Context-Aware Proactive Robotic Companions for Care Homes and Communities

Penelope Roberts

A thesis submitted for the degree of Computer Science

Department of Computer Science and Electrical Engineering
University of Essex

October 2023

Acknowledgements

Firstly, I would like to thank my supervisor Dr Vito De Feo for his unwavering support and kindness throughout our time working together. I am immensely grateful for the patience and faith you have in all your students and your ability to see the best in them even when they cannot.

Additionally, I would also like to thank Dr. Dimitri Ognibene and Dr. Vishuu Mohan for their help, guidance, and opportunities they have given me over the course of this research.

This thesis would also not have been possible without the support of the University of Essex, who kindly provided the studentship to complete this work, as well as a brilliant campus to complete it on.

I would like to say a huge thank you to everyone in the Lavoro Lab for all the laughs, snacks and general kindness that is always present there.

Finally, this thesis would not have been possible without the overwhelming support from my wonderful family and friends. From my parents who experienced both the brilliant and terrible days without complaint or ever losing faith, to my friends who were always there for a coffee and a laugh.

Abstract

The question of whether robots have a place in healthcare is one that is becoming more and more necessary to consider. Shortages of healthcare staff, lack of funding for healthcare services and the continuously growing population have begun to highlight the increasing need for changes in the way we help vulnerable members of our society. In this thesis, I have explored the potential of robots, specifically the humanoid robot Pepper, to help fill this gap when it comes to the care of the elderly who may have a diverse range of needs by providing users with aid and companionship in their daily lives whether that is retaining independence for longer at home, or as part of a larger care community. I have verified that through the development and implementation of a biologically inspired cognitive architecture onto the Pepper robot, it is possible to create a robotic companion that can be of assistance to patients and carers alike, by completing requests and engaging in social interaction with users and in turn, reducing the workload of carers. The following chapters document this through a series of practical and simulated experiments that have formed multiple software modules including a cumulatively growing artificial episodic memory system, a human-aware navigation system and a social intelligence module. Through this, Pepper has been able to successfully learn and recall experiences while accounting for current social and environmental contexts and interact with users through a multimodal interface to provide a sense of independence and personalised companionship. The architecture also includes a swift-learning human-aware navigation system capable of navigating and mapping unstructured environments. This method has enabled Pepper to quickly map/re-map new and previously visited environments with average training of single rooms taking up to 120 seconds on average and larger rooms (> 10,000 data points) taking up to 350 seconds.

Table of Contents

Table of Figures

1 Introduction

The focus of this thesis is to develop a biologically inspired context-aware robotic companion for home care and communities that would act as both a healthcare assistant and a companion to users. The project is developed with a specific focus on elderly patients requiring long-term care in their own homes or within care homes, while also allowing for potential expansion to a wider range of users in the future. The companion robot was also designed under the assumption that it would be used mainly within an indoor healthcare environment, such as care homes, hospitals or within a user's own home.

The goal of developing this project was to assist users with daily tasks, provide assistance when required, give feedback to users in response to queries and requests, as well as act as a general social companion for those who would wish it. At the same time, it also aims to assist healthcare providers (nurses, care workers, family members etc.) who are currently responsible for providing help to patients with these tasks, in order to reduce the pressure and workload that is currently placed on them. Because of this, it has been important to create a companion that would be efficient in both a patient's own home or in a shared care facility and that would be able to adapt to either environment with minimal input from humans in order to do so.

To accomplish this, a software architecture has been designed to provide the necessary functionality for a Pepper humanoid robot to perform these kinds of actions and continuously gain new knowledge from its experiences with users. This includes development of software to provide a human-aware navigation system for moving in unstructured and unpredictable environments, a biologically inspired cumulative memory for learning and recall of experiences, and a social module to allow for user recognition, conversation and companionship, as well as a monitoring module to keep track of environment changes and provide feedback to the robot. This companion is designed to interact with a diverse range of patients who require various levels of care, in real-time, within multiple unstructured environments.

The work in this thesis has been based on several key research objectives, namely:

1. To implement a biologically inspired memory system to enable a robot to learn and recall experiences.
2. To develop a quick learning navigation system capable of navigating dynamic unstructured environments.
3. To create a social intelligence software module to enable a Pepper robot to provide companionship and aid to users.
4. To investigate potential links between biological navigation and episodic memory functions in the human brain, and the possible benefits to robotics when applied to artificial systems.

1.1 Motivation

In recent years, an increasing life expectancy and an ageing population has seen the United Kingdom's healthcare system experience an increasing demand for services, while also experiencing a significant lack of available funds and caregivers [1], [2]. The most recent available government study shows that over the course of one year, the UK spent £257.6 billion on providing healthcare services, with over £48 billion spent specifically on providing healthcare to patients with long-term conditions [3]. This was a 2.9% increase on the total spending on previous years, highlighting the continuous growing demand for services over time. One of the main reasons for the increase in demand can be directly linked to the country's growing population. Predictions calculated in mid-2021 placed the current population at 67 million people, the largest reported to date [1]. With this larger population also expected to have a greater life expectancy, it is predicted to continue increasing in size with 72.9 million people predicted to be part of the UK by 2041 [4]. The increased life expectancy and continuous advances in medicine are now reflected in the UK's older population with 600,00 people listed as aged 90 or over, which continues to increase despite statistics indicating lower birth rates [5], [6]. Consequently, the number of people relying on health services for long-term care later in life as well as unofficial care from relatives or friends has also been shown to be increasing, with 40% of people over 65 living with long-term illness or disability [6]. The increase in demand, as well as issues with funding, has created issues with the distribution of care, specifically in social care, where funding currently provided for age-related conditions is not enough to meet the demand from those in need [7].

This thesis proposes that the introduction of an intelligent robotic companion to healthcare could therefore provide substantial benefits to people needing care, specifically the older population, who may not have access to it otherwise. There are also multiple advantages to care facilities as well as informal carers such as family and friends who are relied on to supply care. Financially, it can help to reduce the overall cost of long-term healthcare by taking on the some of the roles currently performed by carers. Tasks often classed as low-skill work, such as medication reminders, fetching and finding objects and answering basic queries can all be automated to reduce the need for carers for these tasks and reducing the overall demand on health providers, allowing them to focus on more specific, individual care for patients. Additionally, by providing services for users in their own homes, the introduction of a companion would help people retain a greater sense of independence for longer while living alone, while also ensuring there is sufficient monitoring in the case of emergencies reducing the number of people requiring places in official care facilities. Consequently, this would reduce the cost on patients for additional human carers as well as significant costs of entering into a care home where the average elderly resident can expect to pay upwards of £730 a week for a basic care home within the UK, plus additional costs if a nursing home is required [8], [9]. By allowing people to remain independent, they not only avoid the cost of care facilities but also guarantee they are still being monitored and help can be called quickly.

There is also the financial impact that care has upon family members. The current number of informal carers in the UK is officially estimated to be around 5 million, though some reports suggest it could be as high as 10 million [10]. Of these informal carers, the most common age group for carers is 45-64 with 41% of carers being in the age group with 29% of this group being listed as caring for a parent [11]. These unofficial carers are often forced to bear the cost of caring for a loved one, as well as potential lost hours in their own work.

The addition of a robotic companion can also produce significant social benefits to users by adding a new source of social interaction to everyday life, as well as acting as an assistant for users looking for additional help with daily tasks. Providing a robotic assistant ensures that users have a constant source of companionship from the robot who can respond to requests and queries, engage in conversation with the user and simultaneously monitor the status of the user in case of emergency. The importance of social interaction, particularly with the elderly who may be isolated or vulnerable, has been noted in multiple research studies [12], [13]. Loneliness can have a particularly damaging effect to a person's mental and physical health, with multiple cases showing increased chances of dementia, faster cognitive decline, and higher mortality risk [14] [15], [16] .

It is also important to consider the risk to physical wellbeing associated with being socially isolated. As people age, the risk of falls and accidents increases due to a decrease in motor function such as balance, while simultaneously the risk of serious injury from such occurrences increases [17], [18]. For those living alone or with limited supervision in the case of lack of care staff, there is the risk of being unable to contact help when such occasions occur.

Therefore, this thesis proposes that robotic assistants, able to provide customized care and interactions, present a viable alternative that can help to reduce costs, increase a user's independence, improve patient mood, and simultaneously aid carers with patient healthcare.

By providing these services, the robotic companion can remove pressure from both the healthcare system and the family and caregivers responsible for a patient's health. In a study conducted by the RAMCIP project [19] specific areas were highlighted detailing how robotic companions could provide the largest benefit to users and their carers. Multiple behaviours including detecting obstacles in the path of a user, being able to call for help in an emergency and monitoring a patient's medication intake were noted as high priority to users, carers and medical staff when interviewed. These examples are echoed in the UK RAS paper [7], as seen in figure 1 which highlights their examples of the ways a robotic companion would be expected to help to benefit both patients and carers.

However, this level of care and interaction requires a level of autonomy and social intelligence on the part of the robot. To facilitate this, a software architecture has been developed that will include a growing cognitive memory for learning and recalling behaviours learnt by the robot, a human-aware navigation system for mapping and navigating both new and mapped environments and a social intelligence system to allow the robot to communicate with users.

In addition, its body-agnostic setup means the system can be divided into individual software modules and applied to different devices depending on the needs of the user, which has been done in order to make it as accessible as possible. As it would be unrealistic to expect every patient to be able to instantly access a robot, particularly for individual home use, nor could care homes afford to purchase whole groups at once to completely manage the taking over of tasks, this provides a way of allowing gradual adoption or the use of smaller versions of the companion.

The remainder of this thesis is divided into the following chapters that explain the research completed over the course of this PhD. In Chapter 2, a review of the biological inspiration behind this project is detailed in an in-depth literature review. The review also provides an overview of existing implementations of robotics in healthcare and the impact they have had on patients, as well as the challenges commonly faced when introducing robots into both homes and care environments. Chapter 3 presents the purpose of this work from the perspective of the end user, explaining in detail the different aspects of patients' daily lives that would be affected when introducing robotic companions. In addition, the benefits to existing carers are also discussed. Chapter 4 then presents a detailed account of the cumulative biologically inspired memory system and how its addition to the system allows the robot to continuously learn from encounters with users, store and recall previous experiences to predict possible outcomes, and autonomously plan actions based on observations of the environment. Chapter 5 presents a human-aware navigation system that allows the robot to map and navigate through unstructured and dynamically changing environments such as care homes and users' homes as well as provide aid to users in locating objects, directing users to different rooms or by providing general information about the environment. Chapter 6 then outlines the larger software architecture, including how individual software modules communicate with each other and details the results of experiments conducted to show the effectiveness of the overall system. Finally, chapter 7 presents a final overall discussion of the project to summarize the achievements of the thesis and the future applications of the work.

2 Background and Literature Review

In this chapter, an extensive literature review has been completed to gain an understanding of the background of both the computational aspects and the neuroscience-based aspects involved in the development of this cognitive architecture. There were several areas of research targeted in order to gain a suitable level of knowledge of the current status of both areas, and aimed to include an exploration of the following; 1) The current understanding of biological episodic memory and the connection to the human brain, 2) The biological process of navigation in both humans and animals and the resulting ability to form goal-directed directions, track landmarks and explore unstructured environments, 3) The current research and results of applications of various navigation techniques to robotic platforms and their various advantages and limitations, 4) an overview of robotics in healthcare, with specific focus on existing care-based social robots and the levels of success with a target audience, and 5) finally an overview of the scientific and technical challenges facing the integration of a robotic companion into a user's everyday life in a safe but effective way.

2.1 Biological Episodic Memory

The ability to learn from actions and personal experience is one of the most important features of any cognitive architecture. Whether the system is biological or artificial in nature, being able to recall the past outcomes of specific actions provides a useful way of predicting the possible future and consequences of similar current behaviours. Additionally, having knowledge of past attempts to complete goals and perform actions provides a foundation to calculate how best to adapt to both new and similar situations. The ability to learn through experience, as opposed to being provided with all information prior to an event, creates a way to autonomously learn new behaviours over time, continuously increasing the overall knowledge of a system and further increasing the ability to make a larger range of goal-based decisions in the future.

From a biological perspective, the ability to encode and remember an individual's personal experience is attributed to episodic memory, a form of autobiographical memory extensively studied in humans [20]. The distinction between episodic memory, which involves the encoding and recall of personal experience, and semantic memory, a memory type focused on general knowledge and facts, was first made in the works of Endel Tulving [20], [21]. It is through this particular type of autobiographic memory that the 'when and 'what' context elements of personal experiences are encoded to create a long-term memory.

The role of the hippocampus within episodic memory has been the topic of many studies across the neuroscience field. Research has concluded that the hippocampus is a particularly critical element in terms of both encoding and retrieving learnt experiences, as well as forming associations between new memories that have been recently encoded, and those that were encoded in the past [22]. It is this role in inferential reasoning that makes the hippocampus such a vital part of episodic memory, as it allows the formation of new answers to novel requests based on the information learnt from previous, similar tasks.

The realization of the hippocampus' critical role in the formation of memory was documented in 1957, when Scoville and Milner published findings achieved through experiments with a patient, H.M [23]. The patient had had part of the hippocampus removed in an attempt to cure their epilepsy; however, it was soon noted that the patient appeared incapable of remembering the day's events, even when they had only recently occurred, showing an inability to properly store and retain recently formed memories. Another notable case can be seen in the research conducted on patient E.P, a patient with severe bilateral damage to the hippocampus, and the subsequent effects on their ability to encode new long-term memories [24]. As part of the research, the patient was asked to recall the area where they grew up, before the damage to their hippocampus occurred, which they were able to do with reasonable accuracy. However, when asked to recall their current neighbourhood, an area they relocated to after damage occurred, the patient appeared unable to remember any details of the area despite having lived there for several years at the time of questioning. The research concluded that this was significant evidence of the role of the hippocampus, along with other components of the medial temporal lobe, are required for the formation of new long-term memory in humans. They also made the distinction that this was not the case in terms of memories that had been encoded before the damage occurred, and therefore the same components did not appear to be required for retrieval of remote memories. The research also has consequences for the understanding of spatial memory, which has been discussed later in this chapter.

Debate as to whether this ability is present in animals as well as humans was also a popular and somewhat controversial topic throughout this research. While several studies have claimed to provide evidence for episodic abilities in various mammals, others argue that many of these cases could potentially be explained by non-episodic reasoning and therefore unreliable as proof of episodic memory in these animals. In the work by [25], monkeys appeared to display the necessary ability to distinguish the ‘what’ and ‘where’ elements when required to choose between complex visual images on the basis of whether a particular object was displayed in a specific location. However, as stated in [26], an alternative explanation was that rather than using abilities related to episodic memory, the monkeys were merely forming associations between the reward they were given for successful choices of particular what-where configurations over the course of the multiple trials.

Again, in further tests in rodents documented in [27], normal rats were compared to those with hippocampal damage in order to test their ability to judge inferentially across stimulus pairs with a common element, in order to judge its role in declarative memory. It was found that those with damage were unable to perform this ability seemingly proving the hippocampus' role in declarative memory.

In humans, this ability to infer another's actions and learn from them seems to appear at an early age. Young children can be shown to observe the actions of others around them using a 'like me' approach to study the event, using the other's actions to help them understand their own physical abilities [28]. The ability to infer possible relationships between events and actions is critical for a robot that would be required to work in complex social situations in order to adapt to tasks and continue learning in order to better assist the user. This is particularly important when considering the development of a companion robot as whether the robot is placed in a domestic or care situation it would be a necessary ability to be able to interpret or anticipate what actions a user is about to take so that appropriate aid may be provided early and without unnecessary prompts from the users.

The ability to adapt quickly is vital not only for a sense of autonomy but also for tasks that require cooperation between the robot and a human patient and tasks that require the individual to complete goal-directed actions.

Performance of goal-oriented behaviours is something that can be seen in humans and animals alike with both showing a tendency towards these behaviours whether they are alone or within a shared space [29]. Multiple experiments with various animals have shown the ability to understand not only a goal but also the necessity of cooperation to reach it. A study by [30] shows that through trial and error, chimpanzees were able to work with a partner in order to manipulate ropes to reach food. During the experiments, the chimpanzees appeared to display an understanding of the necessity of performing seemingly unrelated actions (manipulation of the ropes) in order to reach and perform their actual goal (obtaining the food-based reward). The observation that multiple chimpanzees were able to work towards the same goal also implies the ability to recognise not only the actions that another being was making but also the ability to anticipate the possible outcomes of these actions.

In another similar task, a pair of Asian elephants were also tasked with manipulating a series of ropes to achieve a goal. In this particular trial, in order to reach the final goal the elephants would need to act in conjunction with one another as opposed to working on the task alone. The conclusion of the experiment was that not only were the animals able to work together to complete the goal, but also appeared to understand the necessity of their partner being present to achieve this. This was demonstrated when in one case, the animals were observed appearing to wait for their partner to be present before they began attempting the task [31].

Similar goal-directed behaviour can also be seen in individuals when working alone. In one of Aesop's fables, The Crow and the Pitcher, the attempts of a thirsty crow to reach some water in the bottom of a pitcher is described in the form of a story [32]. While the crow cannot fit inside the pitcher, it is eventually able to drop pebbles into it, one by one, until the water is at a level he can reach.

While the fable is just a story, the tale has inspired a number of studies using the New Caledonian crows and a puzzle of a similar kind. The real-world study conducted in [33] was based on having crows attempt to reach floating food, which could only be done by raising the water levels. The crows were shown to quickly begin using objects that matched their desired functional purpose. Through this, it was shown that based on the link between objects and 'rewards', the crows were able to use novel tools to reach a goal. Similarly, in [34] crows were presented with multiple options including tubes containing both sand and water, and the choice between floating objects and those that would sink, Caledonian crows were able to demonstrate an understanding of the actions and requirements needed in order to complete their goal. Additional experiments with jays and rooks appeared to reach a similar conclusion that these birds were capable of understanding the given goal and displaying the required reasoning needed to complete them [35], [36]

While being able to recall, encode and adapt to experiences based on previous scenarios is clearly a critical part of decision-making, research has suggested a crucial link between episodic memory and the navigation of the social context of a scenario [37]–[39]. Similar to the research previously explored, navigating through social situations in a way that seems acceptable to society also benefits from recalling successes or failures in previous situations, and inferring the appropriate actions to take in response. For daily life, understanding the social context within an environment, predicting the reactions of others to current actions, and adapting behaviour to the people present are all key abilities to behave in a socially intelligent manner.

One way the brain is able to accomplish this is explained in the work by [40], where the authors propose that the hippocampus provides an additional purpose within memory by providing a way to 'map' social spaces leading to the ability to adapt to new social contexts.

The apparent connection between the role of the hippocampus in not only recalling and predicting behaviour through its connection to episodic memory but also the potential of navigating social actions provides an interesting consideration for applications in care robotics. For a robot to be able to implement a system that could infer connections between actions connected to objects, people and location as well as the social implications of these actions could be a possible way to create a social companion capable of interacting in a socially meaningful with users while also continuing to learn personalized action sequence for user-specific goals.

2.2 Biological Systems for Navigation

In 1948, Tolman presented his research on the concept of cognitive maps and their role in navigation capabilities in the brain [41]. He described the concept as a way of representing the current environment as an internal model which could then be used to determine appropriate movements through said environment. Focusing his initial research on rats, he proposed that rather than relying on simple one-to-one associations between elements in the environment, the rats instead were able to organise stimuli into a map-like structure, which they used to help understand their environment. This mental internal model of the surrounding area then helps to store spatial information and connections between objects, landmarks and locations which can be used for route planning.

Similar to the research presented on episodic memory, evidence suggests that the hippocampus is also vital to the formation of these cognitive maps and remembering physical locations [42]. In particular, the discovery of place cells and their place in navigational ability provides a significant link between the hippocampus and the ability to form spatial maps [43]. These specialised cells have been shown to activate as a mammal moves to particular sections of an environmental location regardless of the orientation of the animal and are directly involved with route planning, with routes between both known and novel start and goal points able to be calculated [44], [45].

The discovery of grid cells led to further developments in the understanding of biological navigation. Identified in 2004, grid cells can be found in the entorhinal

cortex, a section of the brain located under the hippocampus' location in the human brain [46], [47]. As animals explore an environment, Grid cells are proposed to play an important role in encoding the metric distance, making them essential for path integration as they track the changes in distance across particular directions. Their role in vector navigation also means they play a critical part in route planning through unknown environments, though research suggests this becomes more challenging with the introduction of obstacles in the environment [48].
These specific neurons, along with place and head-direction cells in the brain,
are believed to be major requirements for the formation of the cognitive maps previously introduced by Tolman [43]. The concept of an internal map was further supported by work which revealed the existence of a type of entorhinal cell, referred to as border cells
A notable finding of this work was that border cells which are activated when an animal is in proximity to the border of the environment which they suggested may play a role in how place and grid fields are attached to a geometric frame [49], [50].
Another notable finding of this work was that border cells appear to be influenced by visual landmarks in the environment. The use of landmarks in navigation is crucial as they act as reference points within the environment and again, are a key part of the formation of cognitive maps [51]. Outside of being the location of place and grid cells, the hippocampus has also been found to distinguish between overlapping routes, a function that is necessary when navigating to a correct goal point when presented with multiple potential paths to a goal [52], [53] . The importance of this can be seen when attempts to recall a correct path to a goal, as the overlapping routes can cause interference when calculating routes.
To avoid errors, the system must be capable of distinguishing between the
different pathways. Some studies have presented further evidence to suggest that in order to perform this function, the hippocampus relies on contextual information from the environment.

The context aids in the recall due to the initial routes including unique information related to that specific context. This then aids in the separation of routes that are found to intercept each other at the moment of recall [54].

However, while the hippocampus has been shown to be vital for some spatial navigation tasks, its actual role in path integration, one of the main forms of navigation has been debated. Work by [55] found that rats with serious lesions on the hippocampus was able to complete a path integration task in darkness, suggesting that path integration can be performed even with hippocampal damage.

Applying what knowledge there is of biological systems to artificial ones presents a significant challenge and has produced multiple solutions across various studies.

One of the most significant biological methods of navigation is path integration, which can be observed in both humans and animals to varying degrees [56] . The method involves a being continuously monitoring its own location with respect to a 'home' vector, such as a nest, and using that information to find its way back along the same route [57]. The method provides the crucial ability for an animal to return to locations without the use of external cues. However, the method is prone to errors, with large errors often appearing in direction headings and the calculated distance that had been travelled. To counter the created errors, it is possible to use landmarks within the environment to 'reset' the current position. Known locations of landmarks provide a reference point that can be regularly checked in order to check the current the position is accurate and corrects errors [58]. This is especially applicable knowledge within the context of this thesis as the navigation method (see Chapter 4 for explanation) follows a similar method for having the robot navigate the unstructured home environments and has interesting potential for solving the issues found during experiments with the Pepper robot as well as being directly relatable to how the navigation model is calculating the correct pathways for the robot.

2.3 Applications for Robotics

One of the most recognised methods of navigation in robotics is the use of 'dead reckoning', where odometry data collected from a robot's sensors is used to track the rotation of a robot's wheels as it navigates an environment. The data collected from wheel encoders helps to track the position of the robot by calculating the rotation of the wheels over time as the robot is moving through an environment. The method itself can be compared to the biological method of path integration previously discussed due to the similarities both methods share in the way the moving being (animal or robot) is tracked compared to the starting location. When applied to a robot, the 'dead reckoning' approach has the advantage of being a relatively simple method that collects real-time measurements of a robot's current position and takes neither a great amount of computational power nor time to collect, only requiring that the platform has the encoders necessary [59]. However, while cost-effective compared to other methods, it also has the notable disadvantage of being prone to inaccuracies due to interference, whether this be by wheel slippage on incorrect floor types or by collisions with or misinterpretations of other objects disrupting the readings [60]. It is possible to counter this limitation in a similar way to biological path integration by including landmark information when processing data after exploration. Another well-known approach to the navigation problem is SLAM (Simultaneous Localisation and Mapping) based methods. Traditionally, SLAM aims to provide a robot with a method of autonomously navigating and localising within an environment, through the creation of spatial maps representing the explored area .SLAM methods typically aim to answer the question as to whether it is possible for a robot to autonomously map an environment for the purpose of simultaneously navigating and localising within that map [61], [62]. As SLAM methods do not require prior knowledge of an environment in order to complete their mapping it provides a significant advantage over other navigation methods that commonly require at least some prior knowledge of the area to be navigated.

While multiple SLAM methods have been proposed, many share notable similar disadvantages with each other, with many facing issues such as the accuracy of the map generated, as well the computational power required to generate them. Over multiple research studies, several methods of how to approach the problem have been suggested with various success. These approaches include EKF-SLAM [63], [64], Graph Slam [65], [66] and Fast SLAM [67].

