School of Computer Science and Engineering

Faculty of Engineering

The University of New South Wales

Smart Home Automation Through Voice and Gesture Control

by

Jake Palandri

Thesis submitted as a requirement for the degree of

Bachelor of Engineering in Software Engineering

Submitted: April 2024

Supervisor: Prof. Claude Sammut

Student ID: z5313097

Abstract

This report aims to discuss existing literature and determine the future course of action for the progression of the thesis, alongside the consideration of project-dependent preparations to supplement the in-progression honours thesis, as a requirement of software engineering.

Existing gaps within the literature consisted of gaps within the integration of smart home technologies with computer vision techniques for home automation. There is a significant amount of research into human action recognition and behavioural monitoring in ambient assisted living environments for the care of the elderly and disabled but none covers the integration of this into smart home automation and predictive device control.

Project-dependent preparations were determined early on, selected in conjunction with the main thesis aims, these being the implementation of computer vision software for human action recognition, training an AI model to predict user movements and control devices accordingly, and developing a front-end web interface for manual device control and gesture registering.

Overall, the preparations aim to support the thesis aims and assist in the development of knowledge to achieve the outcomes whilst allowing for the research to be undertaken to the highest possible standard.

Preliminary progress involved the trialing of the Robot Operating System (ROS) with the goal of integrating a Kinect depth sensor with ROS and controlling an external device through data from the Kinect, as a proof of concept for the project.

The planned progress was put forth to detail the plans for future progress of the work across thesis B and C. The current outline suggests week-by-week progress of the thesis aims outlined above, with each step completed within two weeks and evaluation of the system completed by week 8 of term 2. This is subject to change, determined by weekly progress.

Acknowledgements

The author acknowledges the assistance from their supervisor, Professor Claude Sammut, as being integral to their progress across Thesis A. The author also acknowledges the use of this template for LaTeX, graciously provided by the School of Computer Science and Engineering.

Abbreviations

AAL Ambient assisted living

AC Alternating Current

AI Artificial Intelligence

ASCII American Standard Code for Information Interchange

BMS Behaviour Monitoring System

CSA Connectivity Standards Alliance

CV Computer Vision

DVD Digital Video Disc

Distro Distribution

ECHO IV Electronic Computing Home Operator

IAI Institute of Artificial Intelligence, University of Bremen

IoT Internet of Things

IR Infrared

JS JavaScript

LED Light Emitting Diode

ML Machine Learning

MQTT Messaging Queuing Telemetry Transport

PIR Passive Infrared

rclpy ROS Client Library for Python

RGB Red, Green and Blue

RGB-D Red, Green, Blue and Depth

ROS Robot Operating System

STT Speech To Text

TS TypeScript

USD United States Dollars

VCR Videocassette Recorder

YOLO You Only Look Once

Contents

List of Figures

List of Tables

Chapter 1

Introduction

Smart home technology has seen an exponential rate of growth over the last century. The technology has evolved from simple home appliances, to automatic control systems for a wide range of home devices. In particular, each iteration of new smart home technology has evolved to be more user-friendly and more integrated with the user's daily life, freeing up time for more leisure activities, while automating menial tasks.

This report will cover existing literature and the investigation into the future of smart home technology, considering computer vision and human action recognition techniques for the purpose of home automation. Additionally, the implementation of behaviour monitoring for ambient assisted living environments is discussed with its potential for integration into predictive home control in mind.

The plan for the continuation of this thesis will be outlined with a timeline for the completion of the project, covering each of the main functions that are intended to be implemented into the demonstration environment. The progress made during Thesis A will be discussed, including the integration of the Kinect depth sensor with ROS 2, and the preliminary results from the test environment.

Each chapter of the report covers the topics as follows:

- Chapter **??** explains the background for this thesis and related research.
- Chapter **??** outlines the plan for Thesis B and Thesis C.
- Chapter **??** explains the progress that has already been made in Thesis A, preliminary results from a test environment and the proposed technolgy stack.
- Chapter **??** summarises the report.

Chapter 2

Literature Review

2.1 Background

2.1.1 History of the Smart Home

The evolution of smart homes can be traced back to the early 1900s when the introduction of various electric household appliances promised to reduce the time spent on mundane household chores and free up more time for leisure, revolutionising domestic life. Devices such as the electric mixer and iron, the world's first refrigerator in 1913, and the pop-up toaster in 1919, emerged in this period, bringing about the beginnings of a more automated home environment, with the following decades innovating accordingly (?, ?).

In 1966, there was a significant leap forward in home automation technology when Jim Sutherland, an engineer from Pittsburgh, Pennsylvania created the ECHO IV. The Electronic Computing Home Operator (ECHO IV) was the first device designed specifically for home automation and was hand-crafted with surplus electronic parts (?, ?). With the ability to compute shopping lists, control home temperature, limit children's television time with questionnaires and even tell the weather, this was the first glimpse at a whole-home computerised system that was capable of automating every element of home life.

The following year saw the introduction of Honeywell's Kitchen Computer. A computer designed for housewives to use in the kitchen as a recipe storage device with a built-in chopping board. At $10,000 United States Dollars (USD) in 1967 (nearly $100,000 USD today when accounting for inflation), this recipe storage device which had no display, was well described as "amazingly beautiful and hopelessly impractical" (?, ?, ?, ?). No units were ever sold. Despite its failure on the open market, the device received a lot of attention and the public began to imagine a future where homes would be interactive.

Eight years later, in 1975, a group of engineers from Scotland released the X10 protocol. Capable of controlling up to 256 devices on a single circuit, X10 sent messages to devices through a property's existing Alternating current (AC) electrical wiring. At the time, this was an efficient way to send basic signals through large spaces before the widespread adoption of wireless technology in all electronic devices. It later advanced to be controllable from a computer, enabling users to schedule events and run decision-based sequences. The protocol formed the basis for many domestic control installations for several decades and was one of the first protocols to completely cover the home automation spectrum, with power, security and lighting (?, ?).

Throughout the 1980s and 1990s, the idea of having robots as companions was solidified in popular culture through science fiction movies. During this same period, advancements in battery technology and the rapid decrease in the size of microprocessors meant that home robots became achievable, and by 1996, the Electrolux Trilobite was released, the world's first robot vacuum (?, ?). This class of device is now a quintessential smart home product and was a turning point for the industry. The robot vacuum cleaner heralded the integration of the first wave of smart home devices that were not hard-wired into the building. Along with this, the launch of the first-ever internet-connected refrigerator from LG, in the year 2000, demonstrated the increasing integration of technology into everyday household appliances (?, ?).

This brings us to the current century, where, in the early 2000s, technology began to boom. The home computer had become commonplace and the internet had become more accessible and understandable to the general public. More smart devices, like speakers that speak to you about news headlines and weather or act as an alarm, began to appear on store shelves at affordable price points and home automation became a realistically achievable goal.

Today, smart homes are now a reality, they are not exclusively for the eccentric and wealthy. They provide homeowners comfort, security, energy efficiency and convenience at all times, regardless of whether they are home or not. Through the use of innovative technology, homeowners can turn their homes into state-of-the-art machines that can be controlled and monitored from anywhere in the world.

It is clear to see that with every iteration of the development of increasingly quasi-intelligent smart devices, early adopters were promised a more comfortable lifestyle with menial tasks being delegated to machines, saving more time for leisure. With the beginnings of electric home appliances reducing the time required to dedicate to cooking and cleaning, through to interactive devices capable of holding simple conversations with users, this rapidly growing technology shows no signs of slowing down and its future capabilities are near limitless.

2.1.2 Problems with Smart Homes

With increasingly modernised smart home technology, its uses become more versatile, however, this has resulted in ever-growing challenges that hinder its expansive potential. Consequently, whilst many smart home technologies progress to solving existing problems, this allows for newer and more complex problems to point out existing flaws within their modus operandi.

There are several factors that limit the advancement of the industry and the effectiveness of smart devices as a whole. Most of the problems with modern smart homes can be grouped into three main categories:

- Interoperability

- Lack of true intelligence, and
- Requirement for Internet access

(?, ?).

2.1.2.1 Interoperability

There are many different competing standards in the world of smart homes, including devices that use Zigbee, Z-Wave, Bluetooth and Wi-Fi protocols. This means that any given smart home device may not be compatible with another device, even if they are from the same manufacturer. This can be frustrating for consumers who want to create a customised smart home environment that meets their specific needs and preferences when selecting their products and discourages their entry into the market.

There is currently a new protocol, called Matter, in development by the Connectivity Standards Alliance (CSA), in partnership with some of the leading smart home companies, that aims to unify each of these standards into a single, open-source standard and allow legacy devices to be able to communicate with one another. This is a step in the right direction, however, considering historical context, this may result in it becoming an additional competing standard, with lengthy certification processes driving away smaller manufacturers, and, in turn, competition in the market.

Currently, there are many competing smart home platforms and hubs, such as Amazon Alexa, Google Home, Samsung SmartThings and Apple's HomeKit. Even if the consumer does manage to choose devices all running on the same communication protocols, they may still be unable to control them all from a single app or interface. Apple and Google are both notorious for having lengthy and expensive certification processes for third-party developers to integrate their devices into their platforms, which can be a significant barrier for smaller manufacturers, splintering the market further.

This usually leads to one of two possible outcomes for the consumer. Either consumers end up locked into a single ecosystem, unable to switch to a different platform without replacing all of their devices, which can be a significant financial burden. Or, they end up with a mix of devices from different manufacturers that are unable to communicate with one another each with its own dedicated app, which can be inconvenient and more time-consuming than not having smart devices at all.

2.1.2.2 Lack of True Intelligence

Despite their name, smart homes are not actually all that intelligent in their automation abilities. Even in some of the most advanced smart homes, the devices are limited only to simple routines and schedules, and basic decision trees relying on primitive data from a limited number of sensors in the home.

With the recent advent of Artificial Intelligence (AI) and Machine Learning (ML) technologies, it is theoretically possible to create systems that can learn from user behaviour and adapt to their preferences. However, most smart home devices are not yet capable of this level of predictive control and instead rely on the user to program them to perform specific tasks at specific times.

2.1.2.3 Requirement for Internet Access

Finally, the requirement for constant internet access for smart home devices may be one of the most significant inhibitors of the expansion of the smart home industry. Many smart devices sold today often needlessly process all of their commands remotely, meaning that they require constant internet access to communicate with their respective cloud services in order to be able to control the devices, even from within your own home. There are very few options on the market for devices that allow for local processing of commands. For example, Google, Amazon and Apple's respective voice assistants, Google Assistant, Alexa and Siri, all require an internet connection to process voice commands, so without one, they are unable to control any devices in the home.

So if your internet connection goes down, you may be left with a house full of smart devices that are suddenly very dumb. Not only will you not be able to access the devices remotely to control them or monitor your house, but you may also lose the ability to control them from within the same network as they cannot connect to their company's servers.

2.2 Review of Existing Literature

2.2.1 Computer Vision for Smart Home Automation

In a paper published at the 2019 International Conference on Communication and Signal Processing, Mohammad Hasnain R. et al. set out to "develop a smart [Internet of Things] (IoT) based light control system" using computer vision (CV) and artificial intelligence (?, ?). Their objective was to reduce the wastage of electricity due to the "negligence and forgetfulness" of people using the environment (?, ?).

The authors correctly identify one of the biggest issues with presence detection in common home automation deployments. Most setups use an array of infrared (IR) sensors to detect movement in a room but this poses a few challenges. Firstly, any movement in view of the sensors will trigger an action that is intended only for human presence. So if a book falls off a shelf or an animal runs past and interrupts the IR beams, then the sensor will report back a false positive result. Additionally, another limitation of this method, which was not outlined in this paper, is that if a person remains stationary in the room, for example, whilst sitting on a couch, then there will be a false negative result suggesting that there is no human presence in the room.

To solve these downfalls of existing implementations, the authors took a different approach, utilising AI to identify people through a camera in a living space, enabling them to differentiate between objects and people. However, there are a couple of areas that they did not explore that were within reach using the setup that they had implemented and that this thesis intends to expand upon.

A limitation regarding their investigation was that the authors only used the sensors as a toggle for light emitting diodes (LEDs) with the aim of reducing unnecessary energy use. Furthermore, to identify people in the camera, they used You Only Look Once (YOLO), a computer vision AI model for used object detection, classification, and segmentation, which also has a built-in "skeletonisation" feature, allowing you to track a person's posture and make more intelligent decisions based on a person's movements. These two ideas present an opportunity to fill a potential gap in the market making more intelligent decisions in the home using existing technology and real-world implementations. Consequently, this thesis aims to implement a system capable of executing more complex tasks such as controlling other smart devices in the house like a television or blinds with more potential states than just off and on.

Aside from this paper, there has been extensive research into the use of computer vision in smart home deployments. Nonetheless, very few of these discuss the integration of computer vision with smart home devices, and even fewer attempt to implement this integration. There is extensive existing research that focuses on computer vision in homes, but it is mostly regarding intrusion detection and security (?, ?, ?, ?, ?, ?, ?, ?).

V. Patchava et al. implemented a front-end for a smart home with the ability to toggle devices and stream the video feed from a camera with computer vision capabilities, while C. González García et al. implemented a system that could detect the presence of people and measured the rate of true positives, false positives, true negatives and false negatives (?, ?, ?). Others implemented presence detection with CV and were able to identify intruders and any weapon that they were carrying (?, ?, ?). These papers discuss facial recognition software and the ability to alert homeowners of potential intruders. Moreover, the authors discuss integration with custom smart home dashboards and IoT platforms, but do not discuss the integration of these systems with smart home devices and automation, which is the primary focus of this thesis.

2.2.2 Human Action Recognition for Smart Home Automation

Futher research exists into potential methods of person tracking, human action recognition and behaviour analysis using cameras in the home (?, ?). Here, there is more of a focus on using this technology to support the elderly and the disabled, through ambient assisted living (AAL) to "improv[e] their quality of life and [maintain] their independence" (?, ?).

A. Chaaraoui et al. discuss their deployment with both red, green and blue (RGB) cameras collecting information and recognising "key poses" and hand gestures by creating silhouettes of the person, and also by using red, green, blue and depth (RGB-D) cameras using the depth information to more accurately track movements. They were able to achieve this using a Microsoft Kinect camera and depth sensor "with low-cost and real-time performance", which is the same sensor that was used for the development and deployment of the custom home automation system in this thesis, as will be covered in more detail in Chapter **??**. This existing deployment shows promising signs that the proposed setup should work smoothly in a home environment. While they were able to get these systems working in real-time, recognising primitive human actions such as standing, walking, sitting and falling, there was no mention in the paper of actual integration with home devices.

In 2000, J. Krumm et al. followed very similar processes, using background subtraction to create silhouettes of people and overlayinbvg depth data over the top of these cutouts through stereo images (?, ?). However, due to their early adoption of the technology they were somewhat limited by the computational power of computers during this period and had to run far more complicated and unwieldy setups that would not be suitable for a real home deployment today. This meant that their trackers ran at only 3.5 Hz, even with the computational load shared between three computers, and often had trouble tracking more than three people at a time or people wearing similarly coloured clothing.

The authors were able to integrate their system with some home devices with custom programs such as a wireless mouse that could be carried to any table in the room and the clicks and movements of the mouse would be redirected to the nearest computer display. Another program "automatically start[ed] and stop[ped] a [Videocassette Recorder] (VCR) or [Digital Video Disc] (DVD) movie when a person sits on or stands up from a couch." The film would also be "automatically rerouted to different displays in the room, depending on where the person [sat]" (?, ?).

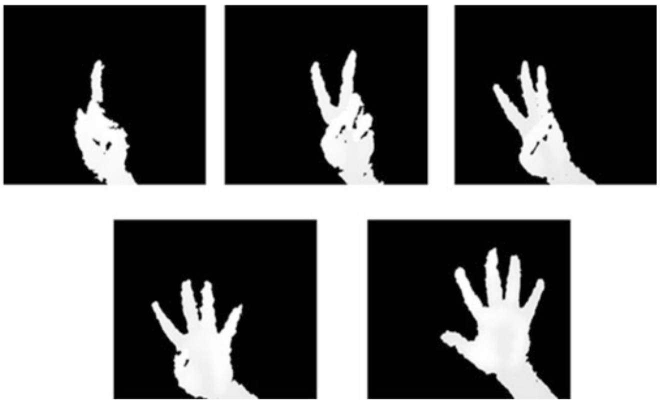
These are similar objectives to what this thesis partially aims to achieve, however, the technology has advanced significantly since then and the proposed setup should be able to run on a single computer with a single camera and depth sensor, making it more accessible to the general public.

2.2.3 Gesture Recognition for Smart Home Automation

In 2017, S. Desai and A. Desai proposed an algorithm for gesture recognition for home automation, also using the Microsoft Kinect (?, ?). Their proposal was specific to only recognising hand gestures, not full-body tracking. This involved segmenting the hand from the image at a specified depth range of 250-650mm, removing the noise and background from the image and converting the RGB into a solid binary black and white image, as shown in Figure **??**. The finger positions were then compared against a set of predefined gestures and then classified. The classification then defines the action that the system should take, sending a signal to an Arduino board to control different home appliances such as a television, charger or fan.

Using this technique, they were able to achieve an 88% accuracy rate in recognising the gestures, a promising result for an early-stage development research project. However, this only worked when users were giving gestures within 65cm from the camera, and only within a 40cm range. This is not practical for day-to-day use as this would either require dozens of cameras per room, spread out every 1.3m, or it would require users to get up from wherever they are in the room and move to the camera in order to control their devices. In this case, controlling their devices manually may be equally as convenient. Due to these limitations, tracking gestures, only via hand movements, does

Figure 2.1: Extracted Hand Gestures from Microsoft Kinect (?, ?)



not appear to be a viable solution for smart home automation, and full body tracking may be required to make this technology more practical for everyday use.

This paper, among others, refers also to computer vision technology recognising gestures through wearable devices (?, ?, ?, ?). Implementing a system to track wearable devices opens up many possibilities for gesture recognition as well as the potential to integrate medical monitoring tools with it. In fact, using the wearable device, T. Starner et al. were able to track the user's gestures against control gestures, which measure continuous input from the user, with an accuracy of 95% and user-defined gestures with an accuracy of 97% (?, ?).

Using a device that interacts directly with the user's body, will significantly increase the accuracy of gesture recognition, as the device is able to measure its own movements, and hence the user, as opposed to a potentially distant camera that attempts to infer the user's movements based on depth data. However, this creates a significant hurdle as the user must be wearing the device at all times in order to control their devices, which significantly impedes the user experience and increases the inconvenience that this technology is supposed to alleviate.

2.2.4 Behaviour Monitoring for Elderly Care

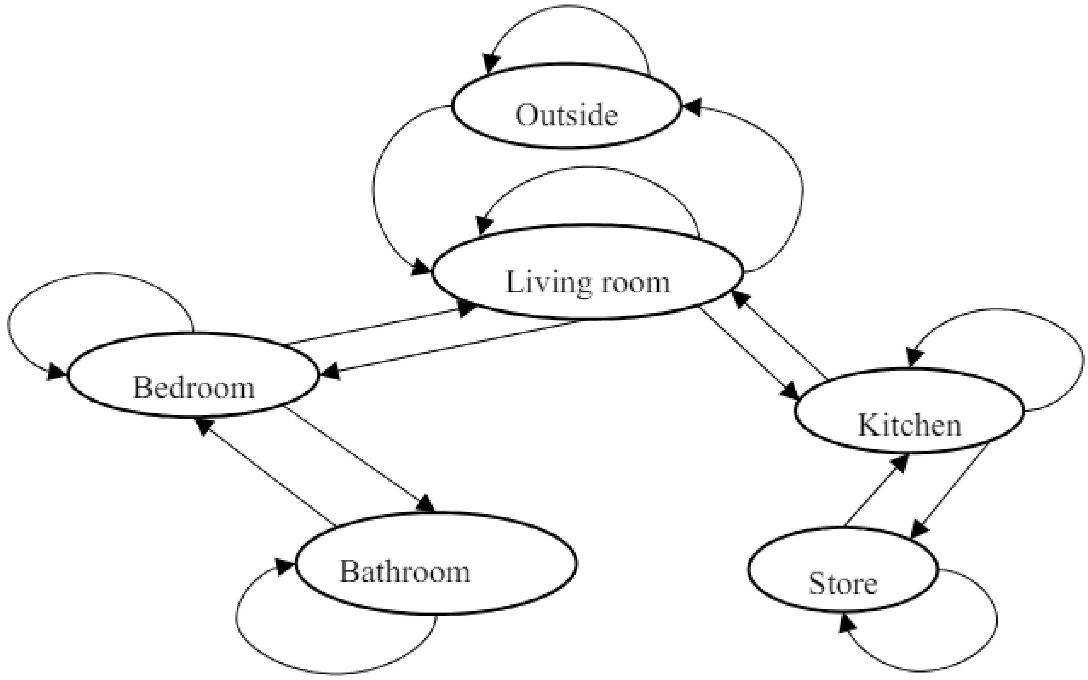
In a 2017 paper, "A Behaviour Monitoring System (BMS) for Ambient Assisted Living," S. Eisa, and A. Moreira constructed an environment that could monitor the behaviour of elderly people in their homes (?, ?). The system used passive infrared (PIR) sensors to monitor the movement of the elderly person between rooms and detect any deviations from their normal routine. The theory behind their practice was that "[a]n elderly person remains in good health as long as he or she can carry on his/her daily activities as usual with no significant deviations from the normal daily routine." This meant that if the system could identify any irregularities, then it could alert a carer or family member to check in on the user to ensure they are okay and potentially catch any health issues early.

This was implemented by treating the user's location in the house (by room) as a state machine, with the user able to transition states by moving between rooms, as shown in Figure **??**. From this, a transition matrix is created for a given time of the day, which represents the probabilities of the user either staying in the same room or moving from one room to another. Training an AI model to recognise the user's regular behaviour, the system could then recognise any deviations from this behaviour. If the behaviour was deemed abnormal, it would then be classified to determine possible causes of the deviation, such as the user oversleeping if they are in their bedroom for too long, Table **??** shows the abnormal behaviours that the authors listed.

Table 2.1: Abnormal Behaviours—Descriptions and related health-declines (?, ?).

Behaviour	**Description**	**Health-Declines**
OverSleeping	An extended stay at bedroom; longer than usual.	Mobility problems, strokes.
LessSleeping	Detected motion at one of the rooms, not a bedroom, during the usual sleeping time.	Having sleepless time due to anxiety, depression or may be developing Alzheimer's diseases.
NotBackHome	Monitored person stayed outside longer than usual.	Having trouble coming back home or get lost or wondering outdoors.
Dead	No movement and long stay at one of the rooms, not bedroom nor outside;	Death.

Figure 2.2: Room-to-room State Transition Model (?, ?).



The model they implemented also self-updated once every week so that it could adapt to the user's "behavioural changes that are not necessarily abnormal behaviours" over time and account for seasonal changes in behaviour (?, ?). This meant that they were able to implement a behaviour monitoring system that learned to adapt over time and consistently monitor the user's behaviour for any irregularities with primitive PIR sensors. If this was achievable with such simple sensors, then the possibilities with more advanced depth sensors and computer vision should allow for more advanced behaviour monitoring and prediction.

There is also some overlap in this field with the computer vision and gesture recognition technologies that have been discussed. M. Oudah et al. implemented a hand gesture recognition system, similar to that of S. Desai and A. Desai, but with the aim of assisting the elderly and disabled in their daily lives by allowing them to use gestures to get support if communication is a problem for them (?, ?). With working deployments of both of these technologies in the AAL field, this shows promising signs for potential in the home automation field.

2.2.5 Voice Control and Multi-Modal Smart Home Automation

Using voice commands for controlling devices in a home is an already well explored area of research, with many existing implementations and products on the market for over a decade. Still, in recent years there have been many developments from individual researchers into creating custom open-source voice control systems for smart phones to control devices in the home. Papers from P. J. Rani et al. and J. R. Aluru et al. both discuss the implementation of mobile apps that take voice commands from the user and send them to a miniature computer, an Arduino board or Raspberry Pi respectively, to control devices in the home (?, ?, ?). Their implementations are both very versatile and could allow for the user to control a wide range of devices in the home, simplifying home device control. However, both of these systems are limited to only voice commands and do not integrate with any other systems, meaning adding devices requires manual programming and setup, making it far less user-friendly and accessible to the general public. They also are reliant on the use of a smart phone application to record the voice commands which is not always practical for the user to have on them at all times. An improved implementation may have static microphones in the home that are always listening for commands, similar to the Amazon Echo or Google Home devices, so that users are not burdened by the extra step of having to retrieve their phone and open an app, at which point they could have already manually controlled the device. Additionally, they both require an active internet connection, which is a significant limitation for smart home devices, as discussed earlier.

In 2020, E. Dutt et al. explored the prospect of multi-modal inputs for controlling a smart home (?, ?). They proposed a system that could control a smart home using both voice and gesture recognition, with the aim of making the system completely hands free, with no physical touch required, making it more accessible to people with physical disabilities. Their system employed IR sensors to detect a gesture or obstruction of the sensor which triggered the master switch to allow voice commands to be processed. Once activated, the user can then use 15 different voice commands (as specified by the users in advance) to send signals to devices to control them. There are several issues with this approach, including the limited functionality available and the primitive gesture detection methods. Only allowing 15 voice commands is a significant limitation as this restricts the the user to only a handful of actions that they can perform with their devices. Furthermore, the gesture detection system is incredibly primitive, only detecting an IR sensor obstruction, which essentially does not detect unique gestures at all but acts as a replacement for a wake word. Despite this, the idea to combine multiple inputs to control a smart home is promising and could be expanded upon to create a more intelligent system, which is one of the goals this thesis aims to achieve.

N. Ganji et al., however, do integrate more complex gesture recognition into their system, using an RGB camera to detect hand gestures, with four pre-defined gestures that determine which device to control (?, ?). This allowed users to use more complex gestures to control their devices, expanding the possibilities for more robust control of devices in the home. This, in conjunction with voice commands, allows for a more natural interaction with the smart home. While this is a significant improvement over the previous implementations, it still has limitations in the number of gestures that can be recognised and the lack of integration with other systems. The gestures and voice commands are pre-defined so the user cannot customise them to their needs without significant programming knowledge. The system also requires an internet connection at all times to process the voice and gesture commands, prohibiting the system from functioning when the internet connection is lost.

2.3 Gaps in Literature

After a thorough review of existing literature in the space that is being explored by this thesis, it is clear that there has been extensive research into many of the facets of the proposed system in consideration. Computer vision is a widely researched field and has been applied to many different areas of smart home technology, from security to gesture recognition. Human action recognition and gesture recognition have both also been studied for decades, and more recently there has been a significant amount of research into voice controls and multi-modal input recognition.

Despite this, there is a significant gap in the literature when it comes to the actual integration of these technologies with smart home devices and automation. Most of the literature either discusses the technology in isolation or discusses implementation into smart homes on a theoretical level. Other papers discuss solving some of the singular problems with smart homes individually, but none cover the full scope of the problems that are faced by the industry. For example, there is vast research that discusses the use of computer vision for intrusion detection and security in a smart home, but the possibility of using this technology for automating tasks and controlling devices in the home is rarely considered. Almost none of the papers have actually implemented the technology into a real-world smart home environment and tested the results. This presents a significant opportunity for this thesis to contribute to the field by providing a real-world implementation of these technologies and evaluating their performance and user experience. Each of the critiques made in the literature review will be goals to address in this thesis, with the aim of creating a more intelligent and autonomous smart home environment that is more accessible to the general public.

2.4 Problem Statement

Smart homes today promise enhanced convenience, safety, and security through the connectivity of more and more devices and the automation of regular tasks. However, several critical challenges prove to be a hindrance to their effectiveness as well as widespread adoption and, in turn, the growth of the industry.

Interoperability issues between devices and existing platforms, the lack of genuine intelligence in these automated systems, and the constant reliance on internet access pose significant obstacles to realising the true potential of smart home technology.

In an attempt to combat these challenges, this thesis aims to develop an entirely custom and customisable, intelligent environment where a user's movements will be able to accurately predict how to control devices within the home, through gesture recognition and behaviour prediction. Utilising local processing and sensor data, the system will enhance convenience, safety, and security without relying on internet connectivity or support from third-party companies to allow interoperability.

Moreover, with several gaps in the existing research and literature in this area, this thesis aims to be able to integrate computer vision techniques and depth sensor technology into a smart home environment, utitlising this data to assist in automation, not just home monitoring and security. This will provide a real-world implementation of these technologies and a live demonstration of the home's capabilities, as opposed to the theoretical implementations that are currently discussed in the recent literature. Of those papers with existing implementations, there will be significant expansion on the capabilities of the system, with the aim of controlling more devices and giving the user the ability to easily customise the system to their needs.

Overall, this research endeavours to contribute to the evolution of smart homes by introducing innovative approaches to enhance their intelligence and autonomy, ultimately improving the quality of life for users.

2.5 Aims and Outcomes

The research objectives of this thesis include designing and implementing artificial intelligence and computer vision for movement prediction and gesture control, integrating these with concurrent voice controls and into existing smart home infrastructure, finally evaluating the performance and user experience of the environment.

With entirely locally processed data, the system will be able to control devices within the home without the need for internet connectivity, enhancing convenience, safety, and security for users. Computer vision techniques will be used to identify users in the home and track their movements and gestures to identify devices, with voice commands being used to specify an action for the device to take.

The anticipated outcomes of this research include advancements in smart home technology, improved user experiences, and insights into addressing the broader challenges in the field as outlined in the literature review. While the research focuses on addressing specific aspects of smart home functionality, certain limitations, such as device compatibility and privacy concerns, will be acknowledged and considered throughout the study.

Chapter 3

Methodology

3.1 Proposed System

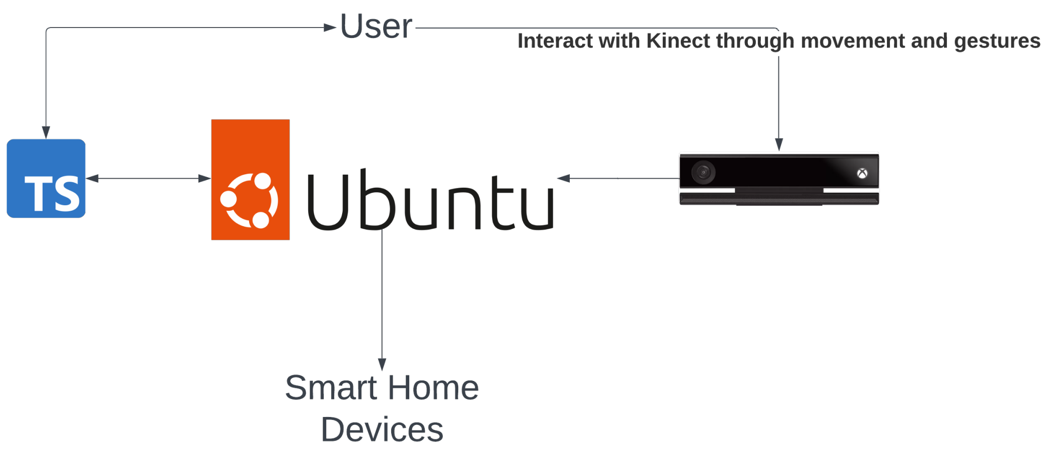
To solve the issues outlined in Chapter **??**, in Thesis A, the following system was proposed. Using a Microsoft Kinect v2 depth sensor, user's gestures can be detected and processed on an Ubuntu machine. Utilising Robot Operating System (ROS) 2, once the gestures are processed, signals can be sent over the ROS network to control devices throughout the home. Finally, a TypeScript (TS) frontend would be developed to enable users to customise their home to make the system adaptable to user needs. This led to the initial simplified technology stack as shown in Figure **??**.

3.1.1 Technology Stack

3.1.1.1 Thesis A

Throughout the development of this system, the technology stack gradually changed over time in order to accommodate growing goals and to overcome challenges. In Thesis A, there were many challenges in getting the Kinect camera to communicate with ROS 2. A very specific environment setup is required in order for these systems to function

Figure 3.1: Simplified Technology Stack



together reliably as shown in Figure **??**.

3.1.1.2 Thesis B

Following this, during the complete development of the system in Thesis B, the technology stack was modified slightly in order to accomodate a more robust and customisable system for managing smart devices in the home as in Figure **??**. Using the open source home automation platform Home Assistant allows for far more versatility in automation abilities, choices for devices and provides it's own user interface, eliminating the need for a separate custom TS frontend.

3.1.1.3 Thesis C

During Thesis C, one of the stretch goals that was set, implementing voice commands and multi-modal device control, was developed. This, again, made some minor changes to the technology stack, incorporating a new TS frontend for users to define their own custom voice commands, and a Vosk speech recognition model integrated as a node on the ROS network. The final technology stack for this thesis as it is currently implemented is as shown in Figure **??**

Figure 3.2: Thesis A Proposed Technology Stack

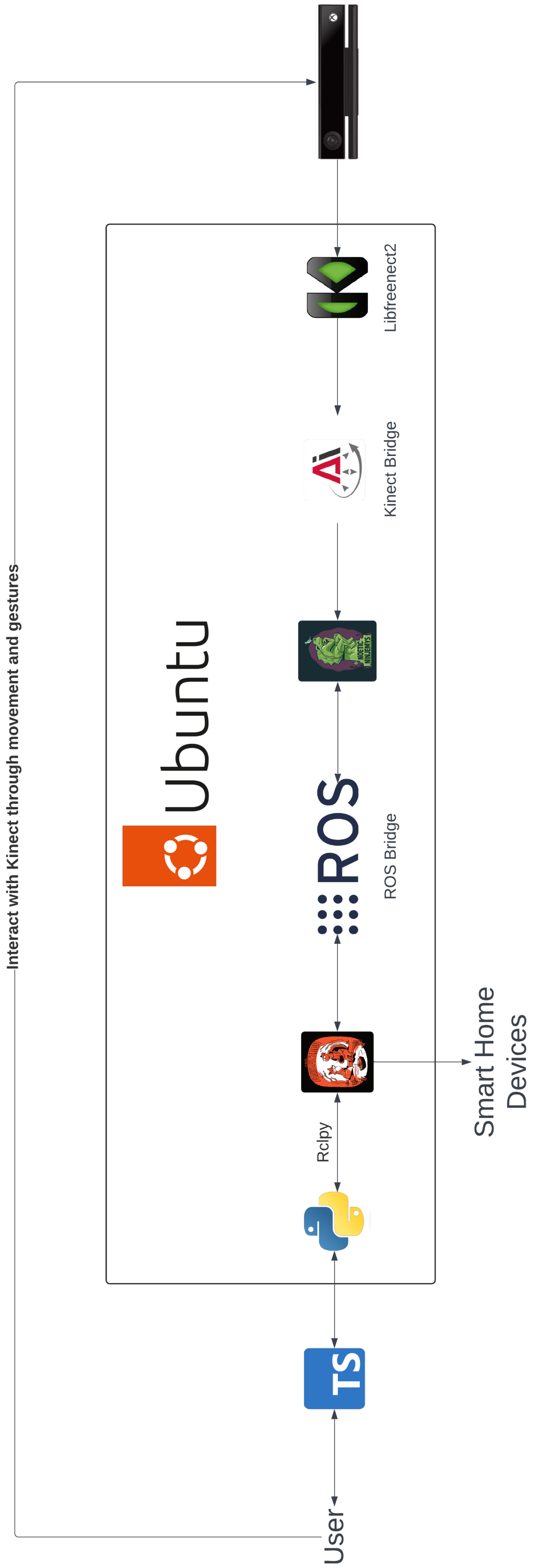


Figure 3.3: Thesis B Implemented Technology Stack

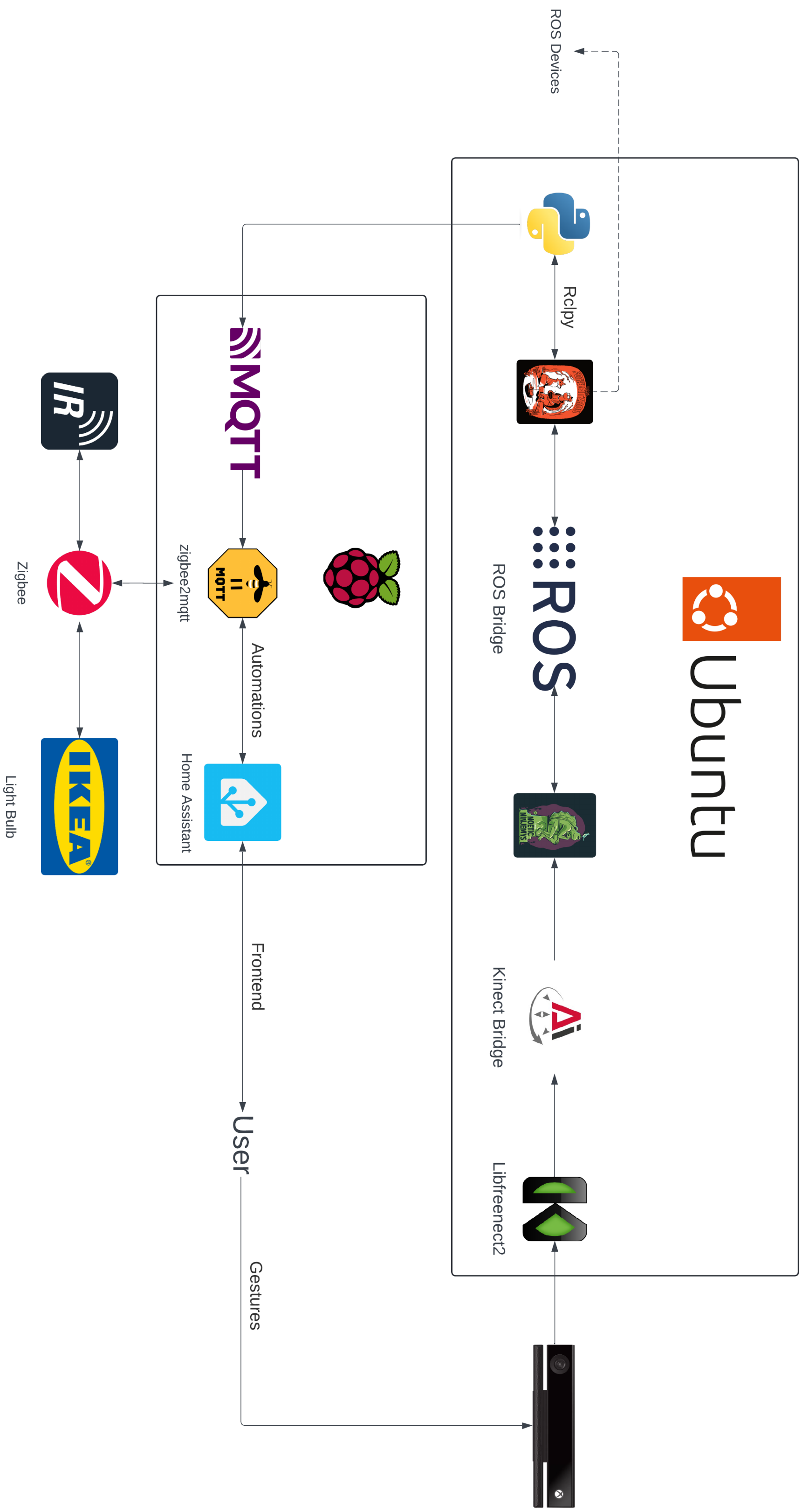
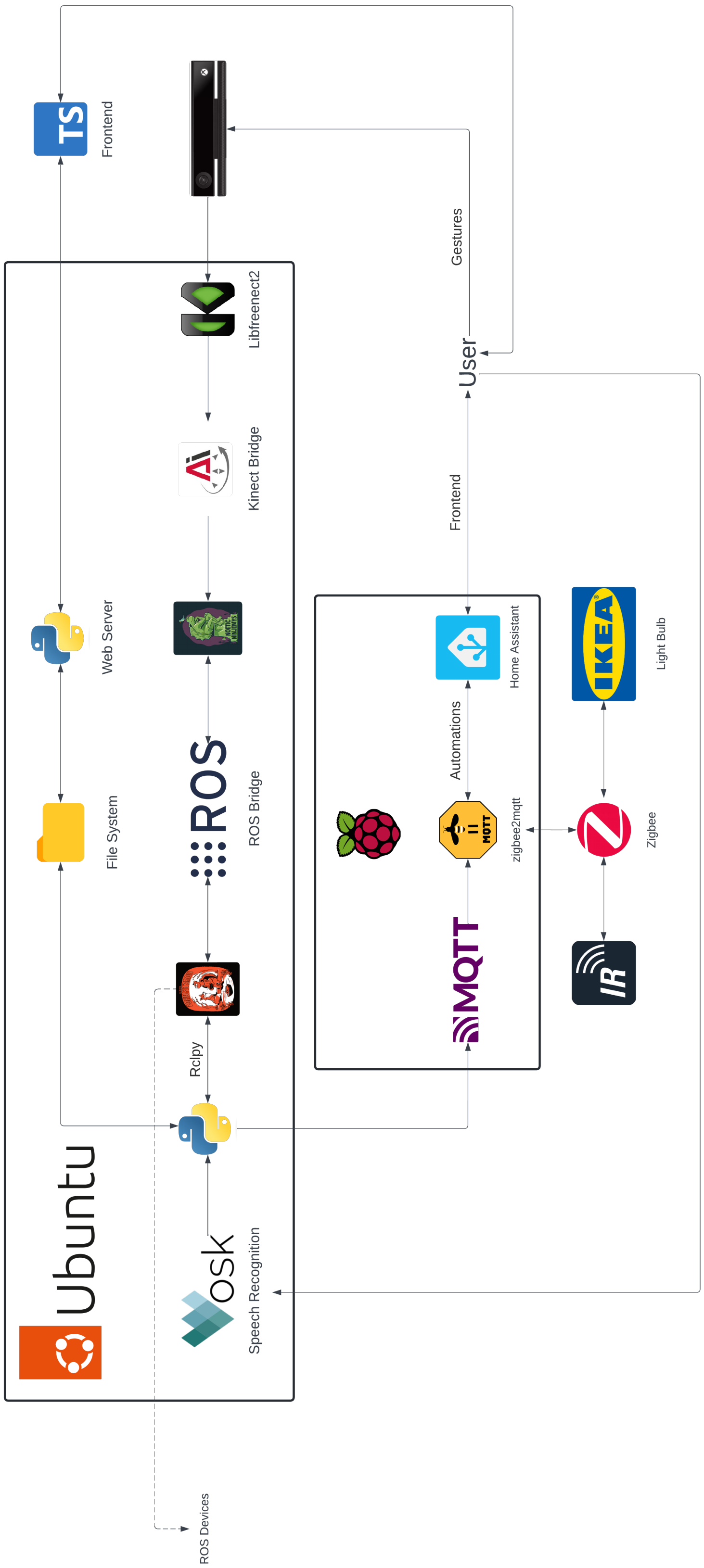


Figure 3.4: Thesis C Final Technology Stack



3.2 Hardware Components

The hardware components used in the development of this system include the following:

- Dell XPS15 9510 Laptop
- Microsoft Kinect v2
- Raspberry Pi 3
- Sonoff Zigbee 3.0 USB Dongle Plus
- Ikea Trådfri Zigbee Bulbs
- Zigbee IR Emitter

The selection of the Dell XSP15 Laptop was due to its immediate availability as it was already owned. This is a powerful laptop with a discrete graphics card and was always sure to be capable of running the software required. However, a laptop of this calibre is not completely necessary and most modern hardware should be sufficient.

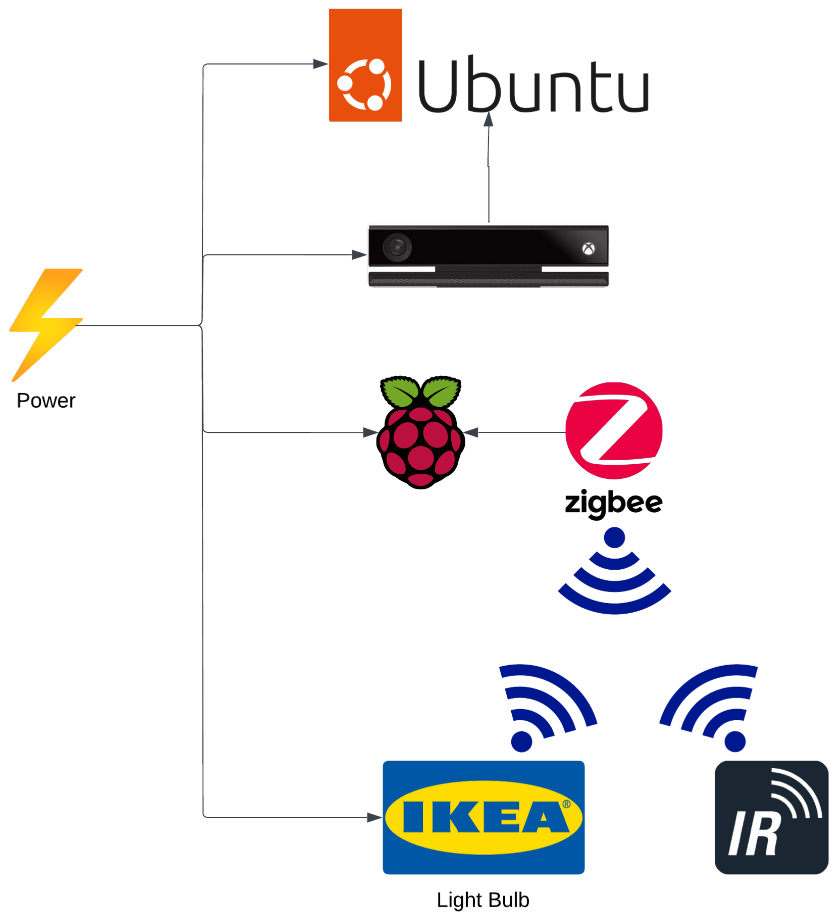
The Microsoft Kinect v2 was also selected due to it being readily available. This, however, is a necessary piece of equipment for this system as it is currently designed. In theory, other depth cameras could be suitable for alternative deployments but this would require significant work to modify the existing environment setup and potentially large portions of the codebase.

The Raspberry Pi 3, was selected for multiple reasons. Home Assistant is relatively lightweight so the choice of hardware was not limited by processing power, therefore a miniature computer such as a Raspberry Pi was appealling due to it's low cost, small footprint and ease of use with Home Assistant. The Pi 3 specifically was chosen as it was available on-hand, though other single-board computers capable of running Home Assistant would also be fitting choices.

The Sonoff Zigbee dongle, which is attached to the Raspberry Pi, along with the Ikea light bulbs and IR emitter were primarily chosen for their low price points. Zigbee is an open source communication standards with lower fees for accreditation than other competing standards such as Z-Wave, therefore their products tend to be cheaper than their Z-Wave counterparts. The dongle is required in order to communicate between Home Assistant and the smart devices. The light bulbs and IR emitter were purely chosen as two examples to demonstrate the functionality of the system. Any competing standard devices or different device types entirely would also be valid choices for the system depending on user needs.

The components should be set up as shown in Figure **??**.

Figure 3.5: Physical Setup Diagram



3.3 Software Components

The key software components used in the development of this system include the following:

- ROS
- ROS Client Library for Python (rclpy)
- Home Assistant
- Messaging Queuing Telemetry Transport (MQTT)
- Vosk Speech Recognition
- YOLO
- Python Web server
- TypeScript frontend

The proposal to use ROS to control devices in the smart home was made very early in the research stages. This was because many of the robots and devices existing in the lab already operated on ROS. Building the smart home on the same system would allow for further research and development into getting these projects working cohesively with one-another, as there are some operations that can be performed by robots in the lab, such as pointing to and retrieving an item, that could be useful in a smart home. In it's current state, the commands to the smart home are accessible by other devices on the ROS network, however there is no integration with the robotics work being done, but this does allow for future expansion of the project.

Rclpy is the ROS library for Python and was one of only two options for development. The alternative was a C++ library, however, due to the nature of the project, utilising AI and the lack of C++ skills upon embarkment of the project, rclpy was chosen to write the code for custom ROS nodes. This is where the gesture recognition software operates as well as the speech to text model and voice command recognition nodes.

Home Assistant was chosen because, as an open source home automation platform, it allows users to customise their environment to their liking to a much higher degree than other mainstream platforms such as Apple HomeKit or Google Home. This allowed a simple integration with the ROS components of our system as there was much more choice for methods to communicate between the platforms, nothing was locked down.

MQTT is a lightweight publisher/subscriber machine to machine messaging service. This, together with zigbee2mqtt, allows the ROS nodes to send an MQTT message over the network and control Zigbee devices in Home Assistant directly. This was selected due to its lightweight messaging service, cheaper products and simplistic implementation. It is worth noting that using MQTT does not limit users to using Zigbee devices as the MQTT messages can be used to trigger automations in Home Assistant. If a user determines that bluetooth, Wi-Fi or Z-Wave devices are more suitable for their requirements, then they are still able with no extra work required.

Vosk's speech-to-text (STT) models were selected for two main reasons. Firstly, these are the same models used by other robots in the Human Robot Interaction lab where this system was deployed. Maintaining a smaller number of tools makes the environment far easier to manage as functionality expands. Additionally, Vosk supports streaming audio directly from a microphone to the model, whereas many other local STT models only support processing entire audio files. Other models such as OpenAI's local Whisper models were tested in development, however the inability to stream audio made reliably interpreting the audio challenging. YOLO was also selected as this was already in use by other systems in the lab and it provides a skeletonisation feature which allows the extraction of the coordinates of key points on a users body to track their poses and gestures.

Python and TypeScript were selected for their simplicity in development with TS chosen over JavaScript (JS) for it's strict typing and as an extension challenge for the developers, previously unexperienced with TS. Other languages for the backend web server and the frontend web app are also suitable choices.

3.4 Component Interactions

The process of enabling communication between ROS 2 and the Kinect camera took many iterations of testing throughout Thesis A, and resulted in a somewhat convoluted setup as shown in Figure ??. However, this was necessary for the goals of this thesis. The Institute of Artificial Intelligence at the University of Bremen (IAI) has developed a set of drivers to allow communication between the Kinect and ROS 1, and another user created a fork of this repository to support OpenCV 4.0, an open source CV library, which was required for this deployment. Since there is no existing Kinect bridge for ROS 2, the data from the Kinect must be processed first by ROS 1 and passed through to ROS 2. Because of this, the operating system required is Ubuntu 20.04, Focal Fossa, as this Linux distribution (distro) supports multiple versions of both ROS 1 and ROS 2, allowing communication between one another. OpenKinect's Libfreenect2 drivers are required in order for IAI's Kinect 2 Bridge to access the data from the sensors on Ubuntu. Finally, Open Robotics, the developers of ROS, provide a bridge between ROS 1 and ROS 2 that allows the translation of the topics, services, and actions between nodes in either direction. These tools, in conjuction with ROS 1 distro Noetic Ninjemys, and ROS 2 distro Foxy Fitzoy, ROS 2 is able to access the data from the Kinect.

Figure 3.6: Kinect to ROS 2 Layers

To integrate this with the current system, rclpy is used to process and register the gestures from the users and signals are then sent to the Home Assistant server on the network via MQTT. A Raspberry Pi 3, is used to run the Home Assistant server along with an MQTT broker, and zigbee2mqtt, a service to bridge mqtt signals directly to zigbee smart devices. From there, Home Assistant is able to control the smart devices over the zigbee network through manual control from the user or automations.

Using rclpy, there are two ROS nodes created to process the speech and the user's voice commands. The STT node takes audio directly from the computer's built-in microphone and streams it to Vosk to process the speech. The result of the vosk model is then published to a topic on the ROS network. The second node subscribes to both the STT topic published by the first node, and the topics created by the Kinect sensors through the layers of translation from the camera to ROS 2. This then processes the voice commands at the same time as the gestures and will send the MQTT message to trigger devices.

Finally, there is a web app that allows users to customise which voice commands are available to control their devices. This is available on the network through a local IP address, or within Home Assistant. The web app accesses a JavaScript Object Notation (JSON) file on the system that contains all device types available in the home and their associated commands. From here the user can add, modify or remove commands and device types entirely depending on the needs of their home. This modifies the JSON file on the system, which in turn triggers an update in the ROS node to ensure that only valid commands are being checked for.

3.5 Code Logic

Once the Kinect data and STT result has reached the final ROS node, there are primarily two processes running concurrently in order to trigger device commands.

First, with each frame from the Kinect's camera, the image callback is triggered which determines a users gesture and then stores it for use later by the voice commands process. To determine a user's gesture, eight points on the users body are extracted using YOLOv8. The hip, shoulder, elbow and wrist of both the left and right sides. Then, using the depth data from the Kinect, the pixel coordinates from the image can be converted to 3-Dimensional world coordinates. In case of any errors in YOLO selecting a point just outside the users silhouette, the depth value is taken as the point closest to the camera within a five pixel radius of the point. Using the wrist and elbow coordinates, a vector can be created to determine which direction a user is pointing. This is done by extending this vector and intersecting it with the bounding box that is the dimensions of the room. Then it can be determined which wall, floor or ceiling is being pointed at. A gesture is only registered if the distance from the users wrist to their hip or their shoulder surpasses a pre-defined threshold. The gesture is then stored in a list as a history with an associated timestamp with a gesture being stored at 250 milliseconds intervals and storing the last 10 seconds of gestures.

At the same time, whenever the STT node sends a new string the speech processing callback is triggered. The output from Vosk provides multiple possibile interpretations of the speech in order to minimise misinterpretations. Each of these alternatives is checked to see if the sentence begins with the chosen wake word, set to "home" by default. The wake word is not required to trigger a command but is required for the user to receive feedback for an invalid command. This was done so that if a user forgets to use the wake word then if a valid command is given then it will still trigger a result. However, if regular speech is picked up not intended to control a device, then the invalid command feedback is not triggered.

Following this, each of the alternative interpretations from Vosk are checked against the JSON object containing all the valid device types and commands. These are checked against a regex string in order to allow users to speak in a more natural manner and still trigger the correct device with other superfluous words in the command.

The regex string for matching a command for a single device is

```
rf".*\b{command}\b.*\bthat\b.*\b{device}\b.*"
```

For the case where the command is "turn on" and the device is a "light," this gives users the ability to say the simple command "turn on that light", or speak more naturally and say "could you turn on that light over there for me please."

The regex string for matching a command for a all devices of a single type is

```
rf".*\b{command}\b.*\b(all|every)\b.*\b{device}s?\b.*"
```

This gives users the option to say the commands "turn on all lights," or "turn on every light," or more naturally "can you turn on all of the lights please."

Once a command is matched, it is added to a list, and once each of the device types and commands is checked against the command with the latest match is selected. This is done so that a user can correct themselves mid-sentence and still trigger their intended command. For example, a user might say "can you turn on that light, I mean that fan". This would trigger the automation to turn on the fan. Once a command is recognised, the timestamp at which the word "that" is said is compared to the history of gestures and the gesture with the closest timestamp to the keyword is selected.

If no match is found for a given alternative interpretation from Vosk, the process loops. If no match is found at all, and a wake word was supplied, then an audio cue is triggered to let the user know that their command was not recognised and to try again. Additionally, if the user gives a valid command but does not gesture to point to their device, another audio message is relayed to the user to ask them to gesture when giving commands. Finally, if a match is found, the command message is sent over MQTT and captured on Home Assistant to trigger custom automations set up by the user.

Chapter 4

Conclusion and Future Work

This report has covered the evolution of smart homes since the 20th century, and how it has reached the current state of the art in smart home technology, as well as existing literature exploring the potential implementations of computer vision technology and human action recognition in future smart homes. Behaviour monitoring in ambient assisted living environments has also been considered for use as an advanced technology with potential applications in home automation. The gaps in the literature show that there is a vast amount of research into each of these fields but very little in actual application and deployment in controlling IoT devices in the home.

A detailed plan for Thesis B predicts an eight-week timeline for the completion of the project, with a focus on the implementation of the CV gesture recognition system and front-end for manual device control.

The progress during Thesis A so far, from the primitive integration of the Kinect depth sensor with ROS 2, has shown that the proposed system is viable and can be implemented over the course of the next term.

4.1 Future Work

The future work will be continued as outlined in Chapter **??**. At the conclusion of this thesis, further research can be undertaken to explore multi-modal inputs for gesture and voice recognition in conjunction to allow for more natural interaction with the smart home.

References

Aluru, J. R., Kadapa, S. K., & Kumar, G. A. E. S. (2021). Voice control iot home automation using voice assistant raspberry pi. *Annals of the Romanian Society for Cell Biology*, *25*, 5819-5825. Retrieved from `https://wwwproxy1.library.unsw.edu.au/login?url=https://www.proquest.com/scholarly-journals/voice-control-iot-home-automation-using-assistant/docview/2564182405/se-2`

Chaaraoui, A., Padilla-López, J., Ferrandez, J., García-Chamizo, J., Nieto-Hidalgo, M., Romacho-Agud, V., & Flórez-Revuelta, F. (2013). *A vision system for intelligent monitoring of activities of daily living at home* (Vol. 8277). doi: 10.1007/978-3-319-03092-0_14

Cucchiara, R., Grana, C., Prati, A., & Vezzani, R. (2005, 4). Computer vision system for in-house video surveillance. *IEE Proceedings: Vision, Image and Signal Processing*, *152*, 242-249. doi: 10.1049/IP-VIS:20041215

Desai, S., & Desai, A. (2017). Human computer interaction through hand gestures for home automation using microsoft kinect. In N. Modi, P. Verma, & B. Trivedi (Eds.), (p. 19-29). Springer Singapore.

Dutt, E., Maduri, P. K., Gupta, A., Rathour, S., & Kushagra. (2020). Touch-less home automation system with voice and gesture control. In (p. 781-785). doi: 10.1109/ICACCCN51052.2020.9362829

Eisa, S., & Moreira, A. (2017). A behaviour monitoring system (bms) for ambient assisted living. *Sensors*, *17*. Retrieved from `https://www.mdpi.com/1424-8220/17/9/1946` doi: 10.3390/s17091946

From sci-fi dreams to everyday reality. tracing the evolution of smart home through history. (2023, 11). Retrieved from `https://www.linkedin.com/pulse/from-sci-fi-dreams-everyday-reality-tracing-evolution-smart-mypmf/`

Ganji, N., Gandreti, S., & Krishnaiah, T. R. (2022). Home automation using voice and gesture control. In (p. 394-400). doi: 10.1109/ICCES54183.2022.9835832

García, C. G., Meana-Llorián, D., G-Bustelo, B. C. P., Lovelle, J. M. C., & Garcia-Fernandez, N. (2017, 11). Midgar: Detection of people through computer vision in the internet of things scenarios to improve the security in smart cities, smart towns, and smart homes. *Future Generation Computer Systems*, *76*, 301-313. doi: 10.1016/J.FUTURE.2016.12.033

Hasnain, M. R., S, R., P, M., & G, S. (2019). Smart home automation using computer vision and segmented image processing. In (p. 429-433). doi: 10.1109/ICCSP.2019.8697997

Hertzmann, P. (2016). The refrigerator revolution.

Invention of robotic vacuum cleaners. (n.d.). Retrieved from http://www.vacuumcleanerhistory.com/vacuum-cleaner-development/history-of-robotic-vacuum-cleaner/

Krumm, J., Harris, S., Meyers, B., Brumitt, B., Hale, M., & Shafer, S. (2000). Multi-camera multi-person tracking for easyliving. In (p. 3-10). doi: 10.1109/VS.2000.856852

Mahajan, M. P., Nikam, R. R., Patil, V. P., & Dond, R. D. (2017, 3). Smart refrigerator using iot. *International Journal of Latest Engineering Research and Applications*.

Nandhini, M., Rabik, M. M., Kumar, K. S., & Brahma, A. (2020). Iot based smart home security system with face recognition and weapon detection using computer vision. *Regular*.

Oudah, M., Al-Naji, A., & Chahl, J. (2021, 6). Computer vision for elderly care based on deep learning cnn and svm. *IOP Conference Series: Materials Science and Engineering*, *1105*, 012070. doi: 10.1088/1757-899X/1105/1/012070

Patchava, V., Kandala, H. B., & Babu, P. R. (2015). A smart home automation technique with raspberry pi using iot. In (p. 1-4). doi: 10.1109/SMARTSENS.2015.7873584

Rani, P. J., Bakthakumar, J., Kumaar, B. P., Kumaar, U. P., & Kumar, S. (2017). Voice controlled home automation system using natural language processing (nlp) and internet of things (iot). In (p. 368-373). doi: 10.1109/ICONSTEM.2017.8261311

Sefat, M. S., Khan, A. A. M., & Shahjahan, M. (2014). Implementation of vision based intelligent home automation and security system. In (p. 1-6). doi: 10.1109/ICIEV.2014.6850818

Spicer, D. (2016, 5). *The echo iv home computer: 50 years later.* Retrieved from https://computerhistory.org/blog/the-echo-iv-home-computer-50-years-later/?key=the-echo-iv-home-computer-50-years-later

Starner, T., Auxier, J., Ashbrook, D., & Gandy, M. (2000). The gesture pendant: a self-illuminating, wearable, infrared computer vision system for home automation control and medical monitoring. In (p. 87-94). doi: 10.1109/ISWC.2000.888469

Stein, J. A. (2011, 7). Domesticity, gender and the 1977 apple ii personal computer. *Design and Culture*, *3*, 193-216. doi: 10.2752/175470811X13002771867842

Us inflation calculator. (n.d.). Retrieved from https://www.usinflationcalculator.com/

Volety, R., & Geethanjali, P. (2022). Smart home automation using wearable technology. In G. D. Gargiulo & G. R. Naik (Eds.), (p. 259-279). Springer Singapore. Retrieved from https://doi.org/10.1007/978-981-16-5324-7_11 doi: 10.1007/978-981-16-5324-7_11

Wilson, C., Hargreaves, T., & Hauxwell-Baldwin, R. (2015, 2). Smart homes and their users: a systematic analysis and key challenges. *Personal and Ubiquitous Computing*, *19*, 463-476. doi: 10.1007/s00779-014-0813-0

X10. (n.d.). Retrieved from https://www.x10.com/

Zhang, X., Yi, W. J., & Saniie, J. (2019, 5). Home surveillance system using computer vision and convolutional neural network. *IEEE International Conference on Electro Information Technology*, *2019-May*, 266-270. doi: 10.1109/EIT.2019.8833773