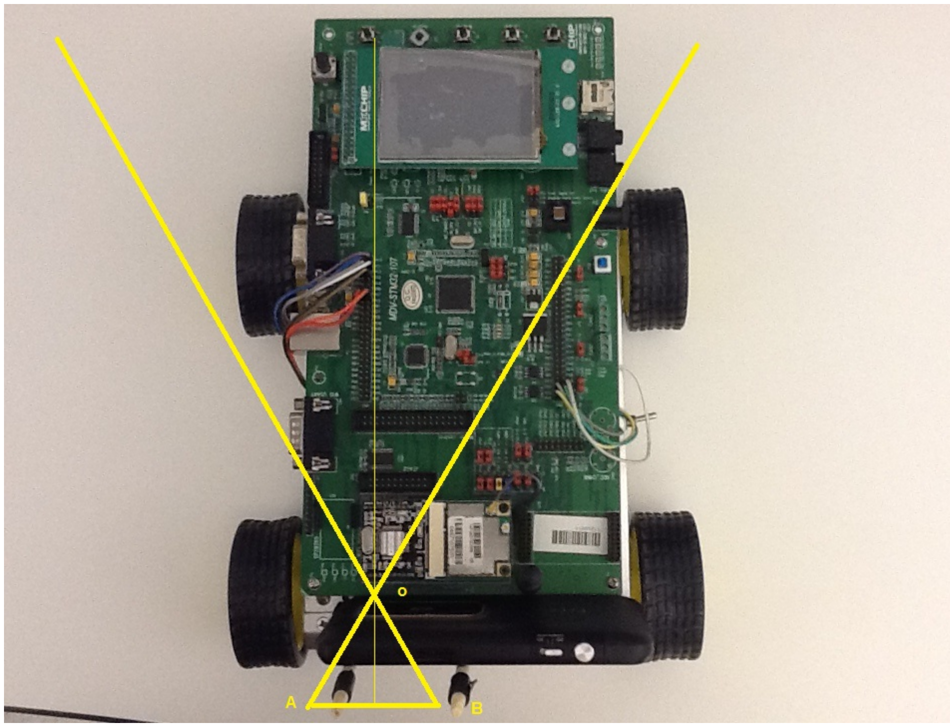


Figure 3.18: Real scene

- If the value of CE is smaller than 30 pixels, we consider that the person is in the range and is not moving left or right.
- If the value of CE is bigger than 30 pixels, and the point C is to the right of point E, then we consider that the person is moving to the left.
- If the value of CE is bigger than 30 pixels, and the point C is to the left of point E, then we consider that the person is moving to the right.

Moving Forward

As mentioned above, we also need a threshold to figure out whether or not a person is moving forward. Here, information about the vertical image resolution taken by the smart phone camera, the vertical range (pixels) of a person's body when he or she is detected in frames, and the vertical angle of view of the camera is required to calculate the threshold.

Still using the HTC EVO 3D as an example, the effective frame in height that can be used to detect a person is 140 pixels, and the real vertical angle of view of a image taken by the smart phone camera is 33.4 degrees.

According to the pinhole camera model, Figure 3.19 shows a real scene and the related mathematical model for calculating whether the person is moving forward or not. In this model, the distance that a person has moved can be calculated through the vertical range (in pixels) of the person's whole body, which is shown in each frame taken by the camera. In the model:

- The line AB stands for a person in the real world, and the line CD is the vertical range (pixels) of the person's whole body which is displayed in a frame. Assuming the person is moving forward, the line A'B' stands for the new position of the person at a moment when he or she is moving, and the line C'D' is the vertical range (pixels) of the person's whole body in a frame which is taken at that moment. The line AA'=BB'=LK stands for the distance that the person has moved. The line OH is the centerline of the triangle OCD. The line OL is the horizontal distance between the camera and the person's original position (AB), while OK is the horizontal distance between the camera and the person's new position (A'B').

- The point O stands for the optical center of the smart phone's camera.

Therefore, in this mode, the difference between C'D' and CD in pixel values can represent the distance that a person has moved. Moreover, by using the value of angles, the value of CH and C'H in pixels, and the original position, the moving distance LK can be calculated with Equation 3.14.

$$\angle COC' = \arctan(\frac{CH}{OH}) - \arctan(\frac{C'H}{OH}) \tag{3.12}$$

$$OL + LK = \frac{BL}{\tan(\angle BOL - \angle COC')} \tag{3.13}$$

$$LK = AA' = BB' = \frac{BL}{\tan(\angle BOL - \arctan(\frac{CH}{OH}) - \arctan(\frac{C'H}{OH}))} - OL \tag{3.14}$$

In our system, the composite robot is controlled to maintain an almost constant distance while a person is moving forward in real time. Thus, we need to find a threshold in pixels. This threshold is used to compare with the value of C'D' in Figure 3.19 when the person has moved forward to a new position.

Here is an example of how to calculate the threshold for detecting whether a person is moving forward or not. According to [37], the average step length of an adult is

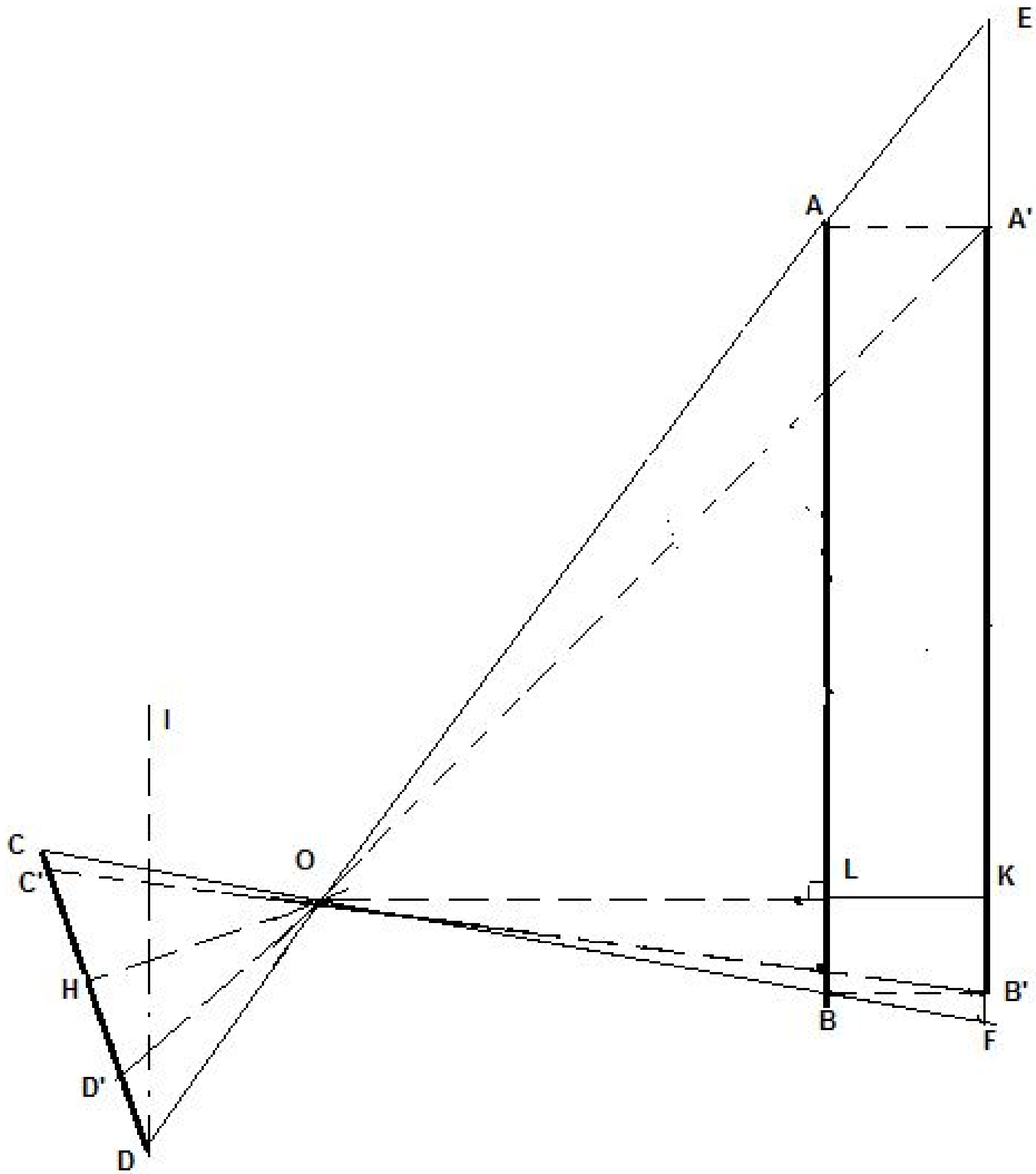


Figure 3.19: [Mathematical model for moving forward

(0.74+0.04/leg length), which means that if a person is stepping forward, the distance of his or her whole body would be about 0.37m from it's original position. Also, we

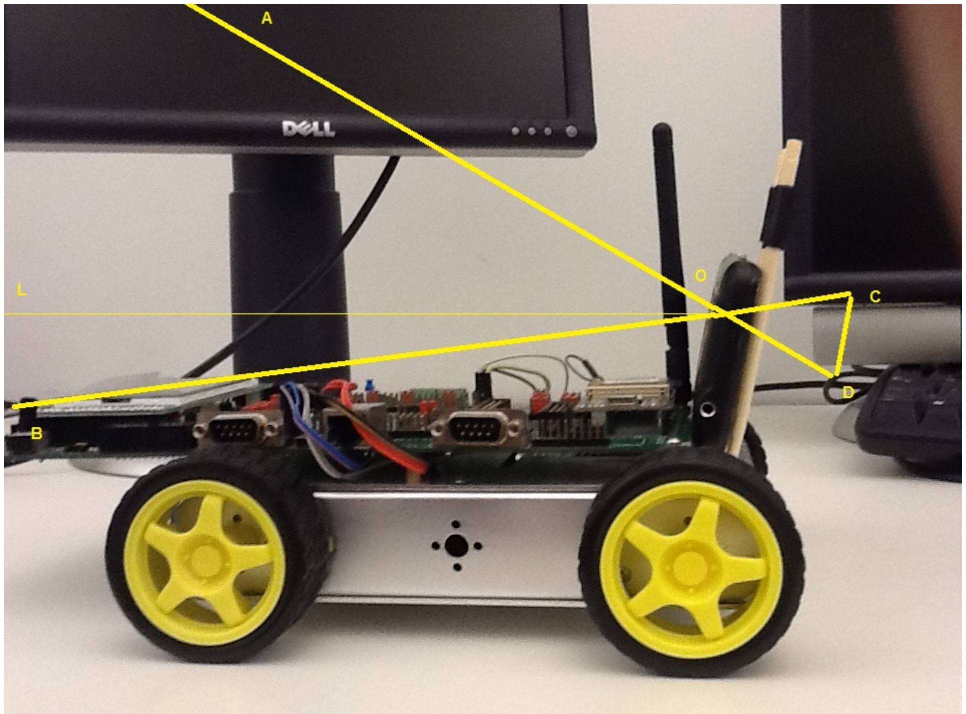


Figure 3.20: Real scene

still assume that the user is 1.8m tall in the example. Therefore, in Figure 3.19, if we consider that the value of line AB is 1.8m, then as mentioned above, the value of angle AOI, which is equal to the angle between the smart phone and the vertical line is 10 degrees, the value of line OL is 2.9m, the value of AL is 1.6, and the value of the line BL is 0.2m. In addition, the value of line AA' is 0.37m, the value of the angle COD is 33.4 degrees, which is the effective vertical angle of view of the camera, and the value of the line CD is 140 pixels. Then the value of the line C'D' is considered as the threshold, which can be used to determine if the person is moving forward. It can be calculated with Equation 3.18.

$$\angle BOB' = \angle COC' = \arctan(\frac{BL}{OL}) - \arctan(\frac{BL}{OL + AA'}) = 0.45 \tag{3.15}$$

$$\angle AOA' = \angle DOD' = \arctan(\frac{AL}{OL}) - \arctan(\frac{AL}{OL + AA'}) = 2.8 \tag{3.16}$$

$$C'D' = \frac{(C'H + HD') * CD}{CH + DH} \tag{3.17}$$

$$C'D' = \frac{\tan(\angle COD/2 - \angle AOA') + \tan(\angle COD/2 - \angle BOB')}{2\tan(\angle COD/2)} * CD = 126(pixels) \tag{3.18}$$

Therefore, in each frame taken by the smart phone's camera, if the vertical value of the person's whole body is smaller than 126 pixels, we consider the person to be moving forward.

3.3.4 Controlling the Robot Car to Follow the Person

In our system, after detecting the direction that the person is moving by using our proposed algorithm introduced above, the smart phone sends the moving command immediately to the chip inside the robot car, and the chip commands the robot car to follow the person.

Figure 3.21 is the robot car we implemented in our system, which was bought from a company called "MXchip Information Technology" in Shanghai. The chip of the robot car, which is marked by a red circle, is programmable. More information about this robot car will be introduced in Chapter 4.

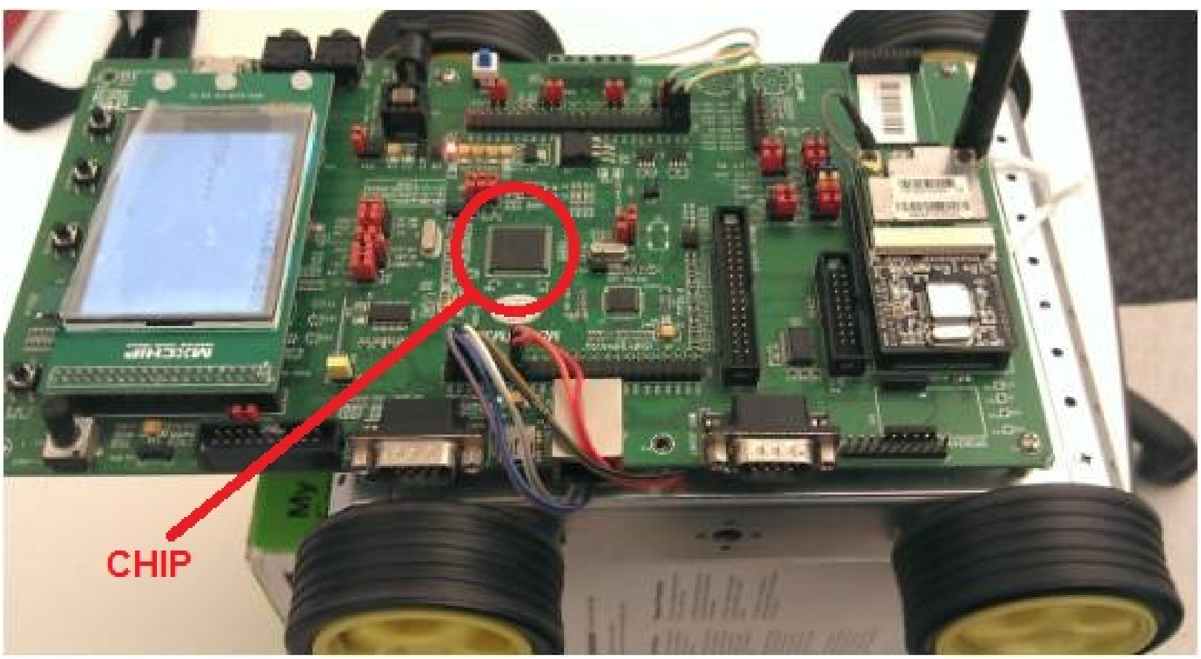


Figure 3.21: The robot car

There are various options for programming the chip of the robot car. For example, a programmed chip can decide how many meters the robot car needs to move, and how many degrees in an angle the robot car need to turn when the person is moving. However, as there will be many parameters to be calculated accurately in this method, it is complicated, and it may be slow in real time. Therefore, we need to find a real time option.

As mentioned in the beginning of this section, the basic concept of our method for following a person is to compare the person's position with thresholds in each frame taken by the smart phone's camera. Then according to the comparative result, the robot car is controlled to move forward in order to keep a constant distance with the person, or it is controlled to turn left or right in order to keep the person in the middle of each frame. From this basic concept, we developed a potential method for programming the chip of the robot car, which is that the chip controls the robot car to keep moving or to keep turning right or left until the chip receives another command from the smart phone.

Therefore, according to the person's position in each fame of the video captured by the smart phone's camera, one control command is sent from the smart phone to the chip, and the chip controls the robot car to respond. The following are some examples to explain how the whole process works:

- If the person is not moving, the chip of the robot car will keep receiving the "Stop"command from the smart phone. Therefore, the robot car will not move.

- If the person is moving left, the chip will keep receiving the "turning left"command from the smart phone. Then, the robot car will keep turning left until the person's center point is detected in the middle of a frame, at which point the robot car will receive a "stop"command. For example, if a person is 1.8m tall and the line CE in Figure 3.17 is smaller than 30 pixels, we consider the person to be not moving.

- If the person is moving right, the chip will keep receiving the "turning right"command from the smart phone. Then, the robot car will keep turning right until the person's center point is detected in the middle of a frame, and the robot car will receive a "stop"command. For example, if a person is 1.8m tall and the line CE in Figure 3.17 is smaller than 30 pixels, we consider the person to be not moving.

- If the person is moving forward, the chip will keep receiving the "forward"command from the smart phone. Then, the robot car will keep moving forward until the vertical value of the person's whole body in a frame is smaller than the threshold, and the chip of the robot car receives a "stop"command. For example, if a person is 1.8m tall, when the line C'D' in Figure 3.19 is bigger than 126 pixels, we consider the person to be not moving.

Chapter 4

Implementation and Results

In this chapter, we give details about the implementation of our composite robot system first. Then, we test the accuracy of our algorithm for visually detecting a moving person's direction. Also, we evaluate our proposed method for activating different web services. In addition, we show examples of the HTC EVO 3D smart phone controlling the robot car to follow a person in real time. Also, more figures are provided to illustrate the outcomes of our system's other functions.

4.1 Implementation

- **Smart phone:** The HTC EVO 3D (Figure 4.1) is the smart phone we use in our system. According to [7], the EVO 3D is an Android OS smart phone and was released by HTC in 2011. Moreover, it has the following features: the EVO 3D utilizes a dual core 1.2 GHz processor; its screen is a 4.3 inch qHD touch screen; the cameras of the EVO 3D is capable of capturing videos in 720p resolution; the version of its Android system is currently 4.0.3.

- **The robot car:** The robot car (Figure 3.21) of our system was developed by the MXchip Information Technology company in Shanghai. This robot car is powered by a pair of 14500 3.7v Li-ion batteries. The robot car's wheels are controlled respectively by four small motors. These motors are connected with a STM32F107 ARM Cortex-M3 Board [15], which incorporates the high-performance ARM Cortex-M3 32-bit RISC core. The ARM core operates at up to 72 MHZ frequency, with 256KB Flash and 64KB RAM internal memory. Also, a 3.2 inch TFT LCD Panel (320*240) with touch screen is mounted on the board. Moreover,

Figure 4.1: The HTC EVO 3D

in order to allow the robot car to be controlled through WIFI, a WIFI module EM380C, which is also designed by the MXchip Information Technology company, is used on the board. The protocol for controlling this robot car through WIFI is TCP/IP, where the IP address of the robot car in our system is "192.168.1.8", and the IP address of the smart phone is "192.168.1.2".

There are five controlling commands for the robot car:

1. Forward: All the motors move forward, and the LCD screen displays 'FORWARD'.
2. Backward: All the motors move backward, and the LCD screen displays 'BACKWARD'.
3. Turn right: The motors on the left side move forward, while the motors on the right side move backward, and the LCD screen displays 'TURN RIGHT'.
4. Turn left: The motors on the left side move backward, while the motors on the right side move forward, and the LCD screen displays 'TURN LEFT'.
5. Stop: All the motors stop turning, and the LCD screen displays 'STOP'.

- **Softwares and Web services:** The face detection algorithm (Haar cascade), face recognition algorithm (Eigenface) and the person's detection algorithm (HOG) we implemented in our system are all from the OpenCV (Open Source Computer Vision) [10], which is a library of programming functions for real time computer vision, and it was developed by Intel®. The OpenCV version we implemented is 2.4.2. The web services we implemented in our system are: Google Calender, the Google Voice Search, and the home automation web service developed by Bassem from the University of Ottawa. Google Voice Recognize is a Google™API that allows devices to access the speech recognition service and understand a user's speech.

- **Smart home devices:** All the smart home devices we used were bought from the HomeSeer [6]. Figure 4.2 from [6] is the smart sensor we implemented in our system to get temperature and luminance information from the smart home. It is a HomeSeer Z-Wave device (HSM100), and it is also a three-in-one sensor including a motion sensor, a temperature sensor and a luminance sensor. Figure 4.3 from [6] is the Z-Wave lamp module we used to control lights in our system. This dimmable lamp module supports ON/OFF and dimming commands, and also includes load-sensing features. Therefore, it not only contains an actuator to turn the light on/off, but also contains a sensor to detect the loading change when the light is turned on/off manually or through automation. If the attached lamp is turned on manually in our system, the Z-Wave lamp module can detect the luminance of the lamp by using the load sensing feature, and change the luminance information on the Home Automation web service.

- **Programming Environment:** The application in the smart phone we designed is programmed in JAVA and C++ by using the Eclipse complier, where the Eclipse SDK version is 3.7.1 , the Android NDK version is R8b and the Android SDK version is 4.0.3 (API 15). The web service we developed is programmed in C sharp and C++ using Microsoft Visual Studio 2010. The robot car is programmed in C using the keil uvision4 IDE and the MDK-ARM [8].

4.2 Experimental Results of Detecting a Moving Person's Direction

We have done several experiments to test the accuracy of our algorithm for visually detecting a moving person's direction in the real world. These experiments are explained

Figure 4.2: The Motion, Temperature and Luminance Sensor

Figure 4.3: The Z-WAVE lamp module

as follows:

- We put the Android OS phone, HTC EVO 3D, on a box as shown in Figure 4.4.

The center point of the camera was 0.2m above the ground.

Figure 4.4: Preparing for the test

- The test environment which is shown in Figure 4.5 contains TVs, chairs, desks, and computers. This is similar to a home environment.
- We invited 8 people to do the experiment (7 men and 1 woman, ages 20 to 28). By using the algorithm, the related thresholds and following distance in our system is calculated for each person according to their height. For example, as calculated in Chapter 3, if a person is 1.8m tall, the minimum following distance is 2.9m from the robot car, and the thresholds are 30 pixels for detecting left or right movements and 126 pixels for detecting forward movements. Then, each person was asked to stand in the middle of the image and the minimum following distance (for 1.8m tall person, it is about 2.9 meters) away from the smart phone. We consider this position to be the original position. After making sure the person is successfully detected in images, we started our experiment. Figure 4.6 illustrates an example of a person standing at the original position.
- During the experiment, each person was asked to move one step left from the original position 10 times, to move one step right from the original position 10

Figure 4.5: The background of the test environment

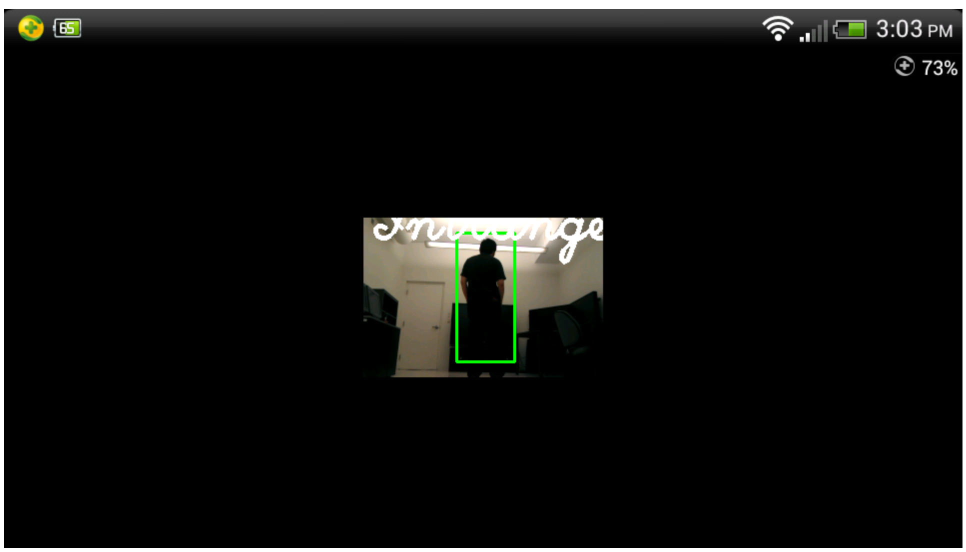
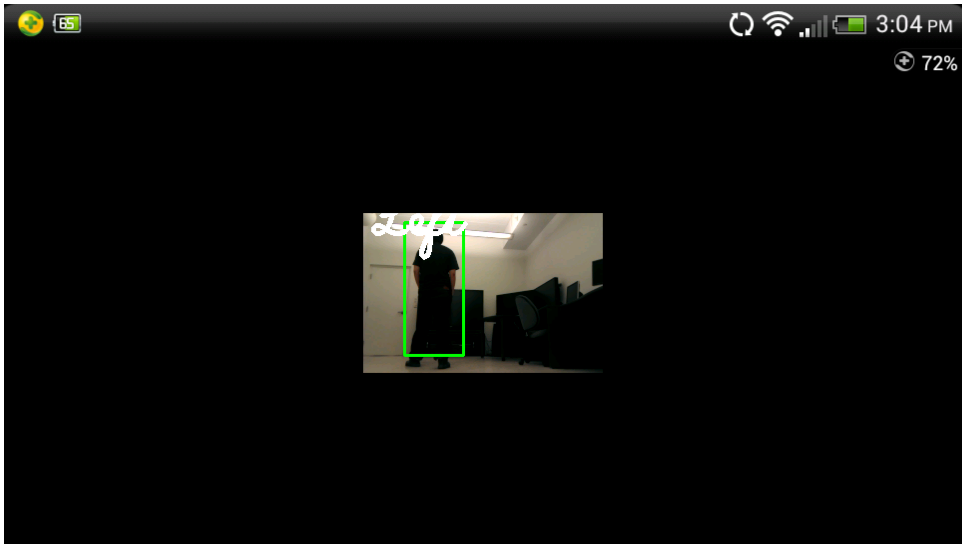
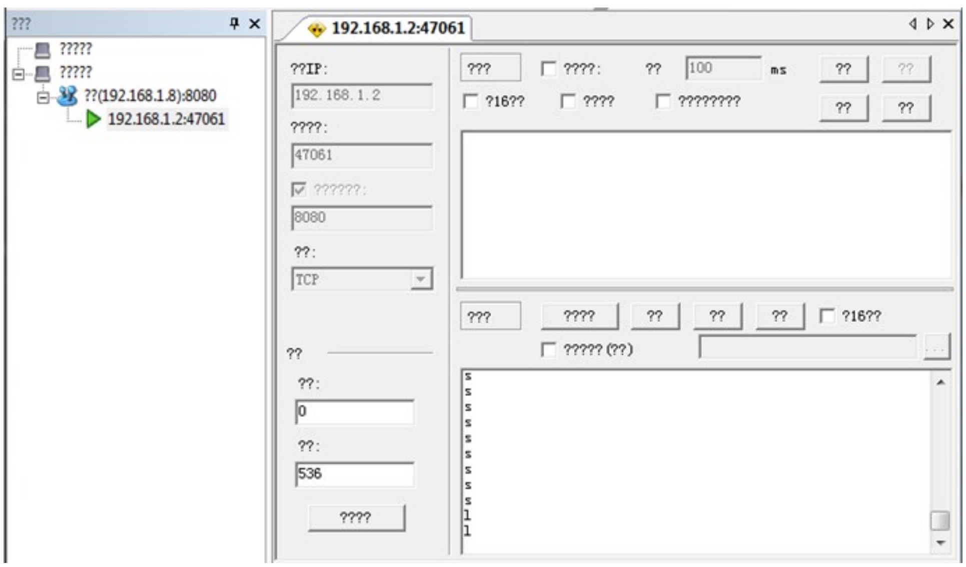


Figure 4.6: An in range result from the smart phone's screenshot

times, to move one step forward from the original position 10 times, and to move two steps forward from the original position 10 times. In order to monitor the results, we used a TCP debugging tool to act as the robot car of our system to display each command that the smart phone sent. If the TCP debugging tool received a correct command according to the direction of the moving person, it



(a) An moving left result from the smart phone's screenshot

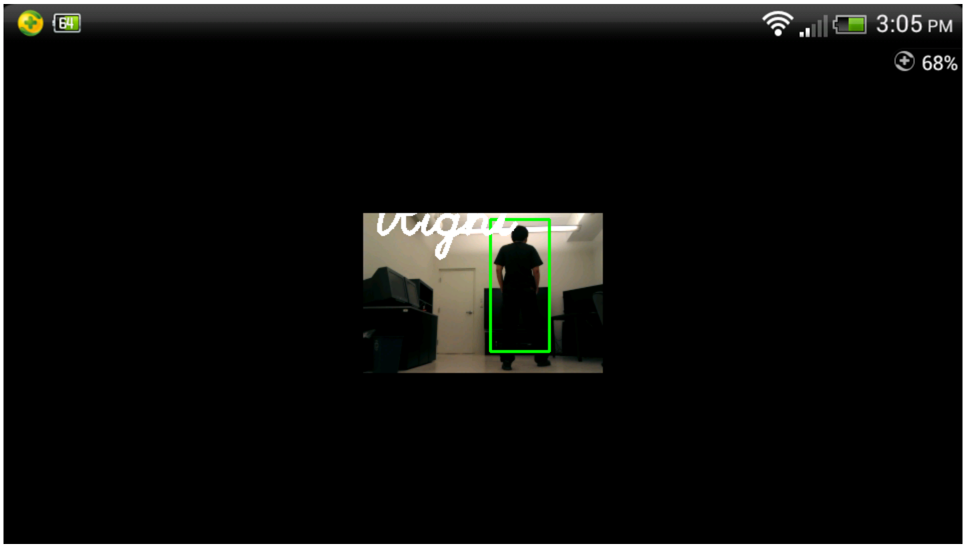


(b) Moving left results displayed through the TCP degugging tool

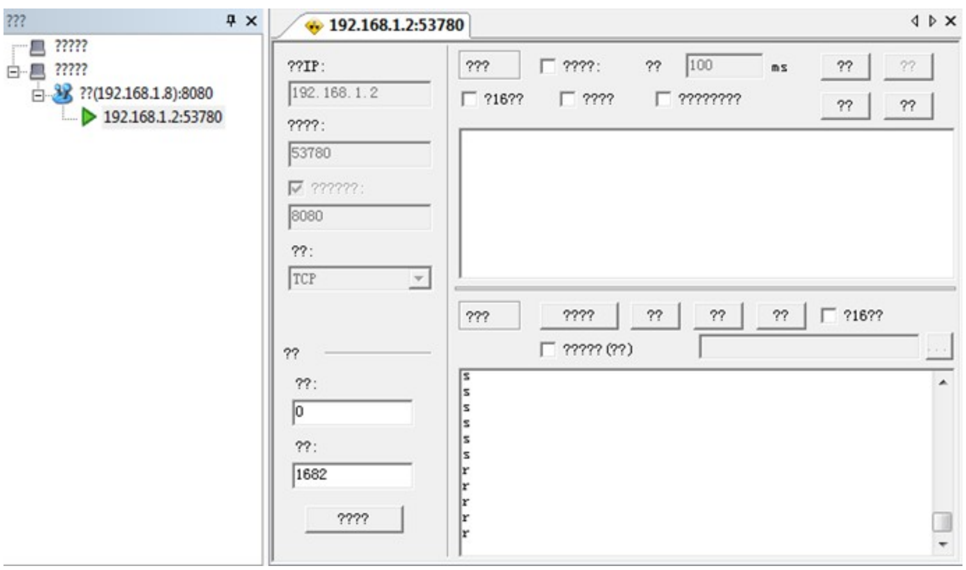
Figure 4.7: An example that a person is moving left

was considered to be a successful detection.

Figure 4.7 illustrates an example of a person moving a step left from the original position during our experiment. In response, the TCP debugging tool began to receive turning left commands 'l'instead of the stop command 's'from the smart phone. Figure 4.8 is an example of a person taking a step to the right from the



(a) A moving right result from the smart phone's screenshot



(b) Moving right results displayed through the TCP debugging tool

Figure 4.8: An example of a person moving right

orginal position. In response, the TCP debugging tool began to receive turning right commands 'r'from the smart phone.

Table 4.1 displays the related thresholds and the minimum following distance according to each person's height, while Table 4.2 is the experimental results of each person.

Table 4.3 shows the average successful rate of detection for left, right, one step forward, and two steps forward movements. From this table, we can see that our algorithm has a high performance when detecting whether a person is moving left or right. The average successful rate of detecting left and right movement is 95% and 96.25% respectively. Although the average successful rate of detecting forward movements is only 72.5% if a person is moving one step forward, it still works well if the person moves more than one step.

Table 4.1: The thresholds and minimum distance for following each person

Person Number	Height of the Person	Threshold for Detecting Left/Right	Threshold For Detecting Forward	The Minimum Following Distance
1	1.80m	30pixels	126pixels	2.9m
2	1.85m	29pixels	126pixels	2.9m
3	1.75m	31pixels	125pixels	2.8m
4	1.75m	31pixels	125pixels	2.8m
5	1.96m	27pixels	126pixels	3.1m
6	1.80m	30pixels	126pixels	2.9m
7	1.83m	30pixels	126 pixels	2.9m
8	1.72m	31pixels	125pixels	2.75m

4.3 Testing Results of People Detection and Face Recognition

As mentioned above, we implemented the HOG algorithm in OpenCV to detect a person from an image. Figure 4.9 shows the detection results, in which a person's whole body is marked by a green square. As we can see, although these results are not so accurate, a person's whole body is still successfully detected, and it satisfies our goal of a person's detection.

As was also mentioned above, we developed a web service by implementing the Eigenfaces algorithm to let the Android phone recognize the user. After detecting a user's face,

Table 4.2: The accuracy of visually detecting a moving person's directionally

Person Number	Detecting Left Successfully (n/10)	Detecting Right Successfully (n/10)	Detecting one step Forward Successfully (n/10)	Detecting two step Forward Successfully (n/10)
1	10	10	8	10
2	10	10	5	10
3	10	10	7	10
4	10	10	7	10
5	7	9	8	10
6	10	9	9	10
7	9	9	5	10
8	10	10	9	10

Table 4.3: The average rate of successful detection of a moving person's direction

The Average Rate of Successful Left Movement Detection (%)	The Average Rate of Successful Right Movement Detection (%)	The Average Rate of Successful One Step Forward Detection(%)	The Average Rate of Successful Two Step Forward Detection(%)
95	96.25	72.5	100

the smart phone sends a screenshot to the web service, and the web service will return the user's name. Figure 4.10 shows several results of face recognition with our system. From these results, we can find that the Eigenfaces algorithm for face recognition actually satisfies our goal of recognizing the user.

4.4 Evaluation of Activation Different Web Services

In this section, we evaluated the method we developed for activating Google Voice RecognizeIntent under the Robot Mode of our system with a questionnaire. We invited 8 people (7 men, 1 woman, age 20-28) to finish this evaluation.

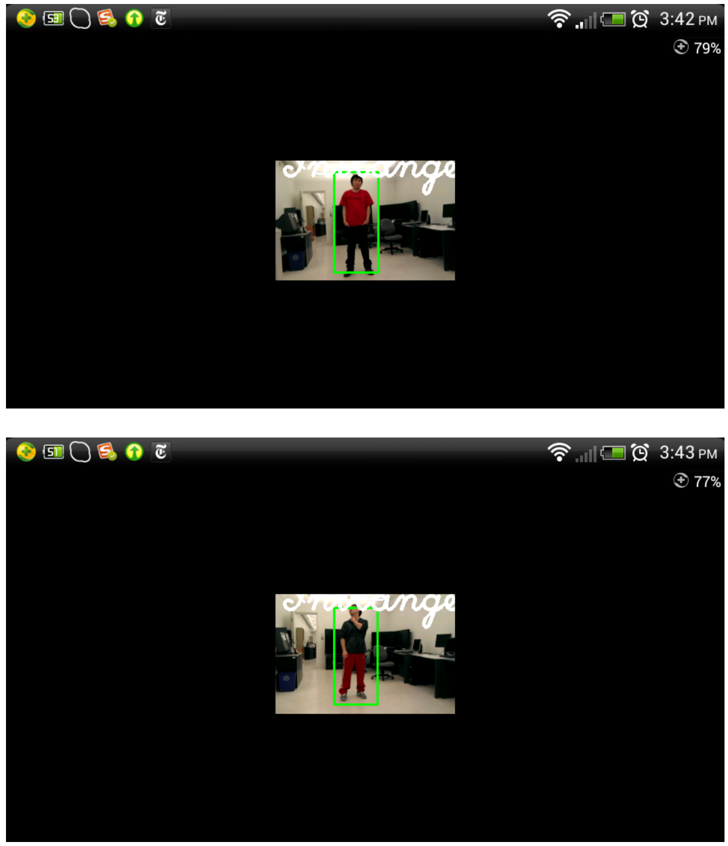


Figure 4.9: Examples of results from HOG person's detection

Before completing the questionnaire, a user is required to sit on a chair while doing anything she wanted (Figure 4.11) while our robot is placed beside her, and it is under its Robot Mode. Then, the user is able to face the smart phone's camera if he or she wants to do a voice command, as shown in figure 4.12.

After experiencing the whole process, the user is required to answer several questions and to give some related scores. The highest score a user can give is 10, and the lowest score is 1. The questionnaire is as follows:

1. How natural it feels to start a conversation with our system.

2. How good of an idea is it our robot to be able to recognize the user.

From the evaluation results, people showed a positive attitude toward the method we developed for activating the voice recognition (7 people were highly satisfied and 1 was satisfied). Adding the face recognition algorithm into our system is also a good idea according to the results of the questionnaire (8 people were highly satisfied).

Figure 4.10: Examples of face recognition

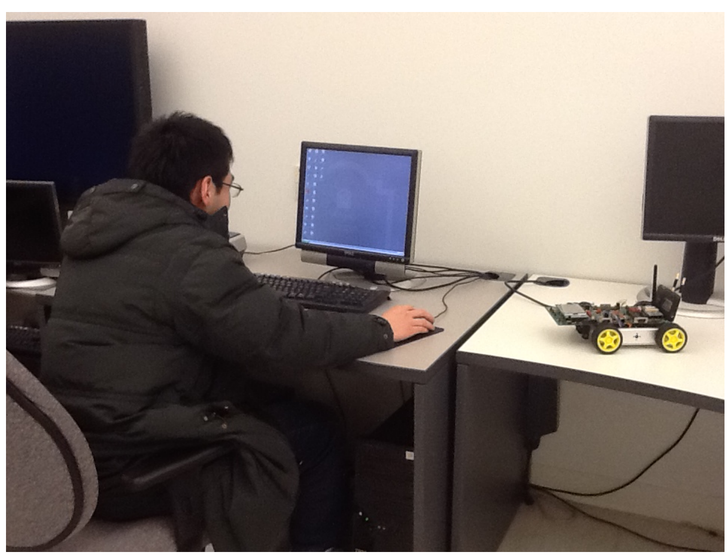


Figure 4.11: An evaluation experiment

Figure 4.12: An evaluation experiment

4.5 Other Outcomes of Our System

Figure 4.13 illustrates the Portable Mode of our application on the HTC EVO 3D. Users can directly read their schedule for the day through Google Calendar. Three options: take notes, home controller, and Robot Mode are listed at the bottom. The take notes option allows users to take a note manually and save in their Google Calendar. The home controller option allows users to control their smart home devices and to monitor their smart home, as shown in Figure 4.14.

Figure 4.15 is an example, in which a face was detected under the Robot Mode, and the Android phone sent the screenshot to the face recognition web service. Then the web service extracted the face, recognized the face by using the Eigenfaces algorithm, and returned the name of the user. After this process, the Google Voice RecognizeIntent was activated.

Figure 4.16 is a real scene, in which our robot is following a person.

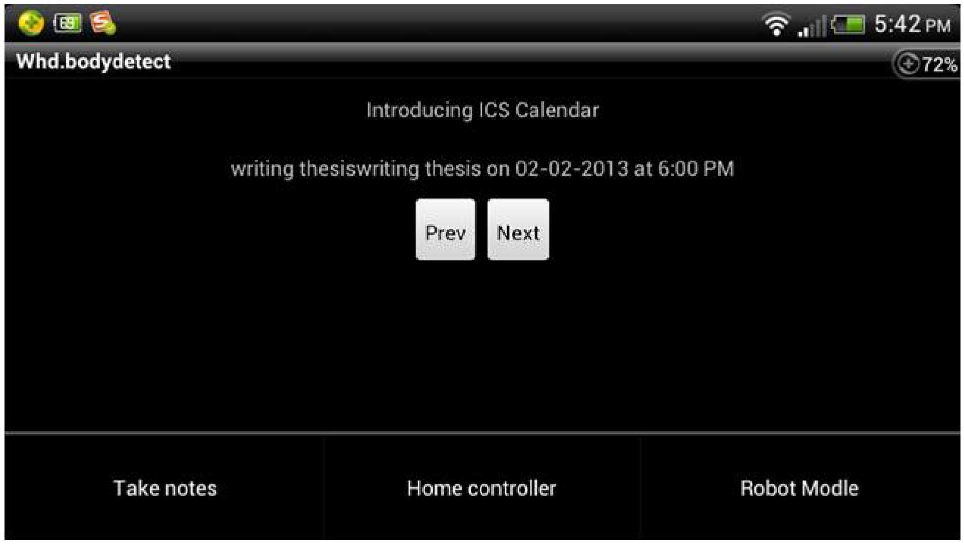


Figure 4.13: The main activity of the Portable Mode

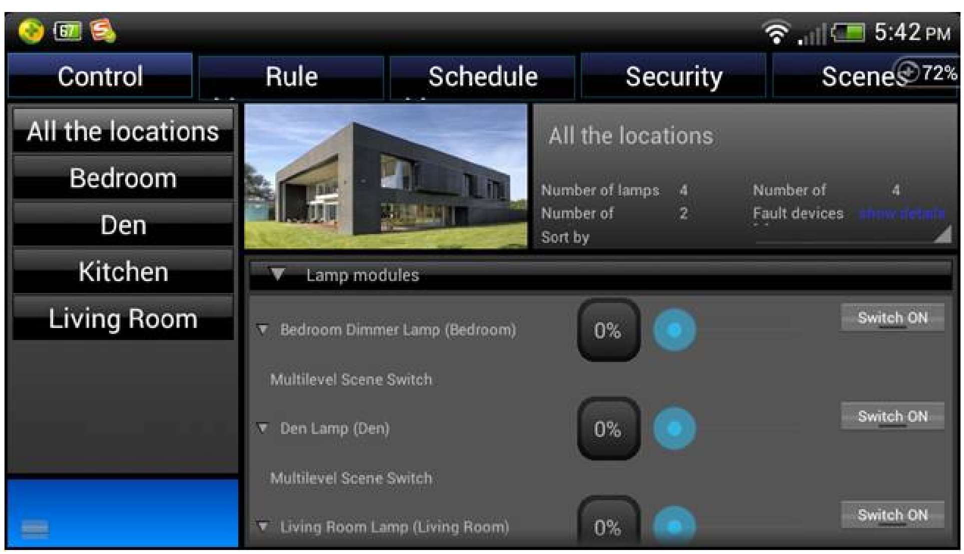


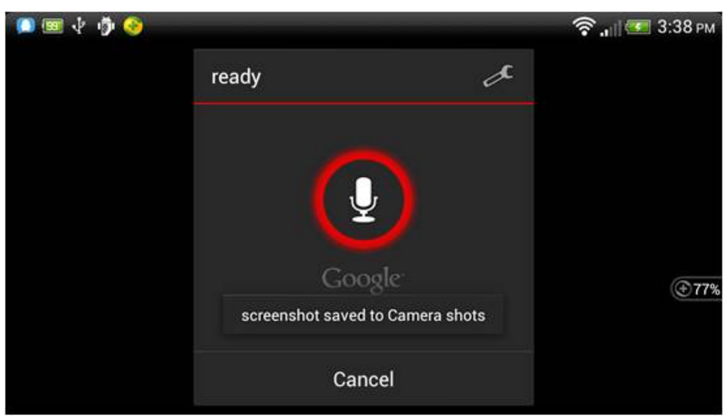
Figure 4.14: The home controller option

(a) The screenshot when the smart phone detects a face

(b) Tthe face area extracted from the screenshot

```
<string xmlns="uploadimage">david</string>
```

(c) Name of the user



(d) Google Voice RecognizeIntent

Figure 4.15: An example of activating the Google Voice RecognizeIntent

Figure 4.16: An example of following a person in the real world

Chapter 5

Conclusion and Future Work

5.1 Conclusion

The home plays as an important role in people's lives, and people always try their best to make their home more comfortable to live in. With the fast development of smart home technologies, we have developed many smart home devices that contain sensors and actuators to serve people at home. We also have many communicating technologies that can be used to act as bridges between smart home devices and interfaces. Moreover, there are many technologies that can be used as interfaces in smart homes. However, computers, tablets, smart phones and robots, all of which are widely used as interfaces in smart homes, have drawbacks. In brief, computers, laptops and robots are not convenient to carry when people leave their homes, and smart phones and tablets still need people to hold them at home when people want to control their smart home devices.

In order to make the control of a smart home more efficient, and to make people's lives easier with an interface, we proposed an idea that is to make composite robots as interfaces in smart homes. A composite robot consists of an Android smart phone and a WIFI robot car. When a person is not at home, he or she can still receive information such as room temperature and control home devices such as lights through his or her smart phone. When a person is at home, she can put her phone on a WIFI robot car. This robot assistant can understand the user's voice-commands in order to take personal notes by using Google Calendar, control the smart home devices by using the Home Automation web service, and follow the user visually. This is a software/hardware solution, which would solve the high cost and portable problems of a robot and the lack of self-movement capability of a smart phone.

In order to activate the Google Calendar and the Home Automation web service through voice-commands under the Robot Mode, we developed a method that implement the Haar-cascade from the OpenCv for the Android phone to be aware of when it should activate the Google Voice search it self. Moreover, we also developed a face recognition web service by implementing the Eigenfaces algorithm to identify the user before activating the Google Voice RecognizeIntent.

We successfully let our system follow a person visually under the Robot Mode by implementing the Histogram of Oriented Gradient (HOG) people detection from the OpenCv to detect a person and using our algorithm to detect the direction that a person is moving. Then, according to the algorithm, the person's position in each frame is compared with thresholds to decide whether the robot car should move forward, turn left or turn right. We also gave an example for calculating these thresholds. We programmed the chip of the robot car to control the robot car to keep moving forward or keep turning right or left until the chip received another command from the smart phone.

Experiments were conducted to test the accuracy of our algorithm for detecting a moving person's direction in real time. Moreover, we evaluated the proposed method of activating the Google Voice Recognize through face detection. The experimental results show that our proposed system is well accepted.

5.2 Future Work

Finding a method for following a particular person is the most important future improvements of our system. It is hard to identify the person according to his or her back view, while if our system is following a person, the smart phone can only see the person's back. However, if our proposed composite robot is able to follow a particular person visually, our composite robot can be more widely used, and it will be a new robotic idea in the future. Therefore, in a potential future, people just need to bring their smart phones, and these can become robot assistants wherever a body can be found. This is a brand new vision in robotics and smart phones.

Improving the person's detection method is another important future work. Although HOG has proved to be a successful method for detecting a person, it still has problems when it is implemented in our system. One problem is that the vertical range of a person's body fluctuates a lot. However, this vertical range is used to determine whether or not the person is moving forward. Also, that is why the successful rate of detection a person moving one step forward is 72.5%.

As several parameters such as thresholds in our algorithm for detecting a moving person's direction depends on the person's height, our composite robot would be better if it could adjust these parameters itself according to the user's information after recognizing the user. Also, once the robot recognizes the user, more personal preference could be added in the future, for example playing the user's favourite songs.

In the future, we intend to improve the robot car. First of all, we will add more sensors on the robot car to avoid obstacles. Secondly, the robot's body will more closely resemble a human. According to the evaluation paper [14], people are more accepting of humanoid robots, especially if they live with them. Ideally, the robot will be able to move upstairs and downstairs to be used in complicated environments.

At last, we plan to add more multimedia components. For example, the smart phone can play Youtube on local media according to the user's command, and our composite robot can provide weather services for the user.

References

[1] http://www.ee.ncu.edu.tw/~fuzzylab/ Accessed January 30, 2013.

[2] 35mm film. http://en.wikipedia.org/wiki/35_mm_film Accessed January 22, 2013.

[3] A camera's angle of view. http://en.wikipedia.org/wiki/Angle_of_view Accessed February 18, 2013.

[4] Evo 3d camera. http://www.camerafv5.com/database/device.php?name=htc_htc_evo_3d_x515a Accessed March 4, 2013.

[5] Google calendar. http://en.wikipedia.org/wiki/Google_Calendar Accessed January 22, 2013.

[6] Homeseer. http://www.homeseer.com/ Accessed February 24, 2013.

[7] Htc evo 3d. http://en.wikipedia.org/wiki/HTC_Evo_3D Accessed January 22, 2013.

[8] Keil mdk-arm. http://www.keil.com/arm/mdk.asp Accessed February 22, 2013.

[9] Lg roboking. http://en.wikipedia.org/wiki/LG_Roboking Accessed January 22, 2013.

[10] Opencv. http://opencv.willowgarage.com/wiki/ Accessed February 22, 2013.

[11] Pinhole cameras. http://en.wikipedia.org/wiki/Z-wave Accessed January 22, 2013.

[12] ragon naturallyspeaking. http://www.nuance.com/dragon/index.htm Accessed January 30, 2013.

[13] Recognizerintent. http://developer.android.com/reference/android/speech/RecognizerIntent.html Accessed January 22, 2013.

[14] Robot nao. http://www.aldebaran-robotics.com/en/Home/welcome.html?language=en-GB Accessed January 30, 2013.

[15] Stm32f107. http://www.embedinfo.com/en/list.asp?id=57 Accessed February 24, 2013.

[16] Sumsung navibot. http://www.samsung.com/my/navibot/index.html Accessed January 22, 2013.

[17] Surface. http://www.microsoft.com/surface/en-CA/surface-with-windows-rt/home?WT.mc_id=MSCOM_EN_CA_HP_FEATURE_132F1ENCA31193 Accessed January 22, 2013.

[18] X10. http://en.wikipedia.org/wiki/X10_(industry_standard) Accessed January 30, 2013.

[19] Z-wave. http://en.wikipedia.org/wiki/Z-wave Accessed January 22, 2013.

[20] Zigbee. http://en.wikipedia.org/wiki/Zigbee Accessed January 22, 2013.

[21] M. Aiello. The role of web services at home. 2005.

[22] T. Akimoto and N. Hagita. Introduction to a network robot system. In *Intelligent Signal Processing and Communications, 2006. ISPACS '06. International Symposium on*, pages 91 –94, dec. 2006.

[23] M. Al-Qutayri, H. Barada, S. Al-Mehairi, and J. Nuaimi. A framework for an end-to-end secure wireless smart home system. In *Systems Conference, 2008 2nd Annual IEEE*, pages 1–7. IEEE, 2008.

[24] S. Baeg, J. Park, J. Koh, K. Park, and M. Baeg. Robomaidhome: A sensor network-based smart home environment for service robots. In *Robot and Human interactive Communication, 2007. RO-MAN 2007. The 16th IEEE International Symposium on*, pages 182 –187, aug. 2007.

[25] S. Baeg, J. Park, J. Koh, K. Park, and Moon-Hong Baeg. Building a smart home environment for service robots based on rfid and sensor networks. In *Control, Automation and Systems, 2007. ICCAS '07. International Conference on*, pages 1078 –1082, oct. 2007.

[26] N. Bellotto and Huosheng Hu. Vision and laser data fusion for tracking people with a mobile robot. In *Robotics and Biomimetics, 2006. ROBIO '06. IEEE International Conference on*, pages 7 –12, dec. 2006.

[27] C. Bisdikian. An overview of the bluetooth wireless technology. *Communications Magazine, IEEE*, 39(12):86 –94, dec. 2001.

[28] M. Broxvall, M. Gritti, and A. Saffiotti. Peis ecology: Integrating robots into smart environments. In *Proc of the IEEE Int Conf on Robotics and Automation*, pages 212–218, 2006.

[29] G. Cozzolongo, B. De Carolis, and S. Pizzutilo. Social robots as mediators between users and smart environments. In *Proceedings of the 12th international conference on Intelligent user interfaces*, IUI '07, pages 353–356, New York, NY, USA, 2007. ACM.

[30] S. Das D. Cook. *Smart Environments: Technology, Protocols, and Applications.* Wiley Inter-Science, 2004.

[31] N. Dalal and B. Triggs. Histograms of oriented gradients for human detection. In *Computer Vision and Pattern Recognition, 2005. CVPR 2005. IEEE Computer Society Conference on*, volume 1, pages 886 –893 vol. 1, june 2005.

[32] S.K. Das, D.J. Cook, A. Battacharya, E.O. Heierman III, and T.Y. Lin. The role of prediction algorithms in the mavhome smart home architecture. *Wireless Communications, IEEE*, 9(6):77–84, 2002.

[33] Eberhart and Y. Shi. Particle swarm optimization: developments, applications and resources. In *Evolutionary Computation, 2001. Proceedings of the 2001 Congress on*, volume 1, pages 81 –86 vol. 1, 2001.

[34] D.M Gavrila. The visual analysis of human movement: A survey. *Computer Vision and Image Understanding*, 73(1):82 – 98, 1999.

[35] N. Hirai and H. Mizoguchi. Visual tracking of human back and shoulder for person following robot. In *Advanced Intelligent Mechatronics, 2003. AIM 2003. Proceedings. 2003 IEEE/ASME International Conference on*, volume 1, pages 527 – 532 vol.1, july 2003.

[36] S.S. Intille. Designing a home of the future. *Pervasive Computing, IEEE*, 1(2):76 –82, april-june 2002.

[37] J. JudgeRoy, B. Davis, and S. Õunpuu. Step length reductions in advanced age: the role of ankle and hip kinetics. *The Journals of Gerontology Series A: Biological Sciences and Medical Sciences*, 51(6):M303, 1996.

[38] J. Kennedy and R. C. Eberhart. *Swarm intelligence*. Morgan Kaufmann Publishers Inc., San Francisco, CA, USA, 2001.

[39] C. Kidd, R. Orr, G. Abowd, C. Atkeson, I. Essa, B. MacIntyre, E. Mynatt, T. Starner, and W. Newstetter. The aware home: A living laboratory for ubiquitous computing research. *Cooperative buildings. Integrating information, organizations, and architecture*, pages 191–198, 1999.

[40] et al K.Okabe, H.Mizoguchi. Olda based arbitrary pattern recognition system with interactive trainability. In *Proceeding of SI2002*, pages 159–160, 2002.

[41] D. Liu and D. Xian. Home environmental control system for the disabled. In *Proceedings of the 1st international convention on Rehabilitation engineering & assistive technology: in conjunction with 1st Tan Tock Seng Hospital Neurorehabilitation Meeting*, pages 164–168. ACM, 2007.

[42] A. Louloudi, A. Mosallam, N. Marturi, P. Janse, and V. Hernandez. Integration of the humanoid robot nao inside a smart home: A case study. In *Proceedings of the Swedish AI Society Workshop (SAIS). Linköping Electronic Conference Proceedings*, volume 48, pages 35–44, 2010.

[43] F. Lu and G.H. Tian. The zigbee based wireless sensor and actor network in intelligent space oriented to home service robot. *International Journal of Communications, Network and System Sciences*, 5:280–285, 2012.

[44] K. Morioka, J. Lee, and H. Hashimoto. Physical agent for human following in intelligent sensor network. In *Intelligent Robots and Systems, 2002. IEEE/RSJ International Conference on*, volume 2, pages 1234 – 1239 vol.2, 2002.

[45] J. Parra, M.A. Hossain, A. Uribarren, E. Jacob, and A. El Saddik. Flexible smart home architecture using device profile for web services: a peer-to-peer approach. *International Journal of Smart Home*, 3(2):39–56, 2009.

[46] A. Sanfeliu, N. Hagita, and A. Saffiotti. Network robot systems. *Robotics and Autonomous Systems*, 56(10):793 – 797, 2008. ¡ce:title¿Network Robot Systems¡/ce:title¿.

[47] S. Sugano and Y. Shirai. Robot design and environment design - waseda robot-house project. In *SICE-ICASE, 2006. International Joint Conference*, pages I–31 –I–34, oct. 2006.

[48] M.A. Turk and A.P. Pentland. Face recognition using eigenfaces. In *Computer Vision and Pattern Recognition, 1991. Proceedings CVPR'91., IEEE Computer Society Conference on*, pages 586–591. IEEE, 1991.

[49] P. Viola and M. Jones. Rapid object detection using a boosted cascade of simple features. In *Computer Vision and Pattern Recognition, 2001. CVPR 2001. Proceedings of the 2001 IEEE Computer Society Conference on*, volume 1, pages I–511 – I–518 vol.1, 2001.

[50] W. Wang and J. Chang. Implementation of a mobile robot for people following. In *System Science and Engineering (ICSSE), 2012 International Conference on*, pages 112 –116, 30 2012-july 2 2012.

[51] K. Werner, J. Oberzaucher, and F. Werner. Evaluation of human robot interaction factors of a socially assistive robot together with older people. In *Complex, Intelligent and Software Intensive Systems (CISIS), 2012 Sixth International Conference on*, pages 455–460. IEEE, 2012.

[52] T. Yamazaki. Ubiquitous home: real-life testbed for home context-aware service. In *Testbeds and Research Infrastructures for the Development of Networks and Communities, 2005. Tridentcom 2005. First International Conference on*, pages 54–59. IEEE, 2005.

[53] M.S. Yousuf and M. El-Shafei. Power line communications: An overview - part i. In *Innovations in Information Technology, 2007. IIT '07. 4th International Conference on*, pages 218 –222, nov. 2007.

[54] J. Zhang, Y. Yan, and M. Lades. Face recognition: eigenface, elastic matching, and neural nets. *Proceedings of the IEEE*, 85(9):1423 –1435, sep 1997.

[55] Y. Zheng and Y. Meng. *Robot Vision: New Research*. Nova Science Pub Inc, 2008.