Visual SLAM is an alternative SLAM method which proposes a method for mapping the environment by visual input only [68], [69]. The method proposes that by extracting key features from an environment, it is possible to perform the necessary localisation functions required by SLAM solutions while only requiring a camera as a sensor. This does have important advantages over other SLAM solutions. First, the method only requires a single type of sensor increases the number of devices technically capable of implementing it. Secondly, the low cost of a camera sensor compared to examples such as lasers. It also removes the environmental limitations noted in other sensors such as GPS sensors that struggle to accuracy record in indoor environments as well as the odometry issues related to wheel slippages and floor types. However, an important limitation of the Visual SLAM solution is the high computational power required to process the visual information being recorded by the camera and the complex algorithms for feature extraction that are required. It can be argued though that the increasing advances in hardware and image processing are slowly reducing the impact of this particular limitation.

The Pepper robot by default also presents its own version of a SLAM system available through Softbank's NaoQI libraries. By using data collected from the robot's sensors, including the lasers in the base and cameras located in the head, the method forms a 2D map of the environment for use in navigation. However, the collection methods used specifically by Pepper have significant limitations. Pepper's lasers are often unreliable due to a lack of accuracy of the results they gather and often missing obstacles in the environment due to height issues or inaccurate readings. A possible solution to this is the introduction of ROS in combination with the 3D scanner built into the Pepper platform, as shown in [70] which allows for a greater understanding of the environment by providing the functionality to convert between laser scans, depth scans and point clouds. It also provides a way for streams of information to be continuously published and accessed. It is however noted that the version of ROS used in this work has issues with latency and providing real-time responses, a problem that future work with ROS 2 may be able to change.

2.4 Robotics in Healthcare

The potential of the use of robotics for healthcare is a topic that has been discussed and researched extensively over many years and the research has presented many forms of robots with various levels of capabilities and intelligence across multiple fields.

Robots can be found performing many roles throughout healthcare including assisting in surgical procedures providing high levels of precision and accuracy, as well as the potential for reduced costs of operations [71]–[73], and as rehabilitation aids encouraging physical exercise and therapy [74], [75] .

However, there is a noticeable theme throughout these studies that implies limitations to their abilities. The environments these examples are conducted in have been structured environments where behaviours and actions can be planned in advance, and do not react meaningfully to outside influence. They, therefore, possess very low levels of autonomy, as they have no need to adapt to unknown situations and most if not all, rely heavily on the human users' input to create the actions for a goal-based situation. This approach works well when the conditions of the environment are known in advance and prior information is provided to the robot regarding this, but for a robot in complicated social and dynamic situations, would be impossible to predict all possible scenarios possible within a constantly changing environment. The range of users, each with different preferences towards behaviours and reactions, creates a situation where a robotic companion’s reactions must be calculated in real-time, to ensure it can adapt to sudden changes in the context and unforeseen requests or inputs.

Telepresence robots can be found in multiple studies [76] . Usually designed to help maintain social contact between users and family members, carers or other healthcare professionals, these robots usually contain video and audio calling features that allow incoming and outgoing communication. While often limited in terms of intelligence and autonomy, evidence has shown that these robots do have positive impacts in terms of increasing social engagement and helping care home residents feel more connected to family members and friends. Notable limitations of the platforms, other than the intentional lack of advanced features, have been recorded as including the lack of privacy within shared spaces, the space and battery requirements compared to simple mobile phone/tablet devices and potential connection issues. It is, however, important to note that the separation of the functionality usually gained through physical devices (phones, tablets etc.) onto a robotic platform could be considered an advantage for those with physical issues and limited mobility.

Multiple examples of care robots presented as animals can be found in a large range of studies. Robotic pet therapy has been found to show similar positive effects for patients as seen with real animals. Perhaps the most well-known example of such a robot is PARO, a small 'pet' robot designed with the appearance of a seal. PARO has featured in multiple studies for over a decade and has helped further understanding of how robots can participate in the daily lives of users. Throughout the various studies conducted with elderly users, often suffering from dementia, PARO has been shown to have positive effects such as lowering anxiety, reducing stress levels and even produced some evidence to show a reduction in pain levels in patients that interacted with it [77], [78]. Importantly challenges to the results of these PARO-based studies often include the lack of data on the novelty aspect of the robot and whether engagement would be as high when the robot is no longer seen as a new addition.

Humanoid robots, such as the Pepper humanoid tested within this thesis, as well as Pearl [79] and Sara [80] are also popular choices for companion robots. Humanoid robots commonly present the advantage of physical movement when compared to smaller animal robots or fixed-placed robots. Both Pepper and Sara have the option of speech and can interact with users creating the potential for conversation, an important aspect of the research goal is companionship. There is also the advantage of a larger range of sensors able to be built into the platforms, providing a more detailed view of the world that can be used for later processing. They do, however, have limitations when compared to smaller, less specialised social robots. 2-legged smaller humanoids, such as Nao [81], have the ability to move yet not for long distances. Pepper often experiences issues with audio and understanding users, particularly in crowds and also is large enough that physical harm could be caused to users if not monitored. Additionally, a number of ethical concerns in regard to the processing and storage of conversations and interactions become apparent when using social robots for longer periods of time, partially those equipped with cameras, audio processing and higher levels of social intelligence [82]. These issues are discussed further in Chapter 3.

2.4.1 The Influence of COVID-19

During the recent COVID-19 pandemic, countries worldwide were forced to adapt to the spread of a virus that was not only highly contagious but also posed a serious threat to the most vulnerable of the population. As a result of the infectious nature of the virus, many countries were forced to implement lockdowns for their population in an attempt to slow the spread of coronavirus and protect the most vulnerable. The consequences of the pandemic were not limited to people's health, but also social and mental wellbeing [][]. The effects of sudden social isolation and an overwhelmed healthcare system created serious consequences for both those stuck in their homes as a result of lockdowns, those forced to isolate due to being contagious and the in general for healthcare workers who had no choice but to expose themselves to the risk of infection in order to continue helping patients.

It was due to this isolation and need for physical distance that robotics once again became a promising solution for healthcare-related issues [83]. The advantages of companion robots, particularly for the very vulnerable, presented an interesting solution towards combatting the negative health and mental effects associated with isolation [84], [85]. The use of robots in general also had an impact on sanitation levels and preventing infection as it allows patients to be treated from a distance, enables the distribution of food and medicines without physical contact and helps to prevent the spread of infection to vulnerable patients as well as from COVID-infected patients to their carers. The use of robotics in general for decontamination and cleaning also had a large impact on keeping areas free of infection while reducing the risk of infection to both patients and healthcare staff [86], [87].

One particularly notable outcome of COVID-19 was the rise in approval of social robots in general. Multiple studies have now concluded that during the pandemic, people's opinion of the use of robots for care work significantly improved and demand for care robots increased [84]. This increase was largely connected back to the effects of isolation and the need for companionship to combat loneliness and prevent decreases in mental health.

2.5 Additional Challenges for Robotic Companions

2.5.1 Cost of Purchase

Creating social companions while also keeping the cost low enough that people can afford them is a big challenge in the development of social robotics. While more technologically advanced robots can provide a larger range of functionality and therefore care, they also increase the price to the average user, making them inaccessible to most 'everyday' people. The cost of robotic platforms is a significant purchase, and for individual people living independently, it is difficult to provide a platform that can be considered affordable as an initial purchase.

In addition, there is also the possibility of maintenance and upgrades. Repairs, technical errors and breakdowns could all potentially provide additional costs to users. To balance this concern, the software developed should be adaptable to users' individual needs by allowing users to customise their system and only purchase the functionality they can place on devices already owned.

2.5.2 Power Consumption

On both technical and financial levels, one of the main challenges when attempting

to implement robotics into people's homes or into care facilities is electrical power. For a robotic platform, there will be a requirement for it to charge its battery on a regular basis, particularly if the robot is used continuously within an environment. For technical considerations, this requires users to have a charging point, easily accessible by the robot, that can remain available for the robot to use when needed potentially causing issues with space that would needed to accommodate this and in general, affecting the robot's portability. Events uncontrollable by the user, such as power cuts or damage to power supplies, will also potentially limit the robot's ability to function and even cease to function should the battery remain unchanged for significant periods, leaving the user without the needed care. In larger facilities such as care homes, this could cause significant disruption depending on the duties that had been assigned to the robot, such as medication distribution or companionship duties.
It would also be necessary to once again consider the financial implications for a user keeping the robot in their own homes. The increase in power usage will raise electric bills, though this could potentially be countered by the financial gain of not paying for human carers as often or even at all.
The challenge then also becomes greater when considering for larger environments, more than one robot may be required in order. In this situation, both technical and financial concerns are increased.

2.6 Pepper and the NAOqi

Throughout this thesis, the Pepper humanoid has been used for multiple experiments and validation testing. Pepper is a humanoid robot originally created by Softbank Robotics and chosen for this research on account of its range of software and hardware components.

Two versions of Pepper were used throughout the research. The first, used over the course of the first two years of research, was a Python-based system requiring the use of Python software libraries and ROS 1 for communication between modules. The second, used in the third year, consists of Java software libraries and requires software to be developed as Android applications in order to run on the robot with connections being moved to ROS 2. The change in hardware and languages was made due to the removal of Python 2, which the first Pepper required as well as issues with the hardware of the first Pepper robot. This began creating issues when trying to integrate newer software libraries, which often required Python 3 environments or a later version of ROS for communication.

Both Peppers have 20 degrees of freedom in the body and include multiple sensors and actuators including:

- 2D cameras
- 3D depth camera
- Microphones
- Lasers
- Sonar sensors
- Tactile sensors in the hands and head

The on-board NaoQi software framework utilises these hardware components to provide needed functionality such as object identification and detection, face detection, speech capabilities, navigation and functionality for an Android tablet attached to the chest.

3 From the Perspective of an End User

While it is important to consider both the technical and biological motivation for this project, it was also vital that the end-users specific needs were studied and considered. Technology aimed at elderly users is going to be significantly different from what would be aimed at children or younger adults, not only in terms of ease of use but also in how the technology would be required to adapt to the specific needs of this particular demographic.

Being able to detect and recognise humans within an environment is an essential feature for a social robot designed for companionship for multiple reasons. The first key reason, directly related to this thesis, is the ability to perform personalized responses and react in a way that is customized based on the person the robot is interacting with. Multiple studies have shown that it is vital that a robot in a care setting is able to react in ways that match the current context of the situation, in order to provide efficient levels of care [88], [89]. Being able to provide personalised care enables the robot to safely help with user-specific tasks, such as medication reminders, suggested actions to achieve a goal and recognising behaviours that could be considered harmful to the user. This is also important for users in environments such as hospitals, as it can be expected that there will be a wide range of diverse personalities and needs making pre-programmed responses impossible due to the number of possible solutions depending on the person's own health requirements. By providing a way to recognize individual humans and connect them with either past experiences or associate them with certain situations, it allows Pepper (and social robots in general) to respond to a request in a way that the user not only finds most useful but also helps to increase the likelihood that the robot will be accepted by the user [88], [90].

3.1 Adapting for Accessibility through a Multimodal Interface

In order to make the system as accessible as possible for a diverse range of people, there has been significant consideration of how the robot should communicate with the users, as well as how to allow them to communicate and respond in turn.

In order to be as effective as possible, a multi-modal user interface has been integrated into the main architecture to ensure that users can interact with Pepper in multiple ways depending on their preference. the companion is provided with a multimodal user interface that allows users to interact with Pepper in multiple ways. An Android tablet, attached to Pepper's chest enables the user to view options on the screen and select options commands by touch and accept typed commands. As the tablet is developed through Android, it is also possible to connect other Android devices to extend the range that users can interact with the robot. Alternatively, users can also interact through speech. The QiSDK libraries, provided by Softbank and used by the Pepper robot provides listeners that can be embedded into the application to detect certain phrases when uttered close to the robot. Hearing these phrases forces the robot to redirect its focus to the users and can provide a response from a list of answers. To expand this speech functionality, a chatbot built in Dialogflow has also been developed which provides the robot with a larger variety of answers and the ability to learn from speaking with users, which allows users to hold conversations with the robot and helps with providing companionship.

3.2 Ensuring User safety

As the robot will interact directly with multiple users, some of whom would be classed as potentially vulnerable due to age or health, user safety must also be a priority throughout this project. However, for interactions with social robots, safety can mean multiple things. Preventing physical harm is one aspect. The robot should not be a risk to users, including avoiding causing harm through collisions or creating a trip hazard to users. In this case, the concerns can be addressed in different ways. Pepper, for example, contains inbuilt, customizable distance limits which force the robot to automatically stop should it detect any obstacle, human or otherwise, in front of its sensors. We can also address this further by allowing the robot to attempt to predict a user's path in order to keep a particular distance when humans are actively moving, reducing the speed the robot moves to reduce the chance of collisions and limiting the movement of the robot's limbs to prevent sudden movements.

Further safety considerations would be ensuring the robot can call for external help when required. For social robots like Pepper, who have not been designed with physical aid in mind, there are limits as to how much aid can be given in the event of falls or accidents. Ensuring the architecture has some way to contact outside help, whether that is relatives or emergency services, is one way to ensure that help is always available to users should they need it.

3.3 Ethics and User Dignity

The ethics of careobots and social robots in general is a topic that is widely debated in current research. One of the key concerns when interacting with a robotic companion is user consent [91]. If the robot is being considered for use in an individual's home, the user should be made fully aware of how the robot will attempt to socialize with them, how the robot will move throughout their home and what kind of assistance the robot can and cannot provide. This could be through tutorials or trials prior to fully installing the companion to allow users to experience the difference with a robot before they fully commit to having a permanent companion. Alternatively, in cases where more than one user is present, such as care homes, new patients should be made aware of the robot before entering or given the choice of not interacting with it if it is presented as a new addition [91]. User consent becomes more challenging when considering health challenges that users may have, such as dementia or mental health challenges if these conditions prevent the user from making an informed choice and whether the choice is being made by the user or by family members [92], [93].

A key consideration when developing robotics for care or any form of social interaction is the dignity of the user it is aimed at. Previous research into social robots has demonstrated a common fear amongst users; that the robot would be a replacement for what would have otherwise been human contact [94]. For this reason, it was important to provide the robot designed in this project in a way that would give users confidence that the robot was there to extend their social interactions rather than replace them. This can be done in several ways; one is to have the robot working in an environment where it shares its duties with a human carer, such as in care homes, therefore providing a constant companion role while also ensuring the user sees that others are still present and that the robot has not replaced a human. Another would be to build features into the robot that enhance the user's ability to contact others, such as the ability to call family and friends through the robot, view video calls on the robot's tablets, and provide contact to medical help so the user remains confident a human is always reachable, and they have not been abandoned.

At the same time, the robot should not presume anything about the user it is interacting with. Assuming the user is lonely or socially isolated can frustrate and offend users which further promotes the argument that the robot should be aware of the context of the interaction and be able to adapt to the varying social needs of different users. Additionally, the robot must be respectful of users. Making demands or trying to be overinvolved in users' lives will create resentment rather than provide aid. This can potentially be countered by allowing users to personalise their care, choosing which services they would like the robot to perform (or alternatively, being able to block specific actions from being performed) and allowing the option of choosing how the robot should address them (e.g., the choice of name, optional title, nicknames etc.) [95].

Alternatively, some research has suggested that when developing robotics, especially when the end user will be classed as vulnerable such as the elderly, attention should be given to ensure there is no deception towards the user on the part of the robot [96]. While deception is often thought of as a malicious act, the authors of [96] suggest this could instead be through misperceptions of the robot's abilities or emotional connection towards the user. Evidence has shown that users interacting with robots often form emotional attachments towards them, even when the robots are designed for non-social roles [97], [98]. While the goal is to have the user accept the help of the robotic companion, becoming overly attached to the point of reducing contact with others in favour of the companion or relying too heavily on the robot for decisions it is incapable of making, potentially risking their safety or judgement [96], [99].

3.4 Ensuring User Privacy

When interacting with users, it is important to consider the value of privacy and address any concerns that may be present about how much the robot is seeing and hearing while going about its daily role. Continuing with the idea of ensuring user dignity, there may be concerns on the part of users with how the information they give to the robot is used. In order to assist users effectively, the robot would require an almost constant feed of data which it would store as part of its experience at a given time. The objects it views, locations visited, and the identity of users are all considered by the robot to be useful information and therefore are stored for potential recall. In addition, it is also possible that some users will willingly give personal information to the robot while in conversation, discuss sensitive topics they may not be comfortable discussing with others, and perform actions in front of the robot they would not be comfortable doing with human companions (washing, dressing, etc.) and it is important to ensure that any private information shared with Pepper is not easily assessable by other users of the robot in the case of shared use (such as a care home) [100], [101].

While the assistance of the robot could be invaluable to someone retaining their independence, it should not simultaneously be an unwelcome intrusion into their private life. Family members or carers should not be able to access information without the user's consent, similarly, the robot should not automatically share information about specific users with anyone (other users, carers, visitors) without first receiving approval from the user in question.

3.5 Financial Cost to Users

Creating a social companion that effectively meets the needs of multiple users, while also keeping the cost low enough that people can afford them is a major challenge. While more technologically advanced robots can provide a larger range of functionality and therefore care, they also increase the price to the average user, making them inaccessible to most 'everyday' users.

On both technical and financial levels, one of the main challenges when attempting to implement robotics into people's homes or into care facilities is power. For a robotic platform, there will be a requirement for it to charge its battery on a regular basis, particularly if the robot is used continuously within an environment. This requires users to have a charging point, easily accessible by the robot, that can remain available for the robot to use when needed. Events uncontrollable by the user, such as power cuts or damage to power supplies, will also potentially limit the robot's ability to function and even cease to function should the battery remain unchanged for significant periods, leaving the user without the needed care. In larger facilities such as care homes, this could cause significant disruption depending on the duties that had been assigned to the robot, such as medication distribution or companionship duties. It would also be necessary to consider the financial implications for a user keeping the robot in their own homes. The increase in power usage will raise electric bills, though this could potentially be countered by the financial gain of not paying for human carers as often or even at all. The challenge then also becomes greater when considering larger environments, where more than one robot may be required in order to meet the needs of several users. In this situation, both technical and financial concerns are increased.

4 A Customised Multimodal Network for Spatial Navigation

For an animal in the wild, successfully navigating unstructured, dynamic environments is an essential trait relied on for survival. The ability to navigate back to a nest or known food source is crucial and often relies on remembering specific landmarks or tracking movements from an original origin point. While humans can now increasingly rely on technology to prevent us from getting lost or to tell us the way home, being capable of navigating dynamic spaces is just as important when faced with crowds or unfamiliar spaces.

This can be a challenge when considering robotics aiming to operate in a shared space with humans due to the potential of changing environments and unpredictable movements. For a robot that proposes the ability to guide or fetch, these issues create several challenges when trying to complete its goals. It needed to be able to map unknown spaces and revisit those maps in order to traverse known environments, while simultaneously adapting to any potential changes in these mapped environments. For example, if a robot maps a user's room it must be prepared that objects counted as obstacles on first observation, may have moved by the time the robot revisits.

Therefore, they must be able to successfully move through environments that are continuously changing while also autonomously adapting to these changes.

In this thesis, it is proposed that this can be achieved by using a swift learning, growing multimodal neural gas network (MGNG) to create internal models of the environment based on the data received during the robot's exploration of the area.

The model is based on research within neuroscience, specifically relating to the navigational ability in animals, their ability to form cognitive maps, the ability to localize and detect familiar environments and how goal-directed behaviour can be achieved. Previously, the base model for this experiment had been used in [102], where it can be seen modelling the 3D peripersonal space of robotic arms to enable the use of goal-directed cooperation in assembly tasks. The work for this model has since been modified and expanded on over the course of this research to implement it onto a Pepper robot, model internal environments to create expandable maps and perform goal-directed navigation using multimodal nodes and route planning through the space. The results of this can be seen in the following chapter. Experiments were performed in both simulation and real-world environments to confirm the robot's ability to perform goal-directed behaviours and navigate both new and previously explored environments after receiving requests from users.

The following chapter will explain the methods used for creating the navigation module as well as provide explanations of the experiments conducted on the humanoid Pepper and the consequential results.

4.1 Swift Multimodal learning of an internal model of the Environment

To train the network model and build an overall map of an environment, an initial exploration is performed by the robot in order to gather multisensory data from the platform sensors. As the platform chosen for this work was Pepper, the data is collected through an Android Application specifically built to work with Pepper's Android tablet. In the final version of the project, the application is built for human-guided data collection, where the robot is initially led through a new area by a human and collects data on the areas it is guided through. Earlier versions of the project, built using an older Python-based version of Pepper, instead relied on random exploration. During early testing, it was discovered that though random exploration required less physical activity from the user, it also had more incidents of failure when mapping when compared to the human-guided version. This was due to the unpredictable nature of the user's environment as well as a hardware issue related to the sensors on Pepper. While the object detection integrated into the random exploration algorithm performed well during tests, the robot was often slowed or stopped entirely due to obstacles going unseen by the robot's sensors. This resulted in the user still having to observe and interject during exploration, therefore it was decided a human-guided version would be more appropriate due to the time saved and accuracy of the data collected, despite the physical element required.

While the language and collection method changed between upgrades, the type of information collected remained largely the same. As the robot moves, it gathers spatial information about its surroundings including the location of the robot, represented in (X, Y, Z) coordinates as well as visual information from landmarks the robot has recognised through object detection during the mapping. In addition to the Coordinate locations, landmark information and orientation of the robot all being recorded automatically during exploration, there is an additional chance for the human guide to add further context. There is an additional option for the human guide to enter the name of the room the robot is currently in, to provide an extra level of data for the final collection. This ensures that related spatial, visual and room information are associated correctly and allows users to give rooms as goals for the robot to navigate to as opposed to only specific landmarks or locations.

When exploration is first started, the robot will begin by first localizing itself to create an origin point that data can be connected back to. From this point on, the robot will track the vector translations between the origin and the current position in space and record them into a data file. The main benefit of using human guidance over random exploration, is it significantly reduces exploration time especially when mapping larger environments, similar to if a human has a guide when entering a new building, they will learn the area much faster. However, it reintroduces the issue of an added human component. The current method of guiding is to physically push the robot which presents a challenge to any users that are not physically capable of this. To try and solve this, work has been started to test an upgrade to the system that would allow the robot to follow the human rather than have the user physically interact. As this was more platform-specific (and varied even between Pepper robots), it was considered out-of-scope for the main focus of this work, which was focused more on the software architecture, though could be a consideration for any future work on the project.

4.2 Landmark Detection with CNN's

Throughout the project, multiple methods for detecting landmarks and processing visual information were tested to determine the correct balance between accuracy and efficiency when applied to Pepper. As the focus of this work was not on the full development of new vision systems, it was decided that an existing model would be selected, modified and integrated into the main architecture to detect both general objects and major landmarks for processing.

The first method considered and used for initial experiments was the integrated object detection on the original Pepper robot. The NAOqi libraries provided by Softbank Robotics for Pepper presented an easy way to allow users to train the model themselves and quickly add new objects to the database for recognition. This was appealing for the initial work as it allowed users to create customised object recognition databases, based on the objects most used in their own daily lives, in a simple but quick way. However, the attempts to integrate Pepper's default object recognition libraries had several issues.

First, it required all objects to be labelled manually by a user after taking an image of the object using the robot's camera. This created a significant level of human interaction just for training a small number of objects. In addition, to be recognised after this first round of training, the objects also had to be placed in a similar position and consistent lighting conditions in order for the NAOqi database to recognise them as familiar objects. Changes in the environment, such as background changes, additional objects appearing in the frame or slightly obscuring small amounts of the object while holding it to the camera prevented the robot from recognising it as the same object. The TensorFlow lite model is specifically built for mobile devices, making it more efficient within the project than using the external TensorFlow model on a ROS server, as was used at the beginning of the project. Having the vision embedded into the robot application also reduces the time between taking the image and returning the processed results, which was a major delay in the process before upgrading.

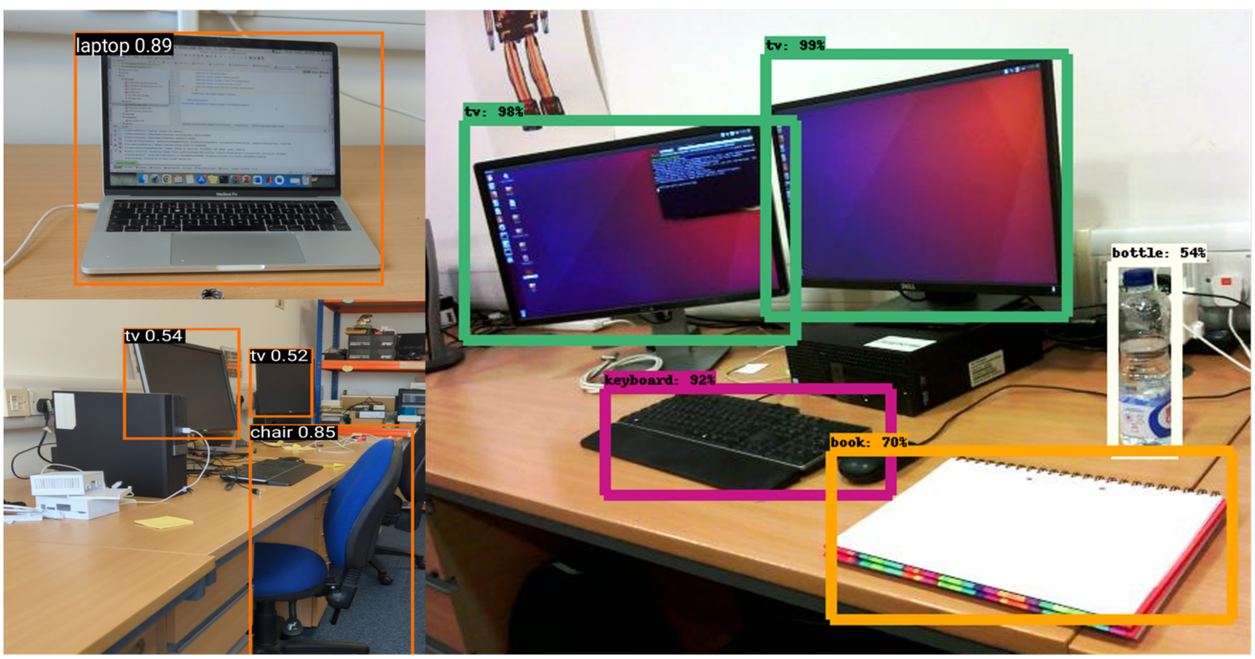


Figure 1 - TFLite was used in combination with a custom Android application developed for and installed on the Pepper Humanoid. When used in conjunction with Pepper's built-in camera, it provided a consistent stream of feedback for object detection and world monitoring. The data is used throughout the architecture to maintain a view of Pepper's immediate environment and provide context for later decision-making behaviour and navigation.

Comparisons were completed between various detection models to be used with the TFLite framework. In order to be considered suitable for recording into the GNG model, objects detected must follow several requirements: Most importantly, the confidence score provided by the TensorFlow detection must be greater than 60%. This figure was decided based on testing within multiple environments, including lab environments and the university I-space to determine a good balance between ensuring more ‘obscure’ items (smaller household items such as vases and clothing etc.) were detected but also ensuring common misdetections that would create false associations within the network and therefore create fake activations in the route planning when later used for goal-directed navigation were excluded from the exploration data. The final model used to observe the environment from the robot’s perspective was the TFLite ‘efficientdet-lite2’ model. The model has been integrated into the main user interface and provides a constant stream of object detection data from the robot’s camera to be processed as part of the overall world observation. This data is what is later included in regular map updates, used by the Observer module to monitor changes in the environment and provide context to the robot while monitoring users.

4.3 Training a Multimodal Gas Network

To build the networks within the navigation module, a growing neural gas model has been used to create the internal models of the environment. The network choice is based on current research in growing neural gas networks [77]. These networks are similar to the original self-organizing map (SOM) networks [78], with the additional benefit of being able to continuously add nodes and grow the overall network as new information is presented making it ideal for the incremental learning involved in continuously updating the exploration maps. In the experiment described below the network began empty, however as the robot explores nodes are continuously grown and added based on the novelty of the information presented. Novelty in this case refers to the distances travelled by the robot and therefore the difference between positional coordinates recorded when calculated through vector translation. Physically, during collection, the distances are calculated through vector translation involving the current position of the robot compared to the origin point. In this module, the grown network represents the internal model of the environment and additionally presents the novel addition of multimodal nodes based on the information gained through exploration. For each node that is grown, multiple elements are encoded to create the possibility of multimodal activation during future route planning and goal-directed navigation. This information can contain positions in space as well as information about detected landmarks.

The model-building process begins by collecting data through either random (robot-only) or human-guided exploration. The multisensory data is collected through multiple sensors built into the Pepper robot and consists of (X, Y, Z) coordinates of the robot's position in space, the orientation of the platform compared to the origin point and the identified landmarks, found through object detection. The data is stored during exploration in a buffer format, before being written to file in CSV format after receiving confirmation that the initial exploration has been completed. The full data file saved contains information such as the (X, Y, Z) coordinates of the robot, the orientation when compared to its original origin point, the name of the room explored (if provided by the user) and the identified landmarks.

Once the initial collection has been completed, the recorded data is passed from Pepper to an external MATLAB server process which filters the information into specific tables to be used for the network training. The two main tables used for the model building are Set A, containing the (X, Y, Z) positional coordinates visited during exploration and recorded through the sensors and

Set B, containing the IDs of the different landmarks and objects detected as well as the coordinate positions the robot was in when these were first detected. The two data sets are then trained for different purposes in the model. Set A is used to begin growing the main gas network, with nodes being added based on the distance from existing nodes to prevent locations too close to each other from being added.

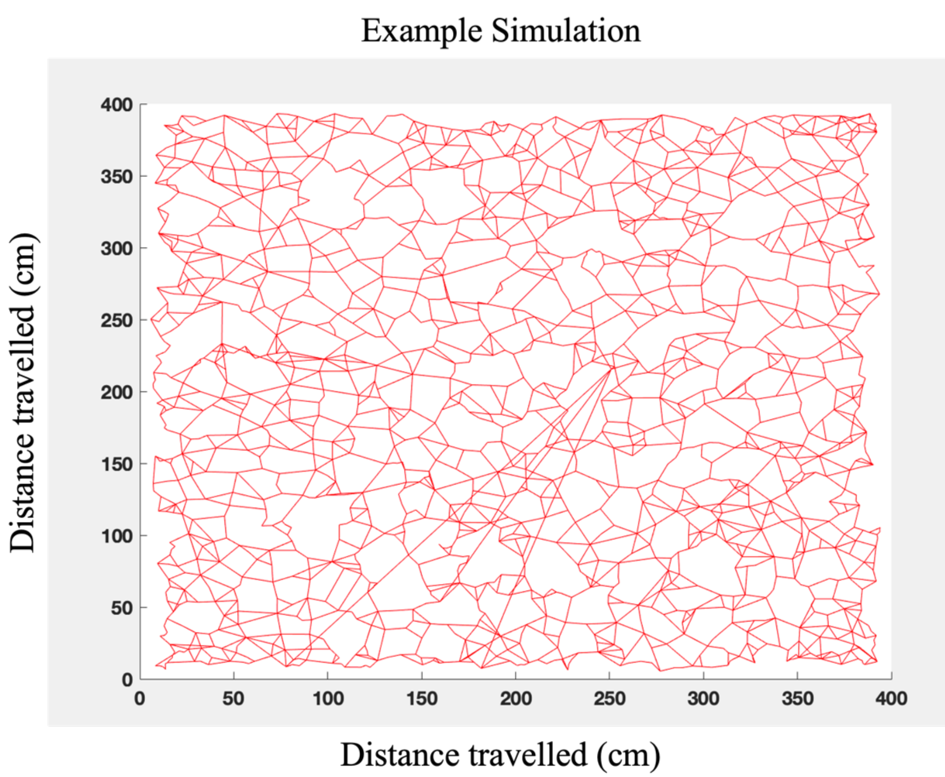


Figure 2 - An example room simulated to show the GNG network training based purely on (x, y, z) locations. Beginning with an empty map, 10000 points of data were trained with each new node added depending on the novelty of the value compared to existing spatial points already entered into the network. The result is a trained map consisting of 1668 nodes reduced from the original 10000. These unique nodes are now available to the navigation system for route-planning.

Set B's data is used in a similar way to construct smaller self-organising maps (SOM networks), specifically one to hold landmarks IDs detected and a second for the associated X, and Y positions. The individual SOM networks are connected via a larger 30x60 node network, that allows for bidirectional travel between the landmark and label hubs. The purpose of this kind of structure of networks is to enable the multimodal nodes for the network, which creates a novel ability to re-trigger nodes in the network in multiple ways. Once trained, the hubs can be reactivated based on cues given by the robot, whether this is the spatial coordinate, room ID or the name of a landmark. In addition to route planning, multimodal nodes are also used for localization, as identifying landmarks in the environment helps to provide information to the robot about its current position in space by activating the associated position in the opposite SOM hub. Structuring the networks this way also allows the robot to form a prediction about an environment, as it is able to calculate in advance which landmarks it would expect to see in an environment. The prediction ability is particularly important for recognising when a location has been altered, when potential routes may be blocked and for triggering new explorations of an area in the event a change is detected in the environment without the additional input of a human. To test this functionality, Pepper was guided around the second and fourth floors of the Network Building within the University. Starting in the robotics arena, the robot was pushed through the labs and surrounding offices, before moving into the lift, exploring the fourth floor and finally exploring the iSpace. This created a large data set with which the model could be tested. During the experiment, data was very quickly collected through the robot's internal sensors and no issues were found in connecting detected obstacles to coordinate positions when examining the collected data post-experiment. The CSV format used to store the data onboard the robot worked particularly well in allowing large amounts of data (continuous exploration across multiple floors) to easily be collected and showed no evidence of buffering issues or corruption when saving the data or during later extraction. It is however important to consider that wheel slippage was observed as a major obstacle during both random and human-guided experiments and proved much harder to correct with this method, than when compared to traditional visual SLAM as Pepper had no additional environmental elements to rely on for correcting its position until re-exploration, when the robot could recognise a change in environment and specifically, landmarks. After some additional testing, the main cause of this issue was found to be the speed at which Pepper was moved by a human guide through an environment. When pushed too quickly, it was common for the robot's wheels to slide, and the risk of the robot tipping became much higher. While there is no current fix for this in the current project (other than verbally alerting any users to the issue), future work would benefit from either an inbuilt warning alerting users before they begin the data-gathering process, or by imposing a limit to the robot's speed during data collection if random exploration is used.

Deviation from this format during testing resulted in under-trained maps and generated impossible routes (through obstacles, across large distances etc.). Notably, one main issue observed was found to be related to the amount of data collected. While the MGNG does not require a specific number of unique coordinate points for the training to be completed, it was found that a minimum number of overall data points were required in order to train a usable map. After additional training through both physical experiments and simulation, the network with current settings appears to train best when the data set contains at least 5000 coordinate points. As a result of this additional testing, a check has been included within the main network to automatically duplicate the data until it reaches or surpasses this limit. This ensures the network is able to train and form connections useable in goal-directed route planning.

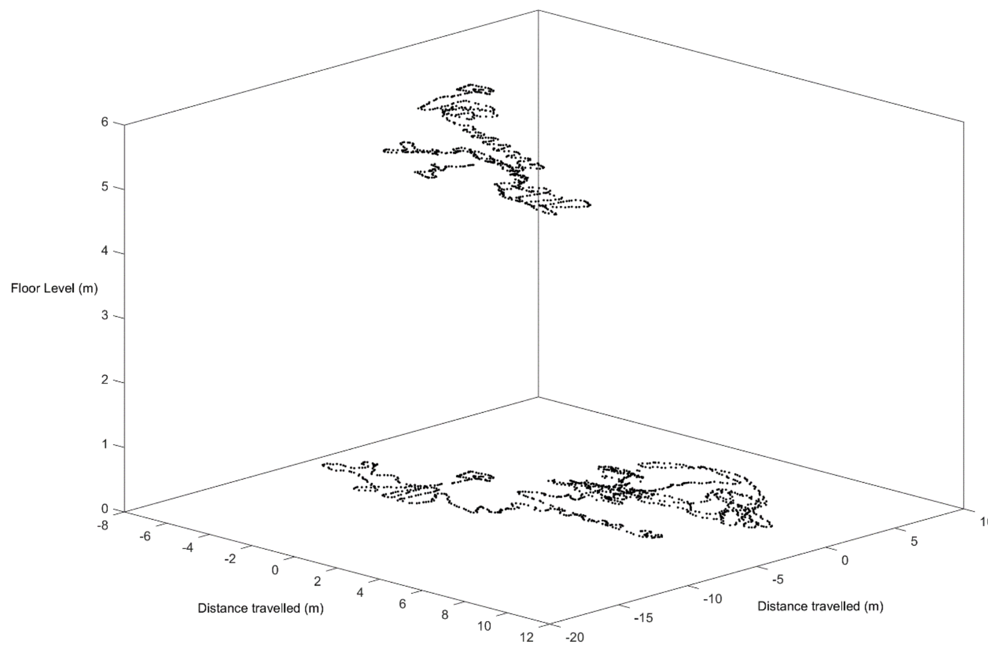


Figure 3 - The data points showing spatial positions visited by the robot during exploration of the 2nd and 4th floors of the University's Network Building. For each position the robot visits coordinates are recorded and encoded into the network, with nodes formed based on the novelty of the position compared to existing ones.

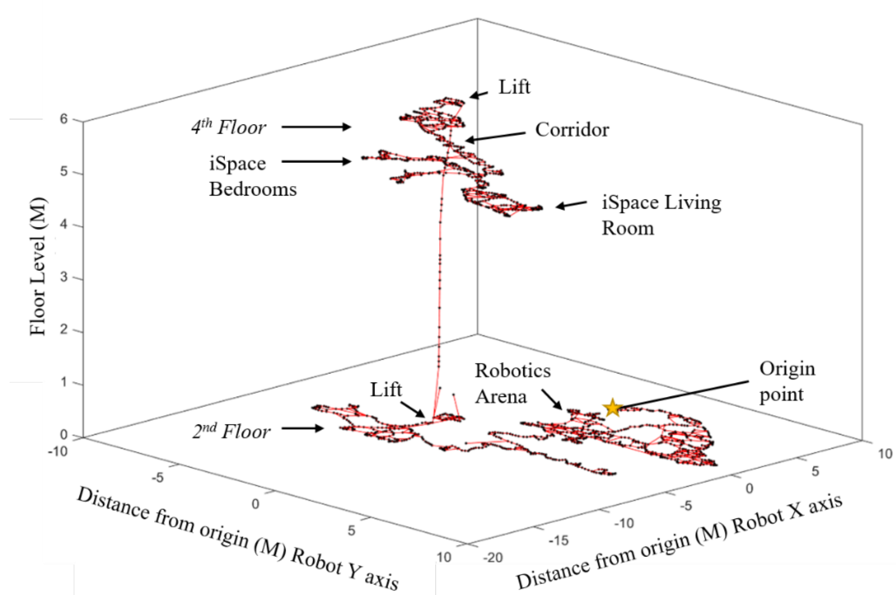


Figure 4 - After training, connections are formed between nodes which can be used at a later time for route planning.

4.4 Multimodal Goal-Directed Learning

One of the key abilities of the model is the ability to perform goal-directed navigation. The advantage of using multimodal nodes in the main model is that they can be triggered when provided with a coordinate or object name in order to calculate a route from a starting location. To provide a default starting position, the robot's current location is used as the beginning of the route, and a user-given goal is selected as the target to reach. To calculate the path to the goal point, a quasi-stationary reward field is applied to the network model which helps define a potential path from start to goal. The field applies values to individual nodes within the network, with higher value neighbouring nodes considered as possible next steps and lower nodes being ignored.

The general reward field can be defined as:

$$R(t) = \mathbf{R_{Goal}} - \sum_{a=1}^{n} R_a(t)$$

Where RGoal is a static component determined by the distance between the Goal and the physical location represented in the internal weight Si of the i^{th} neuron in the MGNG.

The effect is a reward field that is overlaid onto the original map, giving all existing nodes a value that can be used to determine the goal. With each node reached, the same process is applied to find the next step - determining the most valuable neighbour and moving towards it. This method works well when the environment does not change, however, the environments used for testing were unstructured and dynamic obstacles are expected. To account for this, another variable is added Ra is introduced the effect of another agent's movement on the robot's own goal and reward field. R(a) is defined:

$$\mathbf{R_a(t)} = \frac{1}{\mathbf{Z}}\mathbf{e}^{\frac{-(\mathbf{si}-\mathbf{S_a}(\mathbf{t}))^2}{2\sigma_R{}^2}}$$

where $S_a(t)$ is the location of Agent 'a' at time 't' and Si is the physical location represented in the internal weights of the i^{th} neuron in the MGNG. The effect of Ra is that the new agent acts as the temporary goal of the current robot, mimicking a collision. This creates the opposite behaviour than before, in that rather than aim towards the goal, the robot will attempt to avoid the potential collision due to its now (temporary) low value.

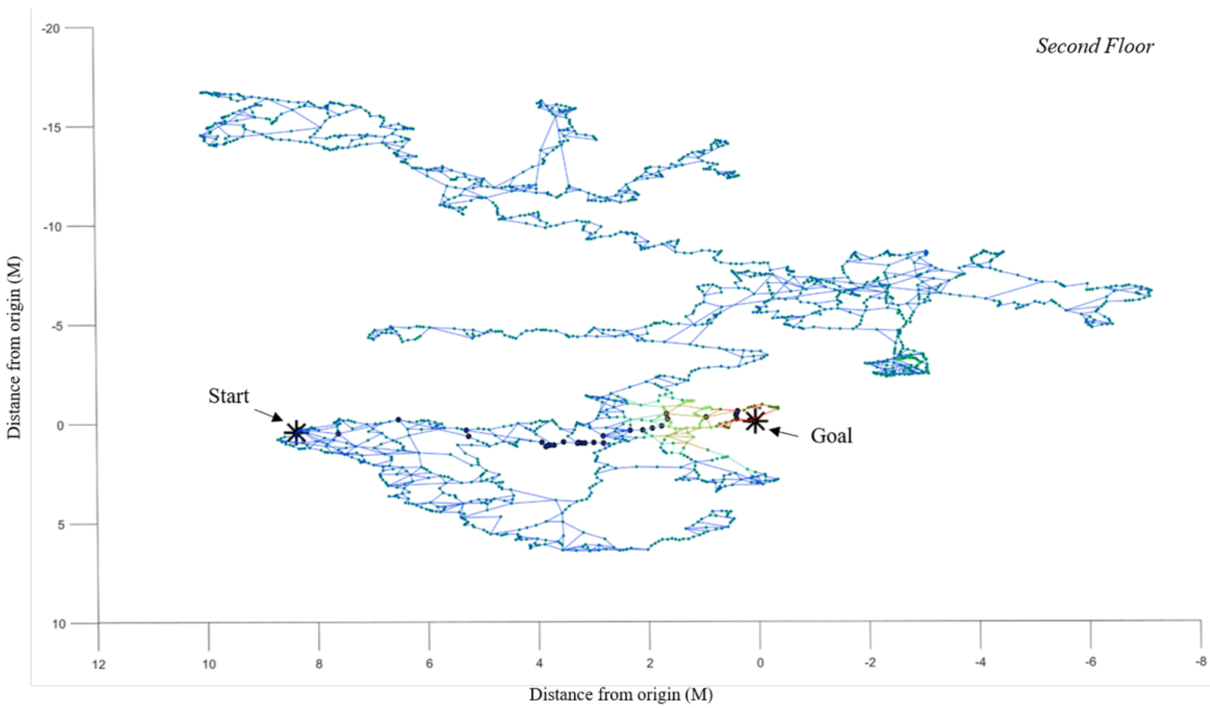


Figure 5 - The 2nd floor of the original Network Building map has been isolated and used here for route planning. Low value nodes, far from the goal given to Pepper, are represented in blue. Those with higher values, representing the goal, are highlighted in red.

This method of goal-directed path planning was intended as a way to provide Pepper with a quick, real-time method of calculating a route through its current environment. The expectation was that Pepper would not only be able to calculate initial routes, but also adapt to dynamic obstacles without relying on fixed positional information for those obstacles. The algorithm performed well in both simulation and physical user tests and was successfully integrated into the overall architecture allowing the route planning to be triggered by users. Through the Android interface (see Chapter 6), users were able to trigger the route planning by requesting an object or by telling the robot to set a specific position as a goal, allowing the robot to react quickly to user-given goals and fulfil requests.

One limitation discovered during testing was related to the distance of the desired route. Over the course of development, the algorithm was tested in multiple environments of varying sizes. While the algorithm was capable of providing a route in the majority of experiments, regardless of the distance requested, there were several factors that were found to affect how efficiently the route was calculated. The distance itself was not necessarily the main contributing factor for this, but the number of location points present along the route within the trained map.

While not developed within the current architecture, the potential solution for this in future work could be limiting the amount of data included within the search by removing irrelevant rooms/environments from the initial calculation. For example, a scenario of exploring a multi-storey building can be used. If a human were to move between two rooms on the ground floor, no matter the location of the rooms, the second floor is unimportant when calculating the route as it has no impact on the ground floor. However, in the current navigation module, the second floor would still be considered initially, even if only to apply negative values telling the robot not to explore the area. The delay this method causes could be reduced by adding an additional calculation of relevant rooms before the route calculation is performed. This would have the advantage of not only providing a quick, high-level way of forming spatial relationships between rooms but would reduce the overall amount of data used for calculating the paths, therefore speeding up the algorithm overall and helping to better meet the real-time reactions expected in the architecture. Additionally, the relationships formed between rooms and confirmed during the execution of the route could then be stored by the episodic memory, to provide quick accessible memories of routes in the future, further reducing the time needed for calculations.

4.5 Building Knowledge Through Exploration

In previous sections, the data used for training the GNG model had been collected across two floors in a single exploration session. This created a continuous stream of data which can be associated with a single origin point (located on the 2nd floor) for the robot and therefore could be used without any pre-processing of the X, Y, Z positional data other than to extract it from the full data file. Prior to practical experiments, simulated data was also collected in a similar way, with all training data produced during a one-off collection. However, it is reasonable to assume that in a real-world scenario, the robot will not always be continuously tracking its movements without interruption. Whether through user actions preventing the mapping from completing or simply by shutting down the robot, the original origin point will be lost, and mapping cannot continue from a saved point. If there is only an initial collection, and therefore only a single map created, at the time of initial setup it limits the architecture to information learnt at a single moment, reducing it to relying on the environment remaining the same for as long as the robot is present and making it unsuitable for continued use in dynamic areas.

To expand the original map and to create a system that could continuously grow over time using multiple data collections taken at different times, Pepper was returned to the Network Building for additional recording on the 3rd floor. This recording took place several weeks after the initial collection and on a floor not visited in the initial experiment.

In the section above the results are shown for the data collected from the second and fourth floors of the university network building. Figure 6 then shows the results of an exploration done on the third floor of the same building. This new data was collected separately and on different days than the first meaning a different origin point and coordinate frame that need to be aligned to the existing one to incorporate the new information. The data collection for this later exploration followed the same method as in previous mapping events. Creating larger maps this way, as opposed to one continuous data collection, proved beneficial in multiple ways. First, it provides a way to allow users to do multiple, small explorations in a time that is most suitable for the, which can be beneficial for those with mobility issues. Additionally, the time taken to model the data collected was significantly reduced. As an additional experiment, the data sets for all three floors were combined and trained as a full set, and confirmed a significantly longer delay period for training than when floors were trained separately and then combined.

To help align the new data with the old, common landmarks are extracted from both sets of data in order to help identify where common access points could be. For this data, the common landmark within all sets was the lift that runs between floors (Seen in fig 6). The coordinates of the landmark on both sets are compared to each other and the difference between the positions is once again calculated using vector translation method.

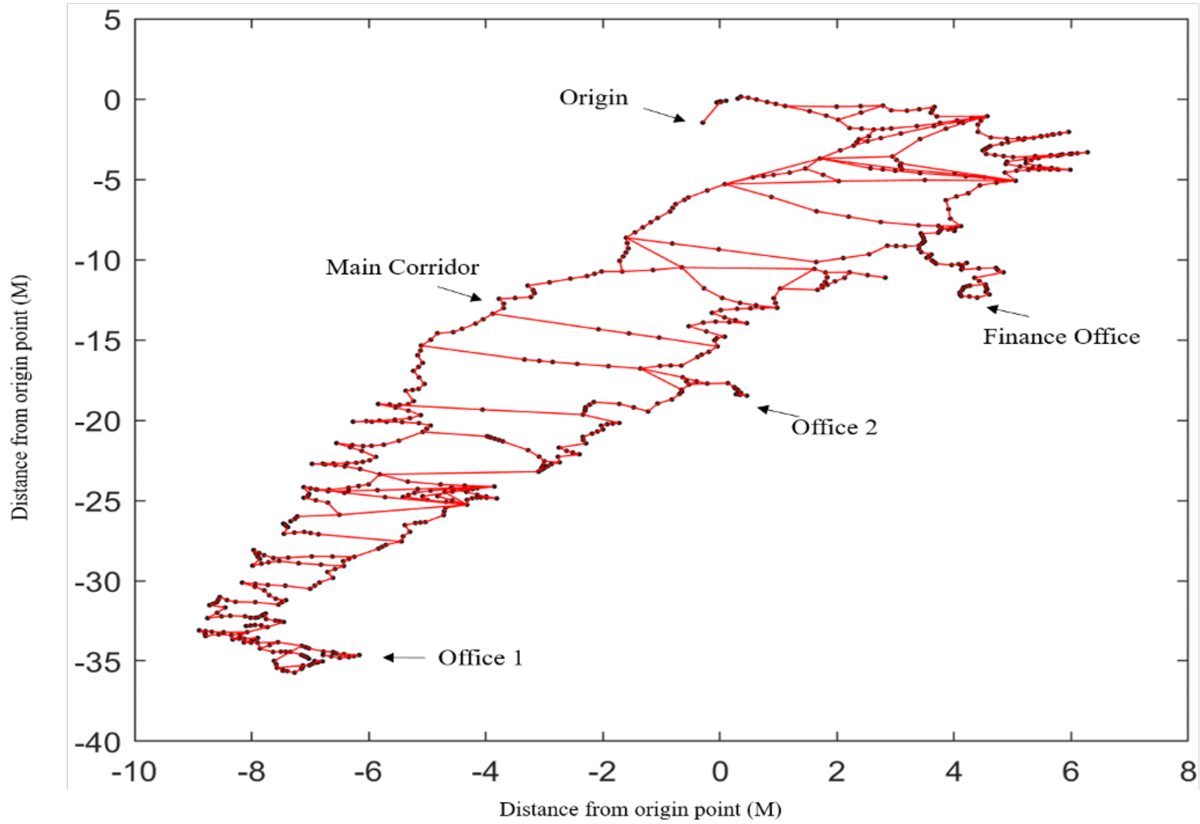


Figure 6 - The third floor was explored at a later time than the original map. The data was later transformed to match the data range of the original map and combined to expand the original model.

Using the difference calculated between the location of the landmark on each floor, the difference can be incrementally applied to every point in the third-floor map, which adjusts the positions in the newly trained map to align with those in the original and allows it to be easily inserted into the original map.

Collecting data at different times, as opposed to an all-in-one session, has several advantages. Firstly, the robot is not limited to the map originally created during the initial exploration. This allows for sections to be updated to account for new changes and for the map to continuously grow as Pepper collects more information

Secondly, depending on the size of the data that is collected, it may be inefficient to train large amounts of data at once due to the time it takes for the initial training to complete. Additionally, this training time would only increase every time data was included if it was required to train the whole network each time new data was collected in order to connect new nodes Therefore it is much more efficient to separate data into smaller sets, to increase the speed which the model can be built and to add to the original map without disturbing data already trained.

This also creates the possibility for future work of creating a central map that can be contributed to by multiple robots. By having several robots collecting data at different locations, an area can be mapped faster and the data can be combined for all to use without having to move the same robot to multiple locations. It also allows a new robot to benefit from the knowledge gained from robots that had been in the area previously, helping to save on setup time and time taken to gather information about a new environment.

4.6 Testing in a local environment

During the project's development, a collaboration with Provide CIC was established to enable testing to be completed in an external 'real world' environment. In the scenario given by Provide, the main role of Pepper within this environment would be to act as a form of receptionist, providing information to patients, managing check-ins for appointments, navigating in a dynamic social environment, and providing directions to users regarding office/room locations. Pepper would ultimately be expected to do this autonomously, without human intervention, and in a safe manner so as to not disturb patients.

The area selected for the experiment was Provide's new GP surgery floor, with a specific focus on the patient waiting areas and corridors leading to GP offices. The goal was to have Pepper map the area, collect data during exploration and then train and build a model to represent the internal model of the area, that could be used as part of the 'receptionist' role at a later time.

To begin mapping, Pepper was placed at the entrance of the waiting room, near the doors leading to the main external corridor, where patients would normally enter and leave the area throughout the normal day. This is where an origin point was set and where any detected landmarks would be related to. As in the previous experiments, human-guided navigation was used to collect the data, which involved pushing the robot through several areas including waiting room areas, offices, utility rooms and medical labs. As in previous experiments, the object detection was run simultaneously to identify notable landmarks tracked as the exploration progressed.

Figure 7 - Data was collected from a new hospital environment, which consisted of multiple obstacles and rooms for the robot to observe. Corridors, doctor's surgeries and utility rooms were explored during data collection.

The data collected was stored in the standard CSV format and saved for processing. This exploration data was then processed in the same way as the original experiment, in order to create a separate network that can now be used for localizing and navigating should the robot return to the area.

The environment was an opportunity to test our model in a fast-changing social environment, and to test the levels of difficulty involved when Pepper had to collect data within a changing, busy environment. Overall, the navigation module appeared to perform as expected and encountered no noticeable challenges from being used in a completely new environment with many unfamiliar obstacles. There were several members of staff and patients within the waiting room area, neither of which hindered the data collection in any meaningful way. The data was successfully used to train the model (Fig. 8) while landmarks were also recorded and trained as per previous experiments within the university. It is however important to note that testing in this type of environment highlighted potential additional components for future work. An additional tag for the navigation CSV data during collection (in addition to room name, obstacles, positional coordinates etc.) would be to note the type of environment currently being explored, such as whether it should be accessible for general route planning or whether access should be somewhat restricted such as for doctor's personal offices or examination rooms. Returning to Provide in future work would provide additional evidence of Pepper's ability to perform route planning while reacting in real-time to unpredicted events as this would require Pepper to use the navigation model to react quickly to unexpected situations while still moving in a way considered safe for patients and employees.

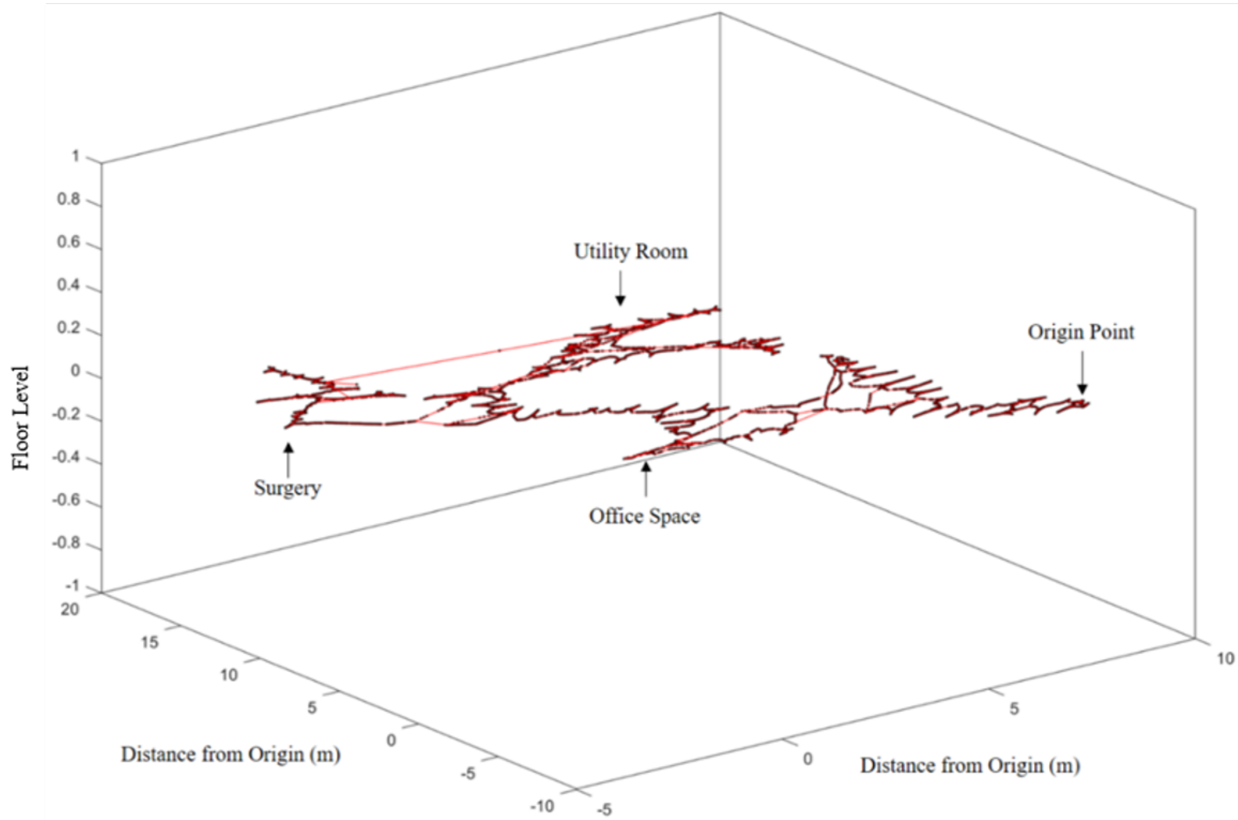


Figure 8 - Following the method described earlier in the chapter regarding the GNG model, the data collected from the Provide Medical Centre was used to train an initial map of the explored environment. The map was able to record from multiple rooms and detect objects simultaneously. The experiment also provided useful recommendations from medical staff as to how Pepper would be expected to react to both patients and staff, as well as the chance to gather verbal opinions from people in the area at the time.

In future work, Provide would also be able to test the social intelligence of the architecture. Interacting with multiple patients over the course of a day, along with regular doctors and nurses, would create an interesting challenge for the robot when distinguishing whom it can answer and how much information it should provide to users. The work style would be like that in a care home, in that there is a distinct difference in the role of residents and staff members, except that in this scenario the robot has no specific long-term users other than staff.

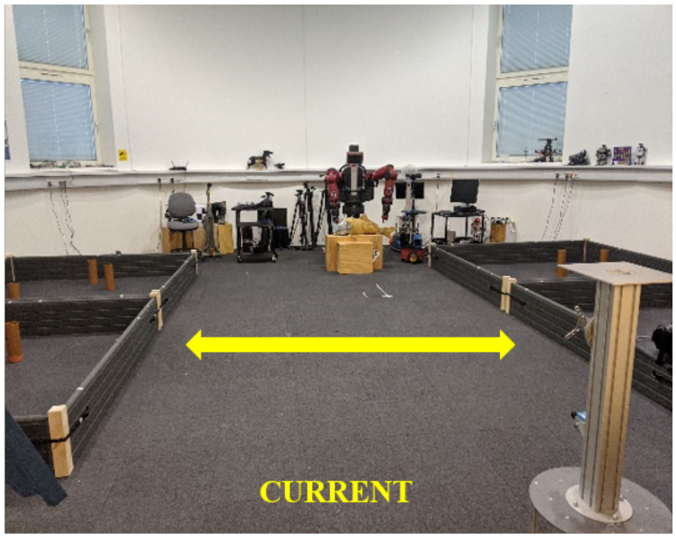
A long-term study to see how the robot develops behaviours and personalises its responses appropriately in an environment where it has a smaller physical environment, but more shortterm users, compared to somewhere like a care home or user's personal home could be an interesting opportunity to study the differences that appear in the structure of the memory system and, in particular, how the robot determines the most appropriate actions to take.

4.7 Cognitive Dissonance in Familiar Environments

One of the main challenges faced when using the previously trained maps for route planning was adapting to changes in the environment between when the robot originally visited and when it returned. The multimodal nodes used in the map training, provide the robot with a prediction of what objects or landmarks can be found in certain locations and any planned routes for navigation will be based on this information in order to reach a goal. However, if the main elements of the room are changed, such as particular landmarks or changing accessible areas, the robot experiences a dissociation between what it expected to find, and what is actually present in the room.

One of the first areas visited by Pepper was the robotics arena within the University. The arena is a very popular area for testing and student classes, and it can be expected that the layout and structure of the room will change often depending on the experiments being performed. This was the case when we returned Pepper to the area, several months after the original training, and attempted to run the route planning functions.

Figure 9 shows the original layout of the arena compared to the picture taken on the latest visit. New barriers can be seen in the middle of the arena, making the accessible areas for the robot much smaller. In addition, there are also multiple objects that were detected in the original exploration that are now no longer present, affecting any goals based on reaching those landmarks, as well as several that were not present before, now creating an obstruction in the robot's planned routes.



(a)

(b)

Figure 9 - Returning to the robotics arena after a period of time confirmed the need to provide Pepper with a way to detect changes within a known environment. The significant reduction of the space available for navigating prevented Pepper from utilising the maps generated using the previous training data. By retraining the map (As an individual room as opposed to reconstructing the full map data which consisted of multiple floors), the network was able to recreate the new layout and integrate it back into the system without major disruption to the full map.

The discovery that a known environment has changed, causes the system to identify and trigger the need for an update to the model to account for new layout, obstacles and landmarks. The new mapping data is collected in the same way as the original, by guiding the robot around the arena and allowing it to record sensor data for coordinate locations and object detection. The data is then retrained using the GNG model to create a new map model for the arena. The advantage of training rooms/floors separately from each other (as discussed in Chapter 4.5) is that the new model of the arena can be inserted back into the main map without disrupting the larger map and with low training times. As the room is trained separately, the training time is based purely on the new data being trained as opposed to retraining the full map, which would take significantly longer depending on the number of areas mapped. This proved to be a particularly efficient way of updating the robot's internal model of the environment. While similar to the method used when adding additional maps to an existing model, retraining and replacing rooms appeared to be a simpler process for the navigation model to complete, as positional data for re-joining the maps was much more readily available such as the current direction/position of the robot, being able to extract starting points from the existing model and having the previous model of the environment as comparison for major remaining landmarks.

4.8 Human-Aware Motion Planning

One of the biggest problems during navigation was not due to the robot but to the inference of other obstacles in the environment. The experiments completed throughout this thesis were all performed in areas that are frequently used and interacted with by humans. This creates a significant problem when attempting to use the maps to perform route planning, as there is no guarantee objects/landmarks have not changed and that extra obstacles will not appear while the robot is navigating. During mapping, static obstacles are automatically accounted for as Pepper never explores through them, therefore the space they occupy is empty within the models and not used during path planning. However, for obstacles that are dynamic and unpredictable, there will be no record in the model, and they cannot be planned for in advance. Therefore, the robot needs a way to react to these changes in real time, without additional human help and continue on towards its goal.

Moving away from an obstacle also presents certain challenges. The robot must consider surrounding obstacles that it does know, e.g., walls, so as not to collide with something while attempting to avoid others. To account for this, two types of experiments have been completed. In the first, a script was created that worked alongside the Python-based Pepper NAOqi libraries to detect and calculate the position of humans in the field of view. When detecting humans , Pepper will attempt to track their trajectory by calculating the difference in their position compared to the robot over time. By calculating which way the person is moving Pepper knows a safe direction to move away from the collision, and can determine the likelihood of the collision happening to begin with. In terms of its ability to avoid obstacles, this method exceeded initial expectations of not only how quickly the robot was able to detect and avoid moving objects but also the speed at which it was able to return to its original path once the obstacle was no longer present. Using sensor readings, Pepper was able to temporally store the movements taken to avoid the obstacle and perform them in reverse order to return to its original position on the route. However, due to being implemented on the older (python-based) Pepper robot, several issues were detected when using this method as part of larger environments and experiments. Issues with Wi-Fi, older ROS systems and transmitting data between the robot and external computers meant that if a completely new path were required (such as if the obstacle stopped in the path as opposed to passing by), the robot would experience significant delays while attempting to call the navigation model for a path calculation, while at the same time being blocked from responding to additional user input. It was therefore decided that in later models of the navigation module, different obstacle avoidance systems should be considered, particularly for unpredictable dynamic obstacles.

In a second experiment, a simulated home environment was used to simulate an environment where it could be reasonably assumed that humans would be moving. The robot in the simulation is provided with a start and goal location which it uses to calculate the initial reward field as shown earlier and the robot begins moving to its goal based on the closest most valuable neighbour. However, in this case, for each step the robot takes, another calculation is performed to calculate the distance to the other "human" agent in the simulation. The robot will not change course unless a collision is detected, which is when the "human" comes within a specific distance.

When the other agent is detected as being within that distance, it is temporarily assigned as the goal of the robot (as described in Goal-directed planning). The newly created reward fields have a -1 multiplier effect on the area surrounding the other agent, effectively making them the least rewarding nodes on the map. This creates a zone around the agent that is now unappealing to the robot, and will force it to calculate a new path based on the new reward values.

This method works effectively with one or more agents and will cause the robot to either pause until it has a valuable node neighbour or to change trajectory in an attempt to find a more suitable path. While experimenting with this method on the physical robot, the algorithm did not appear to have any major restrictions when compared to running in simulation. When encountering unexpected obstacles, Pepper was able to successfully trigger the model, through the architecture, to create a new path with new routes being generated in the majority of scenarios. However, the experiments on the physical robot did highlight some limitations in the model when compared with traditional SLAM methods. Maps generated on data collected through the robots' exploration phase presented a much more challenging environment to calculate routes in than when first tested through simulation. Keeping track of the robot's position in space, its distance travelled and recording observations of the environment were easily managed by the developed software and maps, however, it was observed that changes in the robot's rotational position greatly reduced its ability to recalculate useable routes through an environment. Routes returned by the model were technically correct based on the algorithm but the slight error in positional data that had been collected quickly caused the robot to misjudge the location of obstacles and available pathways. This grew more pronounced the further the robot travelled and the more obstacles it encountered.

From these experiments, it can be concluded that in this particular novel method of human avoidance there is potential for providing a new, quick method of dynamic obstacle avoidance. In simulated experiments, the algorithm performed as expected and even provided additional benefits such as being able to quickly scale to account for multiple dynamic obstacles. However, for future work, it would be necessary to provide additional functionality to better track the rotational position of the robot when performing physical experiments to allow for continuous obstacle avoidance and ensure the robot is accurately reporting its position in space. This conclusion is based on the observed behaviour in which the robot would often successfully plan routes around the obstacles but would misjudge the direction it was facing upon returning to the goal path. Common causes of this error were noted as being flooring issues on recalculated paths (such as dividers between rooms) and incorrect positioning of the robot when originally returning to a mapped environment. The latter could potentially be solved in future work by having the robot focus on and face key, static landmarks when initially mapping and re-entering an environment.

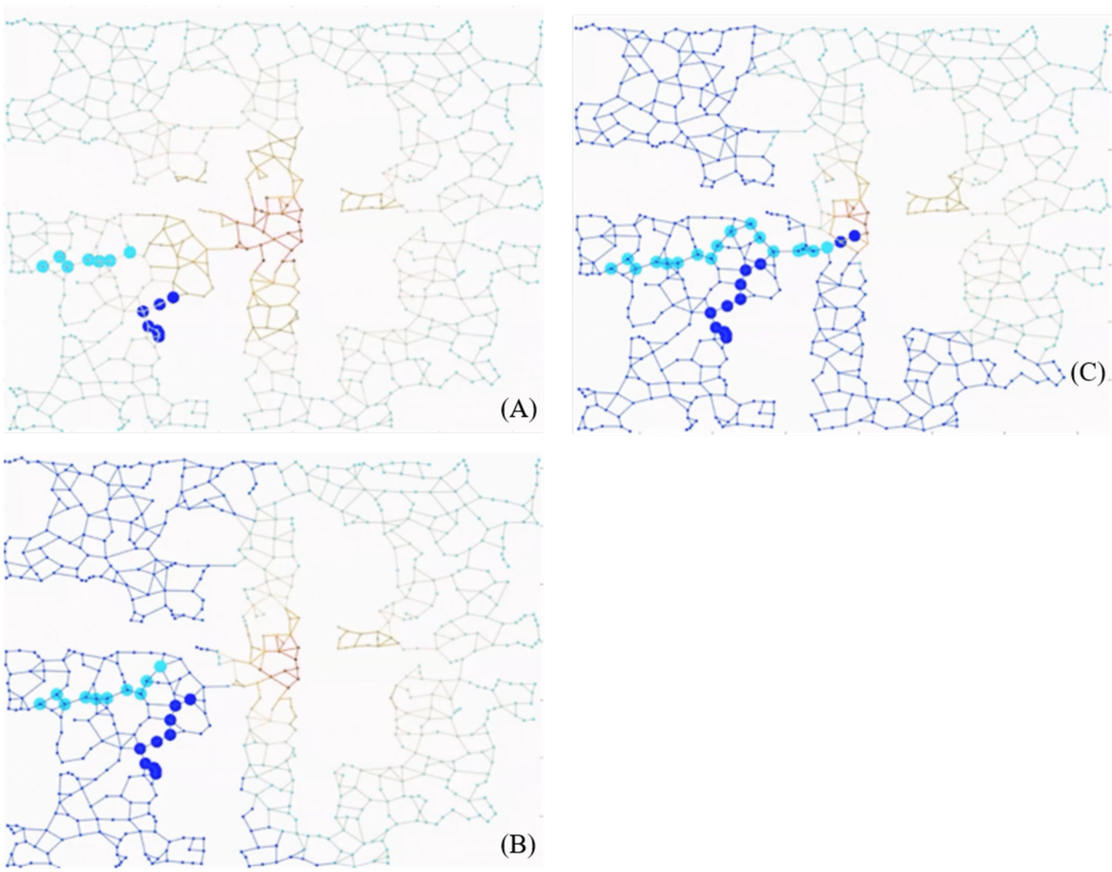


Figure 10 - A simulated environment was used to verify the ability of the navigation to account for unexpected humans appearing across the route. The architecture has been adjusted to provide priority towards any oncoming dynamic obstacles to prevent collisions with humans. The safety of users was a key concern in multiple research studies due to the size of social robots.

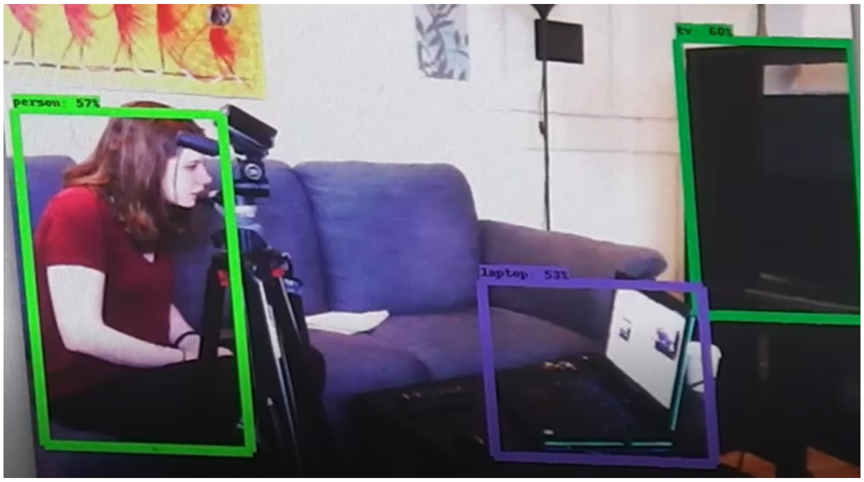


Figure 11 - While monitoring the environment or completing a request, humans presented significant challenges for the robot in terms of quickly adapting to the new environment. Objects that appear unexpectantly are detected using the TFLite object detection. Humans are also detected through the object detection, but it is Pepper's inbuilt sensors that initially create a warning for the architecture to halt navigation. This creates an immediate safety stop for user safety and provides time for the object detection and facial recognition to react and confirm the presence of humans. Navigation routes that were being followed can either be continued if the obstacle disappears after some time, or recalculated through a call to the Observer to trigger route-planning to restart from the current position.

5 Towards Growing Episodic Memory and User-Centric Assistance

5.1 A biologically inspired memory network

The second major system included in the cognitive architecture was the growing episodic memory. To create a companion that was able to adapt to unpredictable situations and diverse user needs in real-time, it was necessary to implement a memory system that would be able to cumulatively grow and learn as the robot gained experience with users, almost mimicking the way humans continuously learn from their own personal experiences with others and their environment.

By integrating this system into the main architecture, it provides the Pepper robot with the ability to tailor actions and responses to specific users as well as quickly adapt to current scenarios, based on the knowledge it had stored from its own previous encounters. This helps to create a personalised companion that will be able to help users with daily tasks and anticipate their needs autonomously without specific human guidance or fixed instructions.

The memory system has been inspired by the biological process of episodic memory in humans, a form of autobiographical memory that allows the brain to retain information based on personal experiences and recall said knowledge when presented with similar scenarios in the future, that are similar to the original experience.

This particular system was based on work previously completed for the EU Darwin architecture as shown in [103] and has been redeveloped to include significant additions and changes compared to the original structure and purpose of the system. Notably, the system was redesigned to have a specific focus on creating a personalised experience for users, the inclusion of new functionality developed specifically for this project and alterations to the core structure of the original code to create new functionality for how memories are assessed and extracted from the memory in order to anticipate user behaviours, consider environmental and social context and form potential goals.

The remainder of this chapter will provide a detailed explanation of the overall memory architecture, including how memories are formed, stored and recalled to predict appropriate actions for the robot to perform and complete user-given requests. It also explains the new additions to the original Darwin memory such as the introduction of user-specific memories, a new method for recalling memories based on temporal order, and a context-specific method of rating memories when compared for extraction. An explanation of the investigation completed to explore how the episodic memory framework can help to understand social context from the perspective of the robot is also detailed in the chapter. A series of experiments are presented that show its integration within the architecture and the consequent results.

5.2 Training a Memory Network

To create a form of autobiographical memory within the architecture, episodic memory has been represented through an auto-associative neural network (AANN). AANNs are a type of network often used for pattern-matching purposes as they are designed to recall full sequences of data from only partial input. In addition, the data presented as an input does not have to match the trained data exactly, as the network is able to connect similar scenarios through associations in the network. The network is trained using a one-shot Hebbian learning method, providing a quick learning method for adding new experiences. As previously mentioned, the initial model for the memory was based on the EU Darwin Project which originally implemented the memory model onto an I-Cub robot [103]. As part of my own prior work on this topic, it was also implemented onto a Baxter robot as part of a master's level postgraduate degree which provided a level of confidence that modifications to the base model could be completed to also suit Pepper and the purposes of this thesis.

To encode experiences into the network, the architecture is designed to replicate the biological functions of episodic memory in that it will attempt to continuously encode and recall information to form connections between situations. Data comprised of the objects present, the actions to be performed on them and any recognised users, is processed using a series of 'hub' networks. These hub networks are designed to work as individual SOM networks, an explanation of which can be found in [104]. The hubs represent individual types of elements that the memory is capable of encoding (the objects, actions, users etc.), with each hub size being directly related to the number of spaces reserved for that type of element in each memory. For example, when configured to store up to five individual users (N = 5), the 'person' hub becomes an N x 1000 matrix for weight training.

A main advantage of this hub-based structure is the ability to have bottom-up activation based on visual cues collected from the environment. The activation of these hubs then directly provides the context needed to build the partial cue, required for full memory retrieval. Not only do the hubs identify recognised objects, aiding in differentiating between objects that have previously been used for experiences and novel objects, but they also provide the temporal order that the elements were originally trained in as part of experiences. This provides further context to the system as to what order sequences should be combined and replayed.

5.3 Basic Structure of a Memory

For each new experience that is encoded into memory, the experience is stored as a vector of 1000 neurons (N = 1000) which can then be unfolded to create a structure made of 20 rows and

50 columns. This 20x50 matrix is an alternative representation of the memory used to store both the elements and their identifying tags, and the temporal order of the memory sequence.

To represent the memory in terms of order and time, each row (1-20) represents the order in which elements of the sequence were learnt and the columns (1-50) represent the element within the sequence. Each element within the memory is identified by a 'tag', found in columns 42-50, which labels the element as belonging to a certain type (object, action, user ID etc.) An example of this can be seen in Fig 12. where a memory has been extracted from the network containing the contained elements and their identifying row tags. When recognised visually, all types of input (users, objects, actions etc.) are initially stored in vectors, which in turn feed into the main systems. The vector sizes depend on the space allocated to them within a memory, which is currently 1-10 for actions, 1-5 for users and 1-42 for object vectors.

An example of a fully extracted memory, in the correct temporal order, is shown in in Fig 12.

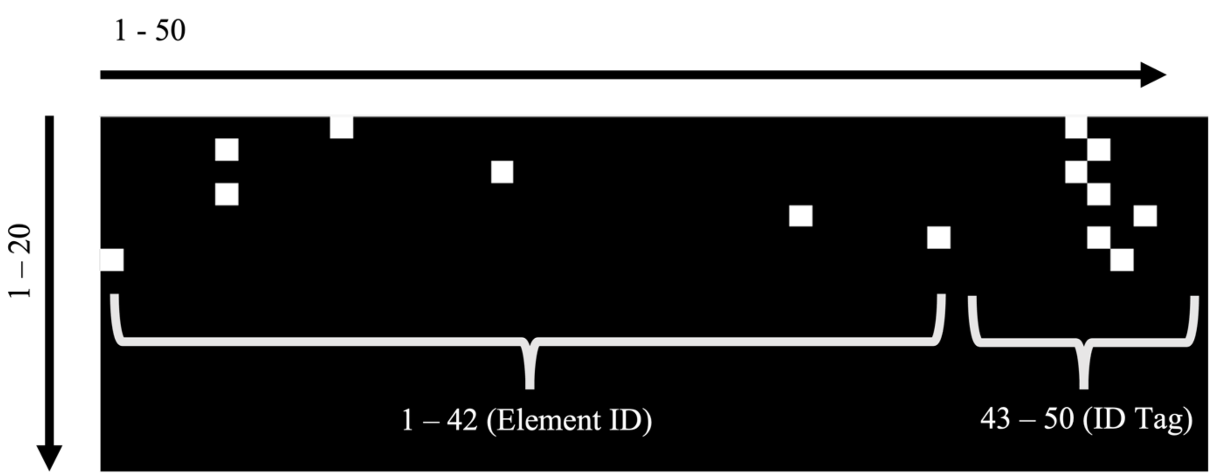


Figure 12 - An encoded memory will follow the same structure for encoding and retrieval, containing a specific order of elements according to the time-based order they were performed in. The 20x50 matrix pictured here is a general example of the structure with the left section of elements representing specific IDs for objects, action and people while the opposite right-handed section contains the identifying tags interpreted by the episodic memory to specify the type of element that is being encoded.

5.4 Recalling Previous Experience using Partial Cues

Recollection of past experience is a key feature of biological episodic memory. It is the recall in particular, that allows the brain to imagine potential consequences of current actions as well as form plans based on previous similar scenarios. In particular, the concept of 'mental time travel' in biology introduced by Tulving [105], [106], proposes the ability of humans to project themselves mentally into the past to replay former actions (and recall the consequences) as well as picture future scenarios and anticipate outcomes. This ability, which Tulving argues is unique to humans, provides an important ability to consider actions, both past and present, using the same biological functions as if they were being performed.

Within the system, past experience is recalled through the use of partial cues, constructed from elements of the immediate environment - such as objects seen, people present or locations – and/or commands given by the user.

Memories are recalled for two reasons within the architecture; First for retrieving an action plan for the robot to use in order to complete a goal, and second in a passive sense to predict what behaviour the user may be engaging in.

To build the cue, the information is collected through the Android interface and passed to the Observer module, a server running externally from the robot that is responsible for processing incoming information and making decisions for how to act. It is here that the information can be processed into a cue and later sent to the episodic memory. Here the cues are structured and stored in individual vectors representing particular types and tagged according to their purpose (object, action, user etc.).

Upon entering the memory, the cues are compared to the hub networks (See Chapter 5.2). Through bottom-up activation of the hubs', individual recognised elements are identified (or noted as unknown, novel elements for learning) and compared to the full collection of previously learnt experiences as part of the extraction process.

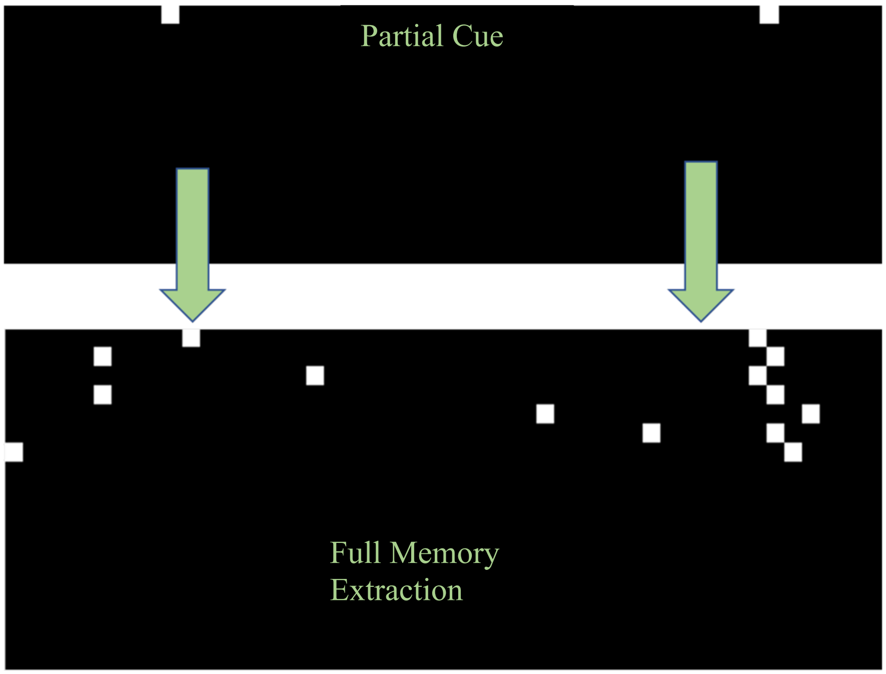


Figure 13 - In the example above, the top image show a basic partial cue developed from a signle identified object. Through the observer, the recognised object is converted into it numerical ID tag (opposed to the string sent from Android) and has been passed to to the smaller hub network. The bottom up activation creates a trigger within the 'object' hub, resulting in the element receiving the identifying 'object' tag. This cue is then used for extraction, where it is compared to the full collection of memories. Based on context and recognised elements, the winning memory is extracted and sent to the Observer for processing (bottom image)

One of the major changes made to the architecture during this project can be seen here and involved changing the way the weights are used when trying to recall memories. Previously, when a cue was built, it was directly compared to existing memories in order to detect the correct temporal order of the experience and therefore the order that the cue would be used.

For example, if the desired goal was to fetch an item and return it to the user, the order that which these actions are performed is specific to the goal. Place them backwards and Pepper receives conflicting instructions that cannot succeed. However, by directly comparing using the weight hubs themselves, it is possible to extract the position of the cue in previous memories based purely on the location of the element in its equivalent hub. This disconnects the system from directly using all stored memories and prevents an unnecessary loop that involves more systems than what was required and shortens both the time and processing required to extract a full memory.

Once a comparison to the experiences has been completed, the system attempts to reconstruct an action plan from related memories that the robot can implement in order to complete the goal. This can be done in multiple ways. In some situations, the memory and therefore the action sequence may already be known in which case the full memory can be extracted and used. In others, a larger memory can be broken apart to extract only the relevant information to form a smaller action plan relevant to the wanted behaviour, or elements may be combined across several memories to try and make a suitable plan. If an individual user is being monitored or the robot is looking for personalised behaviours, plans may be rejected if they are inappropriate for the user or alternatively can be extracted from other users' behaviour to be tested on the current user. The user then has the option to reject this plan (furthering the personalised behaviours). Finally, the objects may be unrecognised in which case the memory will attempt to bind the new objects to the most appropriate action based on former plans.

5.5 Favouring environmental context over manual rewards

So far, throughout previous versions of the memory, the reward structure and values given to each memory have been the main way that 'winning' memories are compared and extracted. This is achieved by providing each experience recorded with an assigned user-given reward value indicating its potential usefulness. In the case that multiple experiences had been activated in memory using the same cue, the rewards are used in order to distinguish the most relevant action sequences, with higher rewards increasing the chance of the memory being selected and lower-value memories being ignored. However, allowing users to provide rewards is not always the most accurate way of deciding the relevance to a situation. What is important to one person, is irrelevant to another. Similarly, one person's idea of a good solution can be the opposite of another's. User bias, multiple users and conflicting rewards for similar memories can therefore all affect the final decision of which actions are chosen in a given scenario. It also creates a reliance on a manual user to assign value to the agent's experiences each time it records a new experience, which conflicts with the autonomy this research is trying to provide.

As a potential solution, an alternative system was developed that separated the reward from the initial experience at the moment it was encoded. Rather than relying on the user to provide a value and experiencing the problem of finding a user to do so, the reward is instead automatically generated for each relevant experience triggered by the given cue during the recall process. Rather than include predefined rewards in the system, the memories are instead judged by how relevant they are given the information provided by the partial cues and available in the environment.

This is done by comparing what the memory holds, to what is available from the point of view of the robot. For example, when provided with two memories with the same reward value, previously the system relies on trying to find overlaps within them and attempts to consider both. Instead, it is proposed that the content is more important. If a memory has three of the objects present compared to another's 2, there is no point in considering a predetermined reward, the second is the most relevant given the context.

5.6 Using Personalized Memory to Prevent Dangerous Behaviours

Up until this stage of development, memories have been triggered and recalled based mostly on the objects detected in the environment. While the experiences stored integrated more information from people, actions and objects identified by the robot, it was the objects that heavily influenced the order of retrieval. The number of objects, identity and the actions performed on them, formed a large part of the partial cue compared to social tags (the people recognised) and memories were recalled based on how accurately they compared to the objects detected in them. However, a key focus of this work was personalisation, and rather than simply remembering who had taught the robot as in previous experience, experiments were started to test the possibility of bringing more of a focus on not only environmental context but social context as well.

This was a key focus of the experiment performed below. It is clear from the interactions we have in daily life, our reactions differ based on the people we are with. Speaking to a known friend is going to influence how we respond compared to if we are speaking to a stranger. Similarly, the better we know someone, the easier it is to recognise when an action they take is contradictory to their usual behaviour.

Not only this, but as mentioned in the RAMCIP deliverable [19], one of the key additions to a companion robot would be the ability to intercept harmful actions taken by the user, or those detected in the immediate environment by the robot. It was stated that as an immediate response, the robot would ideally prevent them from happening where possible. This helps to ensure the safety of the user when alone with the robot and provides a way for Pepper to directly aid the user in real time should such actions be detected.

When considering the context of individual home use and care environments, and the users most likely to interact with Pepper in such environments, the type of harmful action expected could range from those presented by the environment (e.g., obstacles in the way of the patients), to actions being taken directly by the user (e.g. taking the wrong medication).

Therefore, it was necessary to implement this recognition behaviour in Pepper. To do so, an addition was made to the memory network that creates functionality for comparing the current situation to previous experience that the robot has gained specifically with that user.

This aspect of the memory is specifically for monitoring purposes rather than general action planning but retains the functionality of episodic memory by using past experience with a specific person to compare the activities. An experiment was performed whereby a recognised user sat in Pepper's field of view and attempted to interact with a particular object. In this situation, the object was a set of tablets – identified by a QR code that the robot was able to read. Under normal conditions, the robot would attempt to aid the user by providing them with an action plan on how to use the object seen. However, with the additional functionality, and acting under a monitoring mode, the robot instead attempts to recall the experience of aiding the specific user with the object. For current experiments, this functionality is placed in a separate monitoring mode as the general behaviour for the companion in daily situations would be to offer aid based on the predicted goals of the user, and if not possible would attempt to use generalised memories from other users. However, in this specific scenario that itself would be a potentially dangerous situation. In this case, the robot failed to recall any experience involving both the user and the object and was alerted that the user was performing an incorrect action. Because of this, Pepper announces the action to the user and instead builds a new cue for the memory based on the actions and user alone (e.g., the user has not reached for this medication before, but has reached for and taken others) and attempts to match the two memories to extract the correct course of action. The result of this experiment was Pepper correctly identifying the lack of record of myself and the tablets (paracetamol) shown and successfully alerted to potential danger. The robot then announced, through speech, why the action may be incorrect and identified the medication it believed to be correct based on previous experience.

While overall successful in this experiment, it is noted that future work would benefit from an addition to the system that would allow the robot to automatically switch between observation of the environment with the intention to actively aid users and this mode which is specifically looking for incorrect, potentially dangerous behaviour.

5.7 Using Encoded Experience to Promote Shared Knowledge between Users

The addition of user recognition to the system was originally intended as a way to provide a method to distinguish between memories associated with specific, individual users. This was important for separating experiences between users to create an association between their IDs and the preferred way of having Pepper respond and seemed a promising way to fulfil the original aim of having personalised behaviours for users. However, while this did appear to be effective when compared to the original goal of the experiment, it also had the unforeseen consequence of preventing the learning of actions and skills between users when introduced into the larger memory module. This then prevented sharing general knowledge learnt from one user to another, making inference of actions from the encoded memories impossible unless they were specifically trained by the same user. This placed a large limitation on the system and reduced a lot of functionality that was considered important to replicate the episodic memory function. While it is important to distinguish actions that may only be suitable for certain users, just as important is recognizing knowledge that can be reused to help the robot adapt to new users in new environments but facing similar scenarios to the ones Pepper has already become accustomed to.

As a solution, the recall process within the memory module was redesigned with a specific objective as motivation. The robot, when possible, should default to a detected user's preferred behaviour. This means that should the ID be known, memories associated with that particular user will gain priority in the system when being considered for extraction through the partial cue comparison.

In this new system, when presented with a user, Pepper will attempt to extract the most relevant information based on the current environmental and social context. In this case, that refers to attempting to remember information learnt from the recognised person first, with the assumption that actions taught by that person would be more acceptable to them than those taught by others. However, in the scenario where the person making a request is now unknown, and therefore does not meet the requirements for placing the ID tag into the partial cue, the memory will default to searching based on general cues it is able to build with available information. In this case, it is likely the memory will return an experience developed between the robot and another user. In this case, as the robot is unknown and if it is not detected as a dangerous action, the recalled memory will be formed into an action plan for the user and the robot will attempt to assist. If the plan is successful but changed (items no longer present, novel items detected and merged with recalled memory etc.) it can then be reassigned to the new user by combining the action plan and the new social cue. Users can be assigned an ID for future detection once they have completed a quick name-face assignment with the robot, in which Pepper will attempt to associate facial features with certain people to recognise them automatically in future. This ID is then tagged with the appropriate label (for recognition in ‘people’ hubs) and merged with the new action plan. This can then be considered a new memory and encoded back into the main network using the normal encoding process.

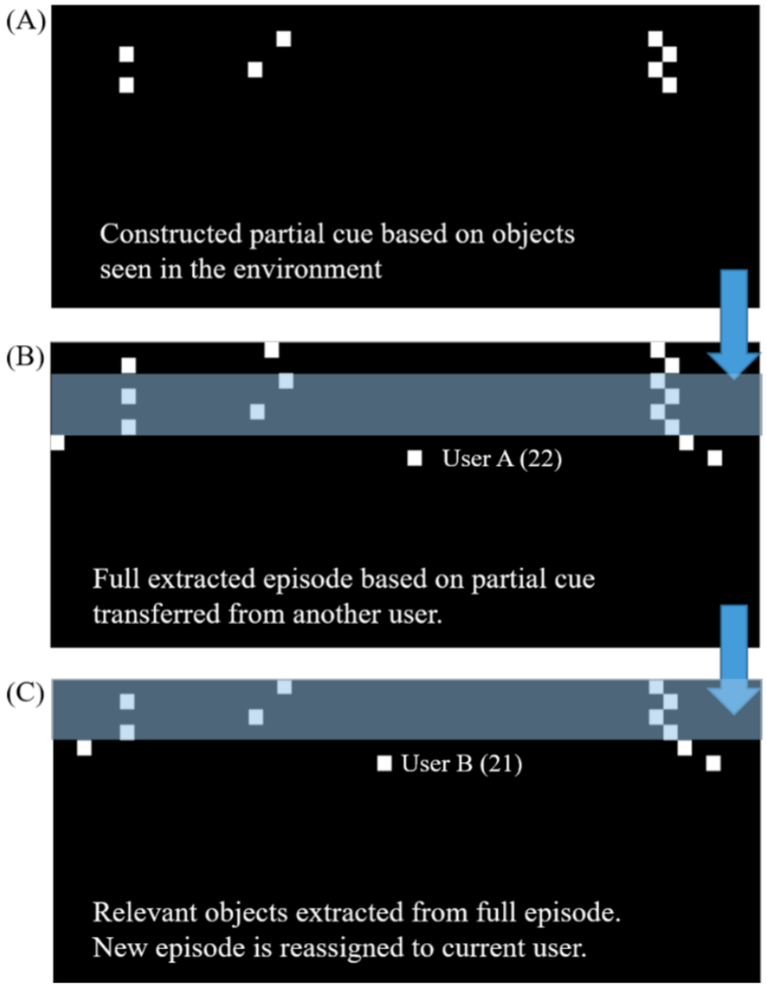


Figure 14 - When extracting general memories for unknown users, the memory will forgo any user-based priority. This allows the extraction of memories based purely on the closest result based on similarity to the current context. The image presented shows the result of the extraction. In figure (B) it can be seen that the memory extracted was actually much larger than what was required in the moment. As a result, it has been automatically shortened into a new shorter sequence (C) including only the relevant objects. As the memory is now considered 'new' by the architecture, the user - after completely facial recognition and entering a name, will be assigned to the new memory during encoding. This allows for personalized behaviour to be specific to the user.

6 Towards a Growing Cognitive Architecture

One of the main objectives for the thesis was to create a software framework that would be domain agnostic, modular and able to link the main functionality of the CAREOBOT project. The architecture needed to be capable of connecting multiple systems (memory, navigation, social etc.) together while also being able to respond to users' requests quickly and in real time. The main requirements of this framework were that it must be able to respond in real-time, it should work across multiple devices, and it should be able to adapt to one or more missing functions. For example, if the companion architecture were to be applied to a device other than a wheeled robot, such as a phone or smart device, modules such as memory and social intelligence should continue to function despite the navigational capability being removed. By designing individual systems in modules, it also ensured that any errors or upgrades to individual systems would be limited to those specific functions, rather than disrupting the functionality of the entire architecture and therefore ensuring that some amount of assistance could still be given.

The remainder of this chapter describes the method used to develop the overall architecture, various experiments and results of testing individual sections of the framework, and the results of the final testing of the architecture.

6.1 The Observer and The Brain

The observer is an independent software module integrated into the companion architecture and created to act as a central point within the framework that connects all other modules (memory, navigation, interfaces, etc.). It is the only module within the architecture that cannot be removed or stopped when applying the system to other platforms or when upgrading modules, as it acts as the connection point for all other services. The observer is responsible for gathering all data taken from user inputs and the external robot sensors and filtering it to the necessary modules to predict the appropriate response the robot should take. The main responsibilities of the observer can include updating the world view based on incoming visual information gathered from the robot (or external devices), creating personalized user-specific reactions from facial recognition, triggering the memory retrieval process, forwarding new behaviours and outcomes to the memory, processing returned navigational directions, triggering the spatial awareness and updating maps with data collected from the robot's exploration.

The observer is a core component of the overall architecture and has been designed to be as adaptable as possible in terms of connecting to different platforms or languages. It does not include platform-specific code, nor does it require other connecting modules to be in a specific language in order to process the information being sent.

Taking further inspiration from biological processes, the design of the observer is partially motivated by the Global Workspace Theory (GWT). The GWT, proposed originally by Barnard Baars [107], [108] as a proposal for how human consciousness works within the brain, presents the theory that while the brain is continuously processing large amounts of information, only a small amount of this is deemed to be relevant and therefore suitable to be brought into awareness. It is this smaller amount of conscious information that is then used for calculating actions and reactions in order to work towards specific decisions and goal-oriented behaviour. Simultaneously, while the rest of the information remains in the background, it is unconsciously being processed while still remaining accessible if it should suddenly become relevant at which point it will be brought into conscious awareness. The theory states that by processing information this way, it allows the brain to manage the large amounts of information it is constantly gathering from the environment, while only focusing on the parts that are relevant and needed in the current context.

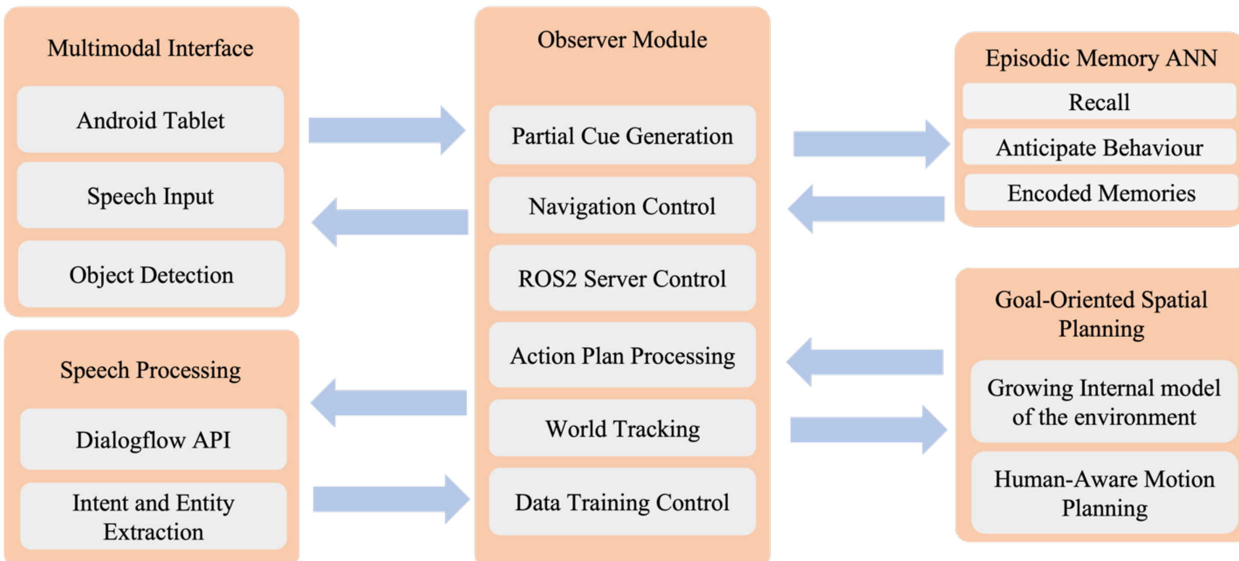


Figure 15 - A graphical diagram of the full architecture is provided. The observer module can be seen here connecting all external models through a series of ROS2 action servers and clients. Data was collected from Pepper, the interface and results of the initial speech processing and sent to the Observer in JSON format to provide an overall summary of all available information. It is the observer's job to filter the information and make a decision on which modules should be contacted. This could be Episodic Memory, Navigation or a status report to the main interface. On return of results from the external module, the observer once again is responsible for determining the order of response to the interface and, by extension, Pepper. It is here that any additional calculations, such as breaking down a navigation route into steps for Pepper, comparing the returned actions to the current world- view (in case of sudden changes) or informing the interface that an action requested cannot be completed.

6.2 Real-Time Communication using ROS2

To create the connections between individual modules, multiple combinations of ROS (Robotic Operating System) were tried and tested over the course of the research. ROS is an open-source software framework, commonly used across multiple robotics projects and platforms and provides multiple libraries for users to modify and adapt into their own projects while still maintaining a common software framework across development and projects.

The motivation to use it within this project came from several places. The development kit provides multiple libraries to enable methods of communication between software, code languages and robotic platforms. The libraries are accessible from multiple operating systems, are able to provide communication between different coding languages and are platform agnostic, meaning they can be used on integrated into robotic platforms without major changes to the main functionality of the libraries. ROS has also been used globally for both individual and large-scale projects, as well as when teaching robotics, which made it an option that not only had wide support in the case of issues during the development of the robotic companion but also made it highly unlikely that the libraries would be suddenly unavailable or inaccessible during the work of this thesis which would have significantly affected the development of the architecture.

The final version of the architecture presented in this thesis uses a combination of ROS2 and basic server/client connections to combine the individual software modules and provide real-time communication between services.

6.3 Creating a Modular Software Framework for a Cognitive Architecture

The overall project is made up of multiple software and hardware elements, a growing memory, a navigation system, a social interaction system, the Pepper humanoid and multiple Python and Java software libraries. A decision was made early in the project to break the components of the planned system into different features and build them separately, creating isolated blocks of code for different functionality. Over time, this resulted in a group of software 'modules', representing the growing episodic memory, human-aware navigation system, social interaction and external devices respectively. There are multiple advantages to creating the architecture this way. Developing the code into these individual software packages as opposed to a single conjoined program, allows the different modules to act independently and are not reliant on the current status of other modules. This means that in the event one module requires upgrading or removal, it can be quickly removed or replaced without negatively impacting the functionality of other modules minimizing the disruption to the overall system. This allows the robot to continue functioning despite one feature being potentially unavailable.

Additionally, the isolated yet combined module functionality once again mimics the concept of the interconnection of processes in the brain. While the modules take advantage of the information provided by one source, they each perform a specific function. The information from each is then used to create a full cohesive plan when returned to the Observer module.

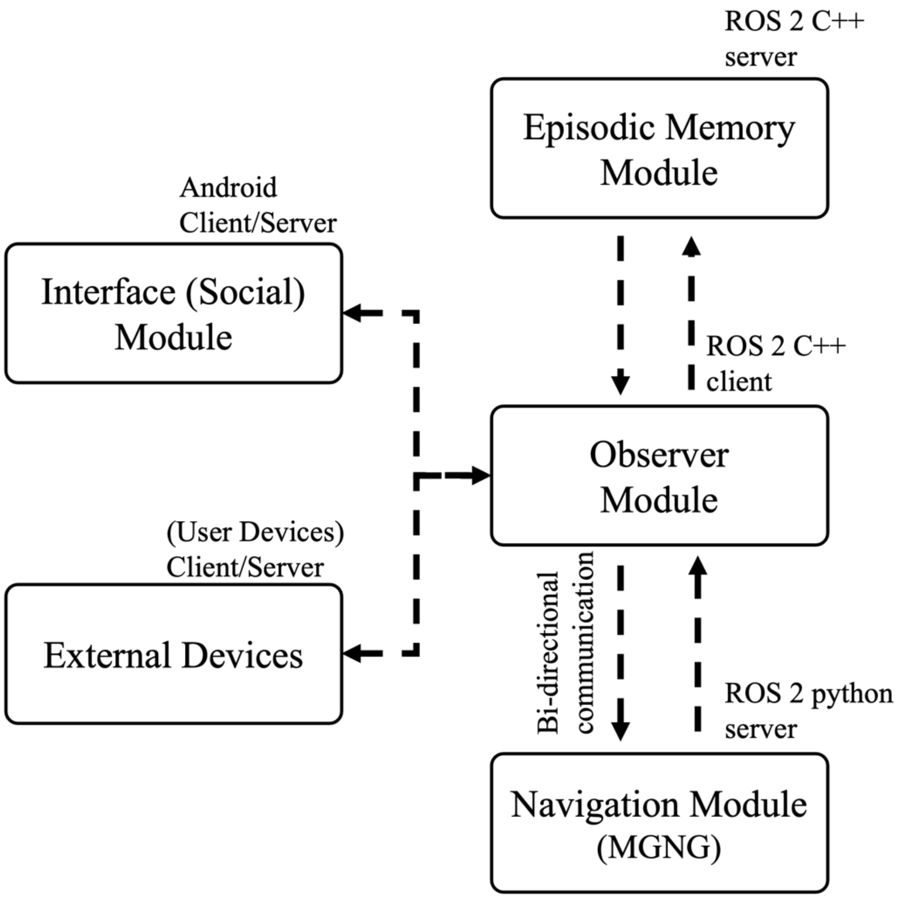


Figure 16 - The software modules are combined into the larger architecture to create an adaptable framework. Except for the Observer module which acts as a central processing module, all others can be removed or replaced without disrupting other modules. The individual modules communicate via server/client connections enabling real-time communication.

6.4 The Importance of a Domain Agnostic System

By creating the software in isolated modules it gives the advantage of a partial embodiment and domain-agnostic system. The largest benefit of this is the possibility to apply the architecture to multiple platforms and devices as it can be quickly adapted to suit the hardware by removing incompatible modules while maintaining some functionality of the main system. For example, if the architecture needed to be placed on a static robot, the navigation module can be easily disconnected by severing the connection to the main observer and minor changes to the code to prevent the attempts to call navigation functions through the interface. As there is no reliance on this system being present in order for the larger system to function, the architecture can easily adapt to its removal, and will attempt to calculate solutions to given goals that do not involve movement by the robot itself.

While this will reduce the functionality of the overall system, as well as limiting the assistance the companion can provide to the user, it also means the functionality is potentially more accessible. Users are not prevented from using all services simply because they cannot afford to purchase or run a larger robotic platform, as modules can be individually installed on other devices. It also allows the system to be easily customized to users who may not wish to use specific modules for any reason. This creates a more accessible and personalized system.

6.5 Accessibility through a Multimodal Interface

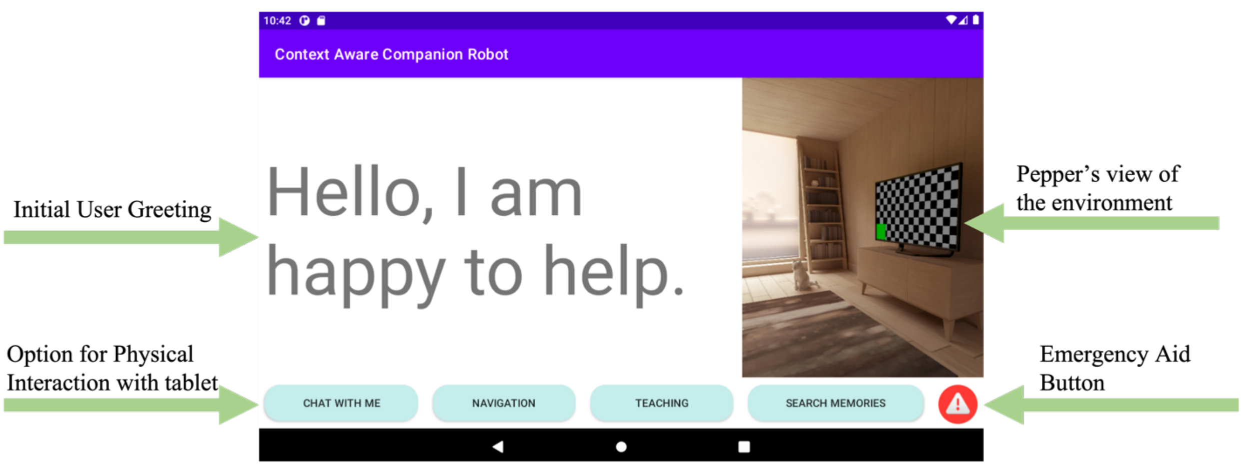


Figure 17 - The multimodal interface has been achieved through a custom app developed specifically for the Pepper robot. The Android application allows for multiple forms of input including typed commands, voice command, or allows users to select options using the set buttons located at the bottom of the interface. In the event that Pepper is currently in a monitoring mode, the interface will also automatically process objects detected in the environment in an attempt to anticipate possible actions.

Considering the target audience of this work, it was important to consider the accessibility of the robot for people who may have a diverse range of needs or physical limitations. There has been significant consideration of how the robot should communicate with the users, as well as how to allow them to communicate and respond in turn.

In order to be as effective as possible, a multi-modal user interface has been integrated into the main architecture to ensure that users can interact with Pepper in various ways depending on their preference. The Android tablet, attached to Pepper's chest, enables the user to view options on the screen, select option commands by touch, input information or requests through typed commands or provide verbal instructions. As the tablet is developed through Android, it is also possible to connect other Android devices to extend the range that users can interact with the robot.

To recognise a user's verbal command, the application takes advantage of the Phrase library as part of the integrated qiSDK libraries. This allowed for base commands to be recognised from a pre-defined list that trigger the robot to respond in a pre-determined way. Hearing these particular phrases also has the benefit of forcing the robot to redirect its focus to the users before providing a response from a list of answers.

These pre-defined responses provide a way to automatically trigger the robot's attention towards the user, however, they are limited to all phrases and answers being input into the system before the user even begins to use the architecture. To attempt to provide all possible conversations and requests would be not only impractical from a development standpoint but almost impossible given the range of social interactions possible. Therefore, an additional chatbot-style agent has been developed and integrated to expand the robot's social capacity and its ability to respond in a human-like way, as opposed to relying on pre-emptive guessing of possible commands. To expand the potential for conversation and accurate responses, the agent has been created using

Google's Dialogflow engine to extend the architecture through the use of Natural Language processing.

6.6 Integrating Dialogflow for Social Interaction

Dialogflow is an online natural language processing framework that allows developers to use the library to develop chatbot-style agents for websites, mobile applications, and other forms of conversational agents. The framework allows users to create 'agents', virtual assistants that interact with users and provide information and help depending on the context of the request. Agents make use of various features within Dialogflow such as flows, intents, actions, entities and forms. In this architecture, intents refer to the intention behind the user's request (such as 'locating' or 'memory extraction'), entities refer to the individual elements detected in the request ('book', 'computer' etc.) and actions are the follow-up reactions the system should take after detection of intents and entities.

Through the API, the integration into Android allows a way to quickly interpret what the user is requesting and trigger other systems through voice/text input.

For this work, an agent was created with the goal of replicating possible carer responses while still maintaining a friendly attitude towards users. Intents have been designed to match the main functionality of the companion architecture which not only helps provide a more 'human' response to requests but also helps process the natural language input of the user into a set of keywords that the observer can understand and process.

The agent was developed over time and through interactions with different users, with both the successful cases and unsuccessful ones, have been used to gradually adapt the intents and template answers into a more accurate and appropriate response.

An example of this training can be seen in Fig 18.

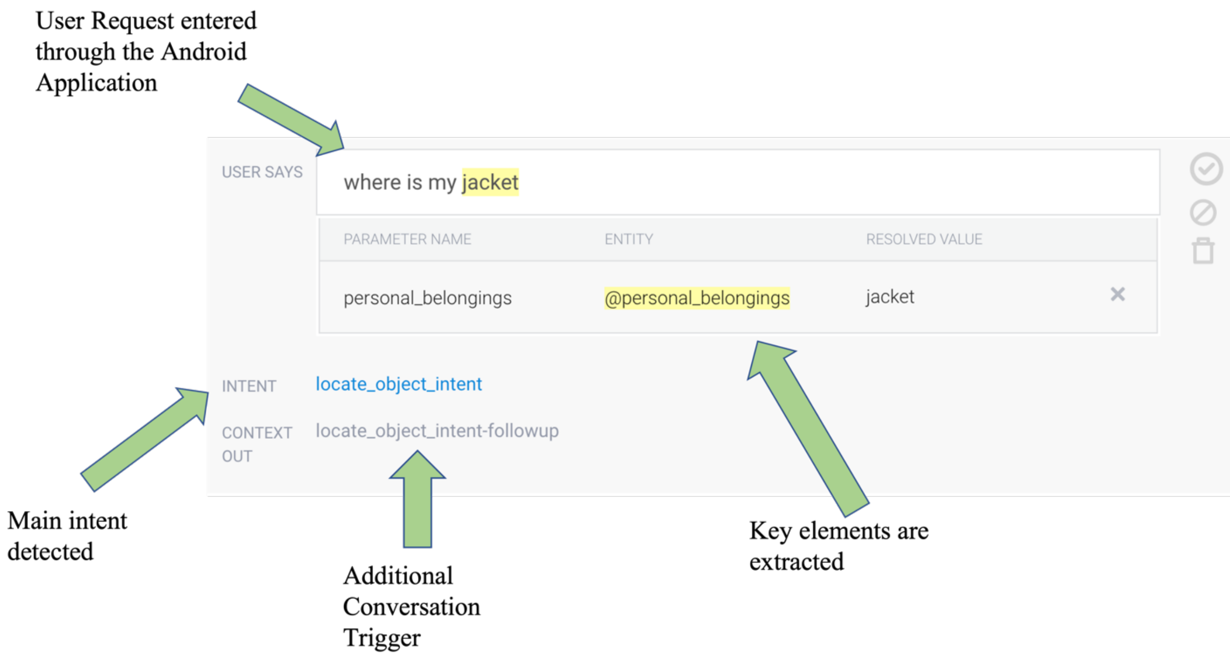


Figure 18 - User requests are processed using Dialogflow to help translate the request into a machine-readable format. Key elements of the request such as the main intention of the user and the context in terms of particular objects included in the request are extracted and returned to the main architecture for processing in the Observer.

To verify the agent's ability to act appropriately, the agent was interacted with multiple times over the course of several months to detect the common intents and entities needed for a conversation. Testing was then completed to show the responses and reactions of the robot using Dialogflow. The agent has been integrated into Pepper’s Android application to process the user input and provide more realistic 'human-like responses. The requests follow a particular pattern of input, processing through Dialogflow, processing the response in the observer (for navigation and memory if needed) and returning a response to the user.

6.7 External Contact for Emergencies

Across multiple studies, one of the key wants identified for social robots by both users and carers was the ability to call for external help in the event of accidents or general emergencies.

This was a particular concern throughout the project as while Pepper has been given several new social and intelligence upgrades, the robot's platform prevents it from providing any form of physical aid to the user. However, as research has shown that falls and risk of injury is high in the elderly, it was important to enable the companion to have some way to call for emergency help should these situations occur.

To provide this functionality, Pepper has been supplied with an external mobile device connected to the robot via a server installed on the mobile. An application was then installed onto the device that contains a server for receiving requests from robot to users as well as a client for sending responses or commands from user to robot.

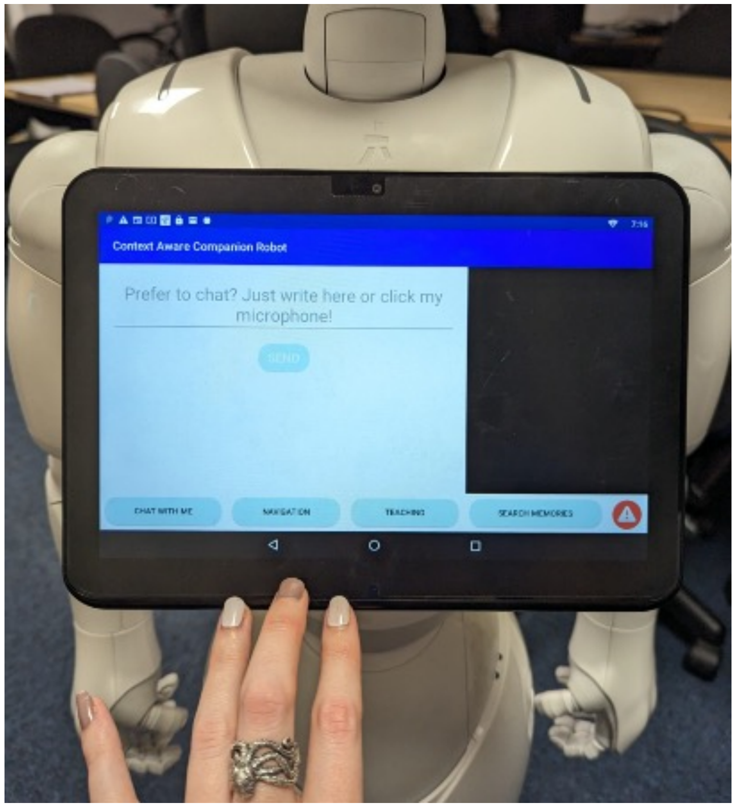


Figure 19 - An alternative way of contacting or providing commands is through typing on the main interface.

From the main social application installed on Pepper, users are given various ways to call for help in an emergency. These include an emergency call button accessible from each screen in the application (See Fig 17), a typed command through the main chat box, and the ability to verbally request help from Pepper. Each option is designed to be as visible and accessible as possible and contains a check once used to ensure accidental touch does not activate an emergency response. In the following experiment, the robot was left in an idle state, the same state that would be presented any time a user was not actively engaging with the companion. The robot can be seen idly monitoring the environment, as described in previous chapters when a user begins requesting help. During the experiment, each method was tested, and the results were recorded, while the companion was then reset between each method test to ensure the results represented that specific activation (as opposed to continuing to react to the previous activation).

For each activation method, a call was placed from Pepper to the server on the robot's external mobile. Each time, the server was able to successfully trigger a phone call on a selected contact's phone. In a real-world situation, this contact is selected by the user during the initial setup of the robot, and so would not require any effort from the user at the moment of calling.

There is however a notable limitation of this method. As the Pepper robot does not contain the technical capability to hold SIM cards, it was not possible to integrate the contact feature directly into the main application. This has resulted in a separate mobile device being required that must be kept close to the robot at all times. As Pepper was not capable of physically holding or supporting the phone, it was necessary to move the phone manually during the experiment to ensure it remained in range of the robot. For future work, this limitation would need to be considered when selecting the robotic platform used for trials, and the work could benefit from using an alternative robot than Pepper.

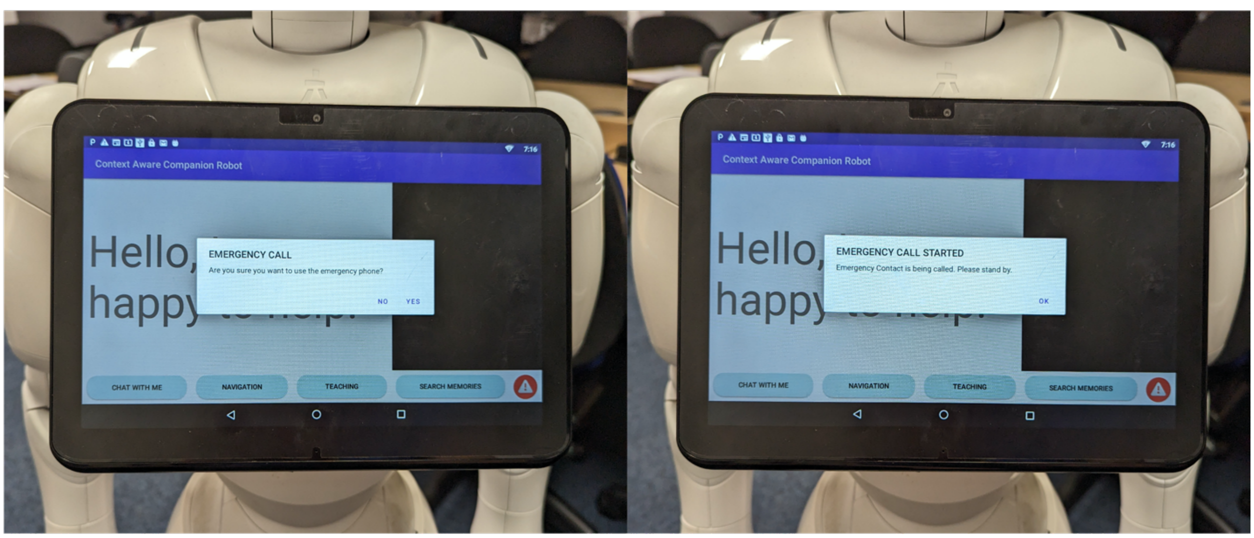


Figure 20 - Warnings are displayed before the emergency call is started. An option for users to cancel is provided.

6.8 Testing of a Full Architecture

To test the full capability of the system to act as a robotic companion, an experiment was designed to mimic the requirements of a daily companion robot in a user's home. For this experiment, a 'full cognitive architecture' is considered to include:

- The multimodal user interface (Chap. 6.5)
- The new episodic memory network (Chap. 5)
- The GNG model for navigation throughout the test area (Chap. 4)
- An external phone and companion app (Chap. 6.)
- Pepper and it's Android tablet

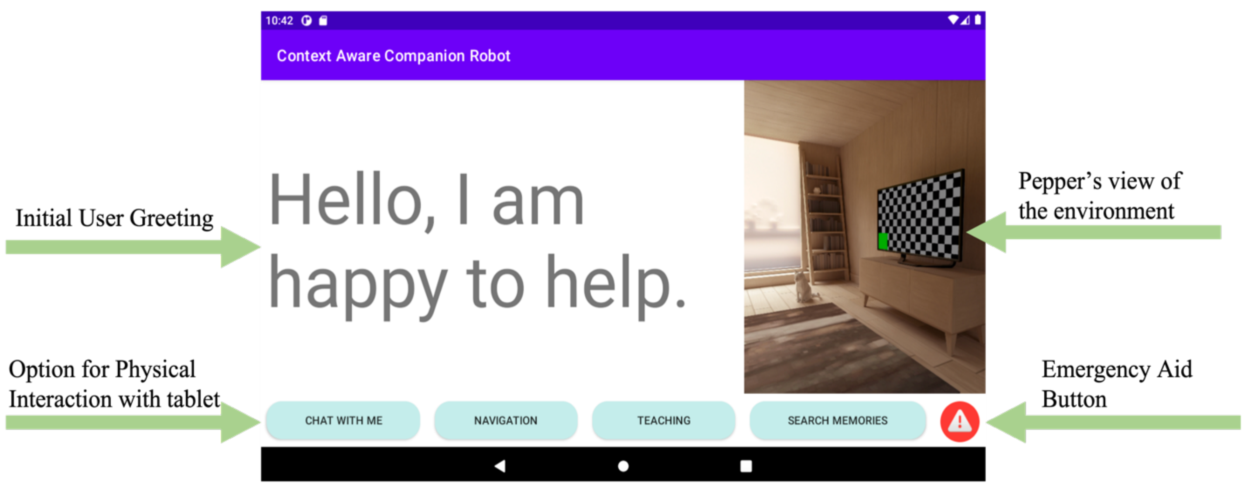


Figure 21 – The interface was used to introduce users to the system. Similar to previous experiments users can choose how they want to interact with the robot.

The goal was to make use of Pepper's new capabilities towards learning and recognising a new user, accepting requests and queries both recognised and unrecognised from the user, processing those requests through the combination of Dialogflow for natural language understanding and then later the observer acting as the centre of the architecture. It would also need to be able to accept new behaviour teachings from the user, locate items and traverse the area using the generated map.

If successful, the experiment would be expected to show the architecture's method of processing requests and adapting to new situations through recall of past experience and use of the mapped area through its combination of all systems including the episodic memory, navigation and social intelligence modules. Prior to the experiment, all knowledge of previous users was removed from Pepper to create a new base for learning, as well as the majority of memories and previous mapping results. In is important to note, that to prevent overfitting of the memories and therefore interference in the recall of learnt experiences, Pepper was provided with three 'core' memories that featured simple action sequences not used in the Lab environment. This would also be replicated in a real-world test as it would be expected for Pepper to know certain base behaviours rather than relying on elderly users to provide all knowledge.

The experiment took place within the Lavoro Lab located at the University of Essex. The lab provided a relatively open space that could be explored and mapped and provided a number of objects for Pepper to recognise making it a suitable area for testing.

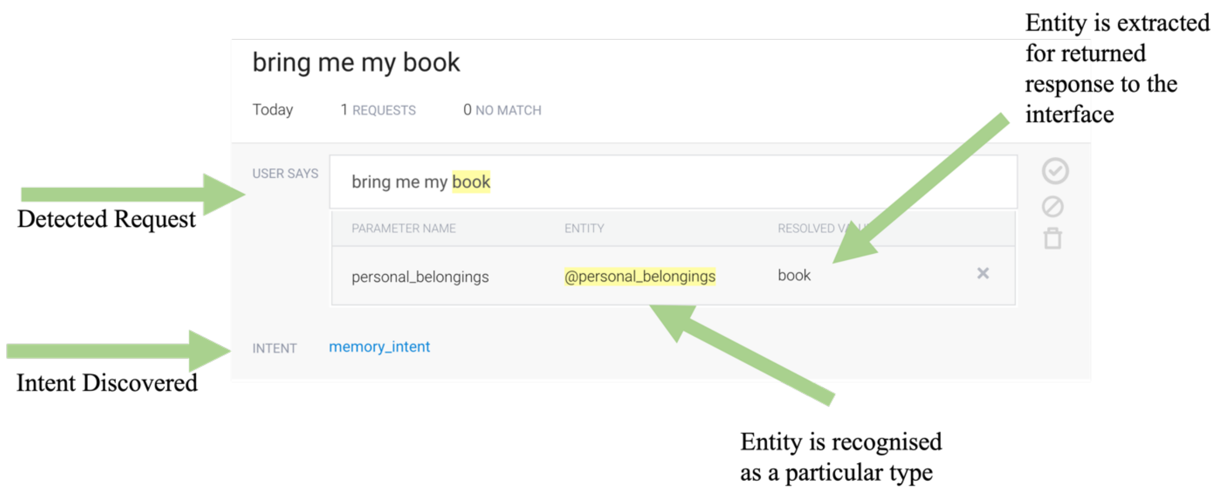


Figure 22 - The user's request was immediately sent to Dialogflow. The agent was successful in extracting the main intent and target entity. The results are returned to the main interface to be processed by the observer after the world context has been associated to it within a larger JSON object.

In this case, the given request was to ask the robot to bring an object (here the user requested their book) to the user's current location. This request contained two components: first to understand the request and the actions necessary to take in order to complete it (a memory task) and then to physically complete the request (a navigation task).

The first process that started upon getting the user request was an API call made to Dialogflow . Though both the Dialogflow call history and the returned information to the Android application, it can be seen that the request is recognised first as a memory task due to the key entities detected with the sentence.

$$\text{“}\underbrace{\text{Bring me my}}_{\text{INTENT}}\ \underbrace{\text{book}}_{\text{ENTITY}}\text{”}$$

Figure 23 - The basic form of extraction of a user request. Two major components are needed in order for the request to be considered valid by the architecture.

This detection is sent back to the Android application with a response consisting of the detected intent, entities found and a text string that can be read aloud or displayed onscreen by Pepper to tell the user what has happened in terms of their request.

The return of the API call triggered a success condition within the interface program to indicate that the processing procedures should now begin by calling the main components of the extended architecture through the server connection to the observer. To do this, and provide the current context of the situation, a snapshot of the visible environment was compiled using information from the robot's sensors, specifically, image frames collected from the front-facing camera, object detection results , ID's of any recognised users and the detected intent of the request, as calculated by the Dialogflow agent. The resulting snapshot included information pertaining to the intention (intent) of the request, the objects available in the world, the specific object requested and the user that is now recognised.

Once compiled the information was then autonomically sent to the central observer module for processing into a partial cue.

We can see the resulting cue for memory in the figure below.

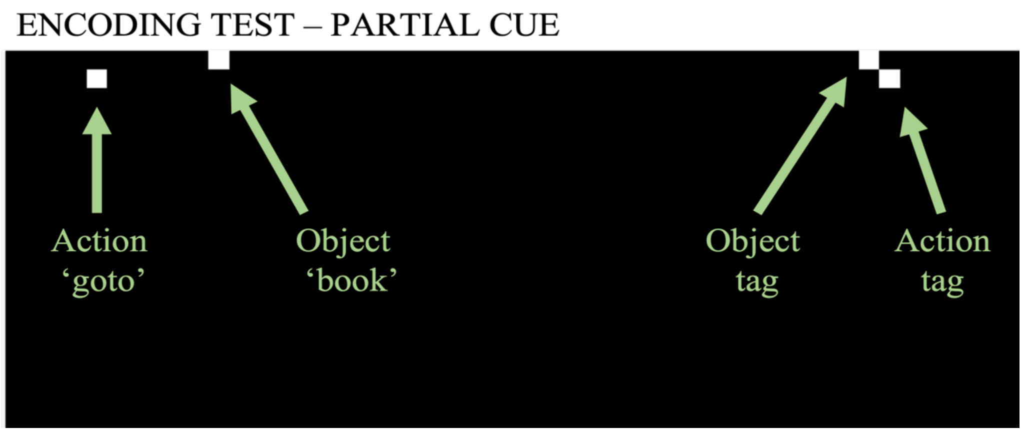


Figure 24 - The cue was built automatically by the observer on request from the interface. As the user was not originally known, the cue does not include a user specific tag.

Through comparison with the existing memories, the episodic memory was able to produce a matching, viable sequence of actions that can be completed by the robot in order to achieve the goal. As the extraction was successful, the result was sent back to the main observer to be processed to determine any additional actions needed from additional modules or whether the interface could be contacted to complete the actions.

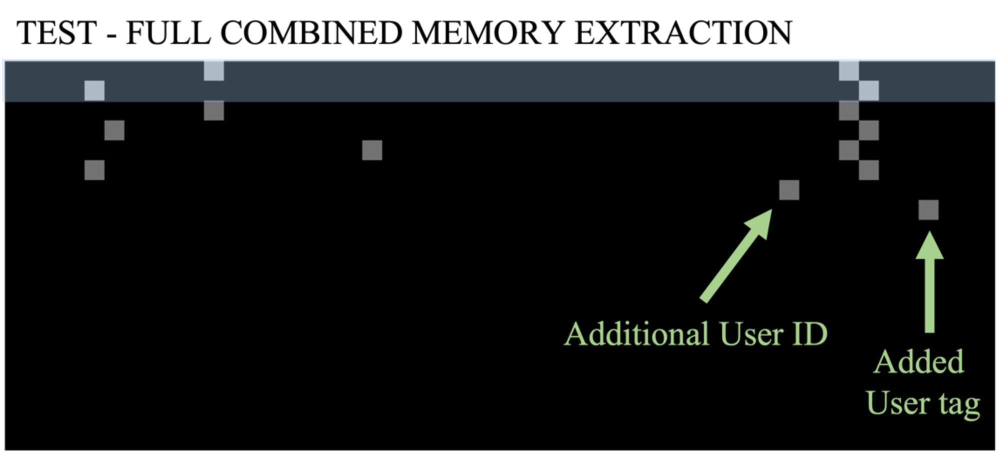


Figure 25 - The extracted memory based on the partial cue. A user tag has been added by the observer ready for later encoding, as the memory will now be considered for personalised responses in the future.

Because of the actions returned in the matching memory contained physical movement by Pepper (the 'goTo' action), the navigation was activated to detect the location of the book and the needed route for Pepper to follow to reach the goal.

To generate the needed path, the navigation used the map that was generated pre-experiment of the Lavoro lab to find a suitable path that Pepper could take.

Here we can see the generated map after human-guided exploration was completed within the lab.

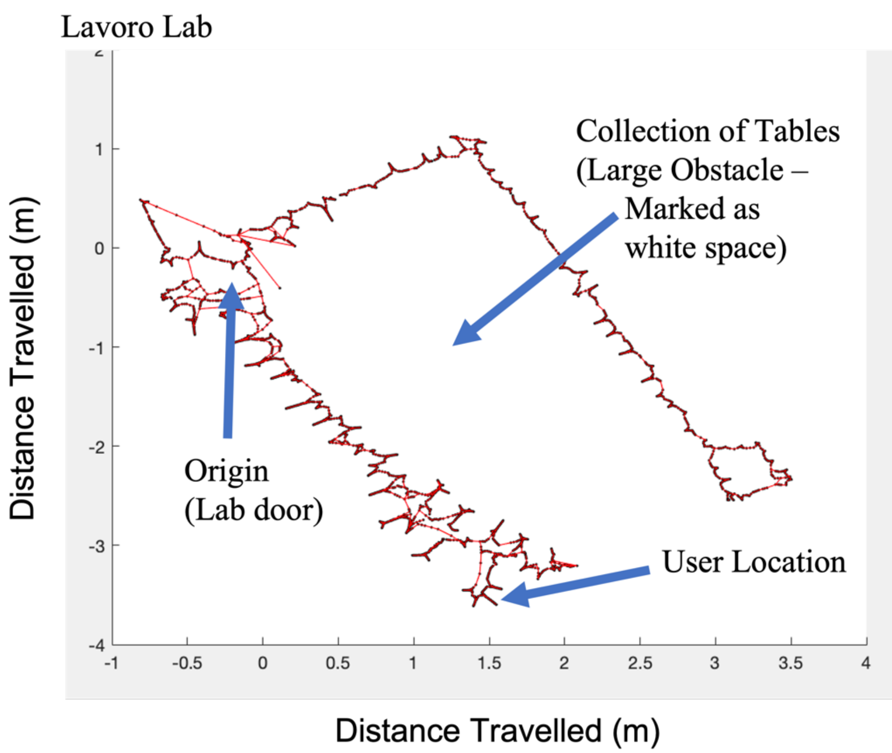


Figure 26 - The GNG model of the Lavoro Lab. Pepper collected data through human-guidance, with the resulting x, y, z coordinates trained using the network to produce the spatial map. The model was generated using 10000 data points, and completed training in approx. 3 minutes.

Using the generated map (Fig 26) a possible route to the goal was calculated and returned to the observer (As described in Chapter 4.). This in turn created a path for Pepper in a series of X, Y coordinate points stored in a matrix that was returned to the main interface. With the activation of both modules, the now processed data (Route to follow, action sequence) was returned to the main Android application. The server response triggered from the returned data, in turn triggered the functions developed to physically move Pepper. The physical movement was achieved through Pepper's Android SDK library. As Pepper moved, the object detection successfully kept tracking the world view of the robot, continuously sending the data to the observer on a 20 second interval. This provided the architecture with up-to-date information so that had the process been interrupted, the system would have begun the reprocessing procedures to assess the interruption, and using the external modules, recalculate the routes and/or action plan using the new information. For this case, there was no external obstacles present to interrupt the path of the robot, and therefore Pepper was able to reach the goal without interruption.

The successful reaching of the goal item also allowed for a direct transformation of the initial planned route to be reversed to calculate the path back to the user. It must be noted here that the physical limitations of Pepper prevented the user from actually receiving the object due to the lack of strength of the robot's joints preventing the lifting of items. However, with this exception , the architecture was able to fully process the request, extract the necessary goal and actions, return the calculated actions to the user interface, and finally control the robot to follow the routes given by the architecture.

7 Discussion

The main aim of this thesis was to provide a cognitive software architecture that when applied to a robot, in this case the Pepper humanoid, would create a social, robotic companion that considered both the social and spatial context of its environment. The work has combined a mixture of research from both biological and artificial perspectives to create a biologically inspired cognitive architecture capable of learning and recalling experiences, navigating dynamic unstructured environments and interacting with users in a socially intelligent way through personalized and un-personalized behaviours.

To complete the initial objective of a biological-inspired memory system, the work discussed in Chapter 5 details the implementation of a growing episodic memory for learning and recalling experiences. This has been presented through the results shown and involved the development of the module to provide this functionality and its integration into a larger framework. By expanding upon the work completed in [103] with the introduction of a social element to the memory, as well as changes to the process of recall and integration to the larger architecture, Pepper was able to not only learn and recall experiences but also distinguish between behaviours of different users, detect some dangerous behaviours, anticipate user behaviour and return action plans for the robot to follow to complete requests.

The second objective was the swift learning navigation system to allow Pepper to adapt quickly and efficiently to new environments. The resulting system, as described in Chapter 4, provides this functionality to quickly map new environments through human-guided exploration, navigate through those environments using goal-directed spatial reasoning, adapt to changes in the environment detected upon return to previously mapped areas and a human-aware element designed to prevent collisions with users.

The development of the social module has been achieved through the combination of the development of the Android application installed on Pepper, the integration of a custom Dialogflow agent for input processing and the addition of the previously mentioned social memory options.

The modules were then combined to form the overall cognitive architecture connected through a series of servers and ROS2 libraries, allowing for real-time communication between modules and continuous updates of the current environment. Together, the software developed provided the necessary functionality to meet several requirements of a social companion.

Notable limitations discovered throughout the development of the architecture are as follows. The Pepper robot itself was an obstacle at some stages of the research due to its lack of physical strength, poor network connections (Pepper 1) and the tendency to lose its ability to correctly hear users in crowds. The initial Python-based platform was less advanced than its Java-based successor, and while the switch to the later platform provided far more benefits than disadvantages, it created a period time where development was focused on learning how to integrate the system in its early stage into a new language, and onto a new platform that lost access to a number of base sensors.

There are multiple potential implications for future work based on the architecture developed in this thesis. As the majority of the work is platform agnostic, it would not be difficult to split the modules and adapt them to new purposes outside of the current objectives. Within healthcare, the navigation system presented in Chapter 4 could easily be applied to any mobile robot such as automated disinfection robots or trolley-based distribution platforms for food or medicine. The episodic memory in Chapter 5 could be easily modified to include a different series of tags making it capable of tracking locations, and landmark-based action sequences through memory alone or altering the social element to further customize the system, recognizing the difference between roles of different users. Outside of healthcare, there are multiple areas it could also benefit. Dangerous tasks that present risks to human operators, such as nuclear robotics, often require platforms capable of navigating through unstructured, unpredictable environments considered too unsafe for humans.

Throughout this work, the architecture was significantly influenced by the current understanding of multiple neuroscience topics including episodic memory function within the brain, navigation techniques for humans and animals and the results of understanding the social context within an environment to influence responses with the goal of replicating elements of this behaviour on Pepper. This influence can be seen throughout the work and results presented in this thesis as well as the overall design of the architecture. Specifically, the overall role of the hippocampus which was shown in the research review to have an influence on episodic memory, understanding elements of social context and mapping of spatial information was a particular influence on the design of the connections in the architecture and the distribution of work between modules. Additionally, the GWT and theory of how the brain processes both conscious and unconscious information heavily influenced the role of the Observer module and the connections to the external module. The architecture as a whole also took strong motivation from research into the Default Mode Network [109] which provides evidence of specific brain regions that activate when the individual is not focused on specific tasks or external events. These regions instead activate during daydreaming, imagination tasks and general self-based thinking (such as experiences). This heavily influenced the decision to have the Observer request possible behaviours and actions from the episodic memory, even while the robot was only monitoring. These 'imagined' scenarios that are returned are often discarded in the current architecture but provide a potential way of allowing the robot to recognise weak areas in its own learning and knowledge in future work.

8 Conclusion

In summary, the goal of this thesis was to create a context-aware robotic companion through the development and integration of an artificial cognitive architecture. This companion was designed to act as a companion or care assistant to users with long-term health conditions as well as function as an assistant to existing carers. The final version of this architecture now includes navigation, memory and social modules that aim to provide real-time, personalised assistance to users in a quick and efficient manner.

The main technological contributions of this thesis can be seen in the swift-learning navigation system, the biologically inspired memory and the overall architecture implementation. The MGNG navigation model and its use of multimodal nodes present a novel way of providing a robot with a quick and efficient way of modelling new and unstructured environments. These models in turn provide an efficient way for the robot to locate objects, guide users, and complete navigation-related goals.

The episodic memory and its use for understanding social behaviours is a novel contribution to the field of care robotics. By predicting and learning behaviours specific to individual users, the robot can offer more personalised responses, ensuring the patient is provided with a companion they can customise to suit their needs and routines, as opposed to relying on predetermined responses. From a larger social and healthcare perspective, providing a personalised companion to users who would otherwise be at risk of not receiving the level of care required, contributes to reducing the risk of loneliness through lack of social interaction in elderly users and helps to prevent the negative health effects often associated with this. Additionally, having the robot provide personalised care and responses assists in removing the fear many patients and their families have of having human contact from healthcare workers replaced by impersonal robotic assistants. The project was specifically designed to assist existing care workers or to fill a role where a human was not already available, to help promote the idea of human-robot collaborative care as opposed to contributing to the fear of an individual having to accept a robot as a carer or workers being replaced by machines.

By designing the architecture in a modular way, the project presents users with a way of selecting specific functions they need for their unique conditions or removing the robot platform element and replacing it with something more financially viable. This helps make the architecture more accessible to a larger user base and has the potential to contribute to the overall integration of artificial companions and carers in general.

Economically, the project contributes to the discussion surrounding the issue of the cost of care, which can be a significant burden to both patients and their carers (see chapter 3) and can significantly affect the level of care patients receive. Robotic companions have been used in multiple studies in recent years, and this thesis provides additional evidence of a robotic companion as a potential solution to both the lack of carers available, cost of care to users and their families, as well as a country wide funding issue.

Finally, I believe the work in this thesis contributes further evidence that robots have the potential to be a beneficial and positive force within healthcare and communities. By directly working to improve the lives of users and assist their carers, whether professional or family members, it helps to form a favourable view of robotic companions in general and should a project such as this be available commercially, would provide a cost-effective way of guaranteeing individual social care when a human option is not available.

9 References

[1] 'Population estimates for the UK, England, Wales, Scotland and Northern Ireland - Office for National Statistics'. Accessed: Mar. 16, 2023. [Online]. Available: https://www.ons.gov.uk/peoplepopulationandcommunity/populationandmigration/populationestimates/bulletins/annualmidyearpopulationestimates/mid2021#the-uk-population-at-mid-2021

[2] MHA, 'Five_calls_to_help_people_live_later_life_well_MHA_manifesto.pdf'. MHA - Methodist Homes. Accessed: Jul. 11, 2023. [Online]. Available: https://www.mha.org.uk/files/8015/8168/1416/Five_calls_to_help_people_live_later_life_well_MHA_manifesto.pdf

[3] 'Healthcare expenditure, UK Health Accounts - Office for National Statistics'. Accessed: Aug. 09, 2022. [Online]. Available: https://www.ons.gov.uk/peoplepopulationandcommunity/healthandsocialcare/healthcaresystem/bulletins/ukhealthaccounts/2020#long-term-care-expenditure

[4] 'Overview of the UK population - Office for National Statistics'. Accessed: Oct. 19, 2022. [Online]. Available: https://www.ons.gov.uk/peoplepopulationandcommunity/populationandmigration/populationestimates/articles/overviewoftheukpopulation/2020

[5] 'Estimates of the very old, including centenarians, UK - Office for National Statistics'. Accessed: Oct. 19, 2022. [Online]. Available: https://www.ons.gov.uk/peoplepopulationandcommunity/birthsdeathsandmarriages/ageing/bulletins/estimatesoftheveryoldincludingcentenarians/2002to2020

[6] 'Mid-year population estimates of the very old, including centenarians: UK - Office for National Statistics'. Accessed: Oct. 19, 2022. [Online]. Available: https://www.ons.gov.uk/peoplepopulationandcommunity/birthsdeathsandmarriages/ageing/datasets/midyearpopulationestimatesoftheveryoldincludingcentenariansunitedkingdom

[7] T. Prescott and P. Caleb-Solly, 'Robotics in Social Care: A Connected Care EcoSystem for Independent Living', EPSRC UK-RAS Network, UKRAS White Papers, Jun. 2017. doi: 10.31256/WP2017.3.

[8] https://www.facebook.com/nhswebsite, 'Paying for your own social care (self-funding) - Social care and support guide', nhs.uk. Accessed: Jul. 24, 2023. [Online]. Available: https://www.nhs.uk/conditions/social-care-and-support-guide/money-work-and-benefits/paying-for-your-own-care-self-funding/

[9] 'Care home fees across the UK explained - Which?' Accessed: Jul. 24, 2023. [Online]. Available: https://www.which.co.uk/money/pensions-and-retirement/financing-later-life-care/care-home-finance/care-home-fees-ak7lP4h1owcg

[10] 'Key facts and figures', Carers UK. Accessed: Jul. 12, 2023. [Online]. Available: https://www.carersuk.org/policy-and-research/key-facts-and-figures/

[11] 'Living longer - Office for National Statistics'. Accessed: Jul. 12, 2023. [Online]. Available: https://www.ons.gov.uk/peoplepopulationandcommunity/birthsdeathsandmarriages/ageing/articles/livinglongerhowourpopulationischangingandwhyitmatters/2019-03-15#who-is-providing-unpaid-care

[12] A. Shankar, M. Hamer, A. McMunn, and A. Steptoe, ‘Social isolation and loneliness: relationships with cognitive function during 4 years of follow-up in the English Longitudinal Study of Ageing’, *Psychosom. Med.*, vol. 75, no. 2, pp. 161–170, Feb. 2013, doi: 10.1097/PSY.0b013e31827f09cd.

[13] J. Golden *et al.*, ‘Loneliness, social support networks, mood and wellbeing in community-dwelling elderly’, *Int. J. Geriatr. Psychiatry*, vol. 24, no. 7, pp. 694–700, 2009, doi: 10.1002/gps.2181.

[14] S. B. Rafnsson, M. Orrell, E. d’Orsi, E. Hogervorst, and A. Steptoe, ‘Loneliness, Social Integration, and Incident Dementia Over 6 Years: Prospective Findings From the English Longitudinal Study of Ageing’, *J. Gerontol. B. Psychol. Sci. Soc. Sci.*, vol. 75, no. 1, pp. 114–124, Jan. 2020, doi: 10.1093/geronb/gbx087.

[15] J. T. Cacioppo and L. C. Hawkley, ‘Perceived social isolation and cognition’, *Trends Cogn. Sci.*, vol. 13, no. 10, pp. 447–454, Oct. 2009, doi: 10.1016/j.tics.2009.06.005.

[16] J. Holt-Lunstad, T. B. Smith, and J. B. Layton, ‘Social Relationships and Mortality Risk: A Meta-analytic Review’, *PLOS Med.*, vol. 7, no. 7, p. e1000316, Jul. 2010, doi: 10.1371/journal.pmed.1000316.

[17] S. Deandrea, E. Lucenteforte, F. Bravi, R. Foschi, C. La Vecchia, and E. Negri, ‘Risk Factors for Falls in Community-dwelling Older People: A Systematic Review and Meta-analysis’, *Epidemiology*, vol. 21, no. 5, p. 658, Sep. 2010, doi: 10.1097/EDE.0b013e3181e89905.

[18] ‘Overview | Falls in older people: assessing risk and prevention | Guidance | NICE’. Accessed: Jul. 24, 2023. [Online]. Available: https://www.nice.org.uk/guidance/cg161

[19] ‘Robotic Assistant for MCI patients at home | RAMCIP Project | Fact Sheet | H2020 | CORDIS | European Commission’. Accessed: Oct. 19, 2022. [Online]. Available: https://cordis.europa.eu/project/id/643433

[20] E. Tulving, *Elements of Episodic Memory*. Oxford University Press, 1983.

[21] E. Tulving, ‘What Is Episodic Memory?’, *Curr. Dir. Psychol. Sci.*, vol. 2, no. 3, pp. 67–70, Jun. 1993, doi: 10.1111/1467-8721.ep10770899.

[22] D. Zeithamova, M. Schlichting, and A. Preston, ‘The hippocampus and inferential reasoning: building memories to navigate future decisions’, *Front. Hum. Neurosci.*, vol. 6, 2012, Accessed: Oct. 03, 2023. [Online]. Available: https://www.frontiersin.org/articles/10.3389/fnhum.2012.00070

[23] W. B. Scoville and B. Milner, ‘Loss of recent memory after bilateral hippocampal lesions. 1957’, *J. Neuropsychiatry Clin. Neurosci.*, vol. 12, no. 1, pp. 103–113, 2000, doi: 10.1176/jnp.12.1.103.

[24] E. Teng and L. R. Squire, ‘Memory for places learned long ago is intact after hippocampal damage’, *Nature*, vol. 400, no. 6745, Art. no. 6745, Aug. 1999, doi: 10.1038/23276.

[25] D. Gaffan, ‘Scene-specific memory for objects: a model of episodic memory impairment in monkeys with fornix transection’, *J. Cogn. Neurosci.*, vol. 6, no. 4, pp. 305–320, 1994, doi: 10.1162/jocn.1994.6.4.305.

[26] D. Griffiths, A. Dickinson, and N. Clayton, ‘Episodic memory: what can animals remember about their past?’, *Trends Cogn. Sci.*, vol. 3, no. 2, pp. 74–80, Feb. 1999, doi: 10.1016/S1364-6613(98)01272-8.

[27] M. Bunsey and H. Eichenbaum, ‘Conservation of hippocampal memory function in rats and humans’, *Nature*, vol. 379, no. 6562, pp. 255–257, Jan. 1996, doi: 10.1038/379255a0.

[28] A. N. Meltzoff, 'The "like me" framework for recognizing and becoming an intentional agent', *Acta Psychol. (Amst.)*, vol. 124, no. 1, pp. 26–43, Jan. 2007, doi: 10.1016/j.actpsy.2006.09.005.

[29] A. A. Bhat and V. Mohan, 'Goal-Directed Reasoning and Cooperation in Robots in Shared Workspaces: an Internal Simulation Based Neural Framework', *Cogn. Comput.*, vol. 10, no. 4, pp. 558–576, Aug. 2018, doi: 10.1007/s12559-018-9553-1.

[30] S. Hirata and K. Fuwa, 'Chimpanzees (Pan troglodytes) learn to act with other individuals in a cooperative task', *Primates J. Primatol.*, vol. 48, no. 1, pp. 13–21, Jan. 2007, doi: 10.1007/s10329-006-0022-1.

[31] J. M. Plotnik, R. Lair, W. Suphachoksahakun, and F. B. M. de Waal, 'Elephants know when they need a helping trunk in a cooperative task', *Proc. Natl. Acad. Sci. U. S. A.*, vol. 108, no. 12, pp. 5116–5121, Mar. 2011, doi: 10.1073/pnas.1101765108.

[32] E. Blair, Aesop, and D. Silverman, *The crow and the pitcher: a retelling of Aesop's fable*. in Read-it! readers. Yellow level. Minneapolis: Picture Window Books, 2004.

[33] 'New Caledonian Crows Learn the Functional Properties of Novel Tool Types | PLOS ONE'. Accessed: Oct. 03, 2023. [Online]. Available: https://journals.plos.org/plosone/article?id=10.1371/journal.pone.0026887

[34] S. A. Jelbert, A. H. Taylor, L. G. Cheke, N. S. Clayton, and R. D. Gray, 'Using the Aesop's Fable Paradigm to Investigate Causal Understanding of Water Displacement by New Caledonian Crows', *PLoS ONE*, vol. 9, no. 3, p. e92895, Mar. 2014, doi: 10.1371/journal.pone.0092895.

[35] C. D. Bird and N. J. Emery, 'Rooks Use Stones to Raise the Water Level to Reach a Floating Worm', *Curr. Biol.*, vol. 19, no. 16, pp. 1410–1414, Aug. 2009, doi: 10.1016/j.cub.2009.07.033.

[36] A. M. Seed, S. Tebbich, N. J. Emery, and N. S. Clayton, 'Investigating physical cognition in rooks, Corvus frugilegus', *Curr. Biol. CB*, vol. 16, no. 7, pp. 697–701, Apr. 2006, doi: 10.1016/j.cub.2006.02.066.

[37] A. N. Frankenstein, M. P. McCurdy, A. M. Sklenar, R. Pandya, K. K. Szpunar, and E. D. Leshikar, 'Future thinking about social targets: The influence of prediction outcome on memory', *Cognition*, vol. 204, p. 104390, Nov. 2020, doi: 10.1016/j.cognition.2020.104390.

[38] H. Eichenbaum, 'The Hippocampus as a Cognitive Map … of Social Space', *Neuron*, vol. 87, no. 1, pp. 9–11, Jul. 2015, doi: 10.1016/j.neuron.2015.06.013.

[39] P. P. Kadwe, A. M. Sklenar, A. N. Frankenstein, P. U. Levy, and E. D. Leshikar, 'The influence of memory on approach and avoidance decisions: Investigating the role of episodic memory in social decision making', *Cognition*, vol. 225, p. 105072, Aug. 2022, doi: 10.1016/j.cognition.2022.105072.

[40] R. M. Tavares *et al.*, 'A Map for Social Navigation in the Human Brain', *Neuron*, vol. 87, no. 1, pp. 231–243, Jul. 2015, doi: 10.1016/j.neuron.2015.06.011.

[41] E. C. Tolman, 'Cognitive maps in rats and men', *Psychol. Rev.*, vol. 55, no. 4, pp. 189–208, 1948, doi: 10.1037/h0061626.

[42] C. M. Bird and N. Burgess, 'The hippocampus and memory: insights from spatial processing', *Nat. Rev. Neurosci.*, vol. 9, no. 3, Art. no. 3, Mar. 2008, doi: 10.1038/nrn2335.

[43] J. O'Keefe and L. Nadel, *The hippocampus as a cognitive map*. Oxford : New York: Clarendon Press ; Oxford University Press, 1978.

[44] B. E. Pfeiffer and D. J. Foster, 'Hippocampal place-cell sequences depict future paths to remembered goals', *Nature*, vol. 497, no. 7447, Art. no. 7447, May 2013, doi: 10.1038/nature12112.

[45] T. Hartley, C. Lever, N. Burgess, and J. O'Keefe, 'Space in the brain: how the hippocampal formation supports spatial cognition', *Philos. Trans. R. Soc. B Biol. Sci.*, vol. 369, no. 1635, p. 20120510, Feb. 2014, doi: 10.1098/rstb.2012.0510.

[46] M. Fyhn, S. Molden, M. P. Witter, E. I. Moser, and M.-B. Moser, 'Spatial representation in the entorhinal cortex', *Science*, vol. 305, no. 5688, pp. 1258–1264, Aug. 2004, doi: 10.1126/science.1099901.

[47] T. Hafting, M. Fyhn, S. Molden, M.-B. Moser, and E. I. Moser, 'Microstructure of a spatial map in the entorhinal cortex', *Nature*, vol. 436, no. 7052, Art. no. 7052, Aug. 2005, doi: 10.1038/nature03721.

[48] V. Edvardsen, A. Bicanski, and N. Burgess, 'Navigating with grid and place cells in cluttered environments', *Hippocampus*, vol. 30, no. 3, pp. 220–232, Mar. 2020, doi: 10.1002/hipo.23147.

[49] C. Lever, S. Burton, A. Jeewajee, J. O'Keefe, and N. Burgess, 'Boundary vector cells in the subiculum of the hippocampal formation', *J. Neurosci. Off. J. Soc. Neurosci.*, vol. 29, no. 31, pp. 9771–9777, Aug. 2009, doi: 10.1523/JNEUROSCI.1319-09.2009.

[50] T. Solstad, C. N. Boccara, E. Kropff, M.-B. Moser, and E. I. Moser, 'Representation of geometric borders in the entorhinal cortex', *Science*, vol. 322, no. 5909, pp. 1865–1868, Dec. 2008, doi: 10.1126/science.1166466.

[51] M. E. Sorrows and S. C. Hirtle, 'The Nature of Landmarks for Real and Electronic Spaces', in *Spatial Information Theory. Cognitive and Computational Foundations of Geographic Information Science*, C. Freksa and D. M. Mark, Eds., in Lecture Notes in Computer Science. Berlin, Heidelberg: Springer, 1999, pp. 37–50. doi: 10.1007/3-540-48384-5_3.

[52] N. J. Fortin, K. L. Agster, and H. B. Eichenbaum, 'Critical role of the hippocampus in memory for sequences of events', *Nat. Neurosci.*, vol. 5, no. 5, Art. no. 5, May 2002, doi: 10.1038/nn834.

[53] T. I. Brown, R. S. Ross, J. B. Keller, M. E. Hasselmo, and C. E. Stern, 'Which Way Was I Going? Contextual Retrieval Supports the Disambiguation of Well Learned Overlapping Navigational Routes', *J. Neurosci.*, vol. 30, no. 21, pp. 7414–7422, May 2010, doi: 10.1523/JNEUROSCI.6021-09.2010.

[54] M. E. Hasselmo and H. B. Eichenbaum, 'Hippocampal mechanisms for the context-dependent retrieval of episodes', *Neural Netw. Off. J. Int. Neural Netw. Soc.*, vol. 18, no. 9, pp. 1172–1190, Nov. 2005, doi: 10.1016/j.neunet.2005.08.007.

[55] S. Alyan and B. L. McNaughton, 'Hippocampectomized rats are capable of homing by path integration', *Behav. Neurosci.*, vol. 113, no. 1, pp. 19–31, Feb. 1999, doi: 10.1037//0735-7044.113.1.19.

[56] H. Mittelstaedt and M.-L. Mittelstaedt, 'Homing by Path Integration', in *Avian Navigation*, F. Papi and H. G. Wallraff, Eds., in Proceedings in Life Sciences. Berlin, Heidelberg: Springer, 1982, pp. 290–297. doi: 10.1007/978-3-642-68616-0_29.

[57] T. S. Collett and P. Graham, 'Animal Navigation: Path Integration, Visual Landmarks and Cognitive Maps', *Curr. Biol.*, vol. 14, no. 12, pp. R475–R477, Jun. 2004, doi: 10.1016/j.cub.2004.06.013.

[58] A. S. Etienne, R. Maurer, V. Boulens, A. Levy, and T. Rowe, 'Resetting the path integrator: a basic condition for route-based navigation', *J. Exp. Biol.*, vol. 207, no. Pt 9, pp. 1491–1508, Apr. 2004, doi: 10.1242/jeb.00906.

[59] J. Borenstein, J. Borenstein, H. R. Everett, and L. Fen, 'Where am I?' Sensors and methods for mobile robot positioning', in *CiteSeerX*, 1996. [Online]. Available: https://citeseerx.ist.psu.edu/doc/10.1.1.228.903

[60] J. Borenstein and L. Feng, 'Measurement and correction of systematic odometry errors in mobile robots', *IEEE Trans. Robot. Autom.*, vol. 12, no. 6, pp. 869–880, Dec. 1996, doi: 10.1109/70.544770.

[61] J. J. Leonard and H. F. Durrant-Whyte, 'Simultaneous map building and localization for an autonomous mobile robot', in *Proceedings IROS '91:IEEE/RSJ International Workshop on Intelligent Robots and Systems '91*, Nov. 1991, pp. 1442–1447 vol.3. doi: 10.1109/IROS.1991.174711.

[62] H. Durrant-Whyte and T. Bailey, 'Simultaneous localization and mapping: part I', *IEEE Robot. Autom. Mag.*, vol. 13, no. 2, pp. 99–110, Jun. 2006, doi: 10.1109/MRA.2006.1638022.

[63] R. Smith, M. Self, and P. Cheeseman, 'A stochastic map for uncertain spatial relationships', in *Proceedings of the 4th international symposium on Robotics Research*, Cambridge, MA, USA: MIT Press, May 1988, pp. 467–474.

[64] T. Bailey, J. Nieto, J. Guivant, M. Stevens, and E. Nebot, 'Consistency of the EKF-SLAM Algorithm', in *2006 IEEE/RSJ International Conference on Intelligent Robots and Systems*, Oct. 2006, pp. 3562–3568. doi: 10.1109/IROS.2006.281644.

[65] J. B. Folkesson and H. I. Christensen, 'Robust SLAM', *IFAC Proc. Vol.*, vol. 37, no. 8, pp. 722–727, Jul. 2004, doi: 10.1016/S1474-6670(17)32064-5.

[66] J. Folkesson and H. Christensen, 'Graphical SLAM - a self-correcting map', in *IEEE International Conference on Robotics and Automation, 2004. Proceedings. ICRA '04. 2004*, Apr. 2004, pp. 383-390 Vol.1. doi: 10.1109/ROBOT.2004.1307180.

[67] M. Montemerlo, S. Thrun, D. Koller, and B. Wegbreit, 'FastSLAM: a factored solution to the simultaneous localization and mapping problem', in *Eighteenth national conference on Artificial intelligence*, USA: American Association for Artificial Intelligence, Jul. 2002, pp. 593–598.

[68] Davison, 'Real-time simultaneous localisation and mapping with a single camera', in *Proceedings Ninth IEEE International Conference on Computer Vision*, Nice, France: IEEE, 2003, pp. 1403–1410 vol.2. doi: 10.1109/ICCV.2003.1238654.

[69] K. Yousif, A. Bab-Hadiashar, and R. Hoseinnezhad, 'An Overview to Visual Odometry and Visual SLAM: Applications to Mobile Robotics', *Intell. Ind. Syst.*, vol. 1, no. 4, pp. 289–311, Dec. 2015, doi: 10.1007/s40903-015-0032-7.

[70] V. Perera, T. Pereira, J. Connell, and M. Veloso, 'Setting Up Pepper For Autonomous Navigation And Personalized Interaction With Users'. arXiv, Apr. 16, 2017. doi: 10.48550/arXiv.1704.04797.

[71] M. E. Hagen *et al.*, 'Reducing cost of surgery by avoiding complications: the model of robotic Roux-en-Y gastric bypass', *Obes. Surg.*, vol. 22, no. 1, pp. 52–61, Jan. 2012, doi: 10.1007/s11695-011-0422-1.

[72] J. Douissard, M. E. Hagen, and P. Morel, 'The da Vinci Surgical System', in *Bariatric Robotic Surgery: A Comprehensive Guide*, C. E. Domene, K. C. Kim, R. Vilallonga Puy, and P. Volpe, Eds., Cham: Springer International Publishing, 2019, pp. 13–27. doi: 10.1007/978-3-030-17223-7_3.

[73] M. M. Fourman and A. A. Saber, 'Robotic bariatric surgery: a systematic review', *Surg. Obes. Relat. Dis. Off. J. Am. Soc. Bariatr. Surg.*, vol. 8, no. 4, pp. 483–488, 2012, doi: 10.1016/j.soard.2012.02.012.

[74] C. Vaida *et al.*, 'Systematic Design of a Parallel Robotic System for Lower Limb Rehabilitation', *IEEE Access*, vol. 8, pp. 34522–34537, 2020, doi: 10.1109/ACCESS.2020.2974295.

[75] G. Morone, I. Cocchi, S. Paolucci, and M. Iosa, 'Robot-assisted therapy for arm recovery for stroke patients: state of the art and clinical implication', *Expert Rev. Med. Devices*, vol. 17, no. 3, pp. 223–233, Mar. 2020, doi: 10.1080/17434440.2020.1733408.

[76] L. Hung *et al.*, 'Telepresence Robots in Long-Term Care Settings in British Columbia During the COVID-19 Pandemic: Analyzing the Experiences of Residents and Family Members', *Gerontol. Geriatr. Med.*, vol. 9, p. 23337214231166208, Apr. 2023, doi: 10.1177/23337214231166208.

[77] W. Moyle *et al.*, 'Use of a Robotic Seal as a Therapeutic Tool to Improve Dementia Symptoms: A Cluster-Randomized Controlled Trial', *J. Am. Med. Dir. Assoc.*, vol. 18, no. 9, pp. 766–773, Sep. 2017, doi: 10.1016/j.jamda.2017.03.018.

[78] S. Petersen, S. Houston, H. Qin, C. Tague, and J. Studley, 'The Utilization of Robotic Pets in Dementia Care', *J. Alzheimers Dis.*, vol. 55, no. 2, pp. 569–574, Jan. 2017, doi: 10.3233/JAD-160703.

[79] M. E. Pollack *et al.*, 'Pearl: A Mobile Robotic Assistant for the Elderly'.

[80] 'What can SARA do - SARA Robotics'. Accessed: Oct. 04, 2023. [Online]. Available: https://sara-robotics.nl/zorgrobot-sara/

[81] 'NAO the humanoid and programmable robot | Aldebaran'. Accessed: Oct. 04, 2023. [Online]. Available: https://www.aldebaran.com/en/nao

[82] K. Blindheim, M. Solberg, I. A. Hameed, and R. E. Alnes, 'Promoting activity in long-term care facilities with the social robot Pepper: a pilot study', *Inform. Health Soc. Care*, vol. 48, no. 2, pp. 181–195, Apr. 2023, doi: 10.1080/17538157.2022.2086465.

[83] M. S. Kaiser, S. Al Mamun, M. Mahmud, and M. H. Tania, 'Healthcare Robots to Combat COVID-19', in *COVID-19: Prediction, Decision-Making, and its Impacts*, K. C. Santosh and A. Joshi, Eds., in Lecture Notes on Data Engineering and Communications Technologies. , Singapore: Springer, 2021, pp. 83–97. doi: 10.1007/978-981-15-9682-7_10.

[84] M. Ghafurian, C. Ellard, and K. Dautenhahn, 'Social Companion Robots to Reduce Isolation: A Perception Change Due to COVID-19', in *Human-Computer Interaction – INTERACT 2021*, C. Ardito, R. Lanzilotti, A. Malizia, H. Petrie, A. Piccinno, G. Desolda, and K. Inkpen, Eds., in Lecture Notes in Computer Science. Cham: Springer International Publishing, 2021, pp. 43–63. doi: 10.1007/978-3-030-85616-8_4.

[85] C. Getson and G. Nejat, 'Socially Assistive Robots Helping Older Adults through the Pandemic and Life after COVID-19', *Robotics*, vol. 10, no. 3, Art. no. 3, Sep. 2021, doi: 10.3390/robotics10030106.

[86] I. Mehta, H.-Y. Hsueh, S. Taghipour, W. Li, and S. Saeedi, ‘UV Disinfection Robots: A Review’, *Robot. Auton. Syst.*, vol. 161, p. 104332, Mar. 2023, doi: 10.1016/j.robot.2022.104332.

[87] K. Ruan, Z. Wu, and Q. Xu, ‘Smart Cleaner: A New Autonomous Indoor Disinfection Robot for Combating the COVID-19 Pandemic’, *Robotics*, vol. 10, no. 3, Art. no. 3, Sep. 2021, doi: 10.3390/robotics10030087.

[88] P. Caleb-Solly, S. Dogramadzi, C. A. G. J. Huijnen, and H. van den Heuvel, ‘Exploiting ability for human adaptation to facilitate improved human-robot interaction and acceptance’, *Inf. Soc.*, vol. 34, no. 3, pp. 153–165, May 2018, doi: 10.1080/01972243.2018.1444255.

[89] B. Bruno *et al.*, ‘Knowledge Representation for Culturally Competent Personal Robots: Requirements, Design Principles, Implementation, and Assessment’, *Int. J. Soc. Robot.*, vol. 11, no. 3, pp. 515–538, Jun. 2019, doi: 10.1007/s12369-019-00519-w.

[90] M. M. A. de Graaf, S. B. Allouch, and T. Klamer, ‘Sharing a life with Harvey: Exploring the acceptance of and relationship-building with a social robot’, *Comput. Hum. Behav.*, vol. 43, pp. 1–14, Feb. 2015, doi: 10.1016/j.chb.2014.10.030.

[91] M. Ienca, F. Jotterand, C. Vică, and B. Elger, ‘Social and Assistive Robotics in Dementia Care: Ethical Recommendations for Research and Practice’, *Int. J. Soc. Robot.*, vol. 8, no. 4, pp. 565–573, Aug. 2016, doi: 10.1007/s12369-016-0366-7.

[92] C. Johnston, ‘Ethical Design and Use of Robotic Care of the Elderly’, *J. Bioethical Inq.*, vol. 19, no. 1, pp. 11–14, 2022, doi: 10.1007/s11673-022-10181-z.

[93] R. Leenes, E. Palmerini, B.-J. Koops, A. Bertolini, P. Salvini, and F. Lucivero, ‘Regulatory challenges of robotics: some guidelines for addressing legal and ethical issues’, *Law Innov. Technol.*, vol. 9, no. 1, pp. 1–44, Jan. 2017, doi: 10.1080/17579961.2017.1304921.

[94] Y.-H. Wu, C. Fassert, and A.-S. Rigaud, ‘Designing robots for the elderly: Appearance issue and beyond’, *Arch. Gerontol. Geriatr.*, vol. 54, no. 1, pp. 121–126, Jan. 2012, doi: 10.1016/j.archger.2011.02.003.

[95] A. Sharkey, ‘Robots and human dignity: a consideration of the effects of robot care on the dignity of older people’, *Ethics Inf. Technol.*, vol. 16, no. 1, pp. 63–75, Mar. 2014, doi: 10.1007/s10676-014-9338-5.

[96] A. Sharkey and N. Sharkey, ‘We need to talk about deception in social robotics!’, *Ethics Inf. Technol.*, vol. 23, no. 3, pp. 309–316, Sep. 2021, doi: 10.1007/s10676-020-09573-9.

[97] J.-Y. Sung, L. Guo, R. E. Grinter, and H. I. Christensen, ‘“My Roomba Is Rambo”: Intimate Home Appliances’, in *UbiComp 2007: Ubiquitous Computing*, J. Krumm, G. D. Abowd, A. Seneviratne, and T. Strang, Eds., in Lecture Notes in Computer Science. Berlin, Heidelberg: Springer, 2007, pp. 145–162. doi: 10.1007/978-3-540-74853-3_9.

[98] L. Damiano and P. Dumouchel, ‘Anthropomorphism in Human–Robot Co-evolution’, *Front. Psychol.*, vol. 9, 2018, Accessed: Aug. 02, 2023. [Online]. Available: https://www.frontiersin.org/articles/10.3389/fpsyg.2018.00468

[99] A. Sharkey and N. Sharkey, ‘Children, the Elderly, and Interactive Robots’, *IEEE Robot. Autom. Mag.*, vol. 18, no. 1, pp. 32–38, Mar. 2011, doi: 10.1109/MRA.2010.940151.

[100] C. Lutz, M. Schöttler, and C. P. Hoffmann, ‘The Privacy Implications of Social Robots: Scoping Review and Expert Interviews’, *412-434*, 2019, doi: 10.1177/2050157919843961.

[101] A. Sharkey and N. Sharkey, ‘Granny and the robots: ethical issues in robot care for the elderly’, *Ethics Inf. Technol.*, vol. 14, no. 1, pp. 27–40, Mar. 2012, doi: 10.1007/s10676-010-9234-6.

[102] U. M. Erdem and M. Hasselmo, ‘A goal-directed spatial navigation model using forward trajectory planning based on grid cells’, *Eur. J. Neurosci.*, vol. 35, no. 6, pp. 916–931, Mar. 2012, doi: 10.1111/j.1460-9568.2012.08015.x.

[103] V. Mohan, G. Sandini, and P. Morasso, ‘A neural framework for organization and flexible utilization of episodic memory in cumulatively learning baby humanoids’, *Neural Comput.*, vol. 26, no. 12, pp. 2692–2734, Dec. 2014, doi: 10.1162/NECO_a_00664.

[104] T. Kohonen, ‘The Basic SOM’, in *Self-Organizing Maps*, T. Kohonen, Ed., in Springer Series in Information Sciences. , Berlin, Heidelberg: Springer, 1995, pp. 77–130. doi: 10.1007/978-3-642-97610-0_3.

[105] E. Tulving, ‘Memory and consciousness’, *Can. Psychol. Psychol. Can.*, vol. 26, no. 1, pp. 1–12, 1985, doi: 10.1037/h0080017.

[106] ‘Tulving, E. (2002). Episodic memory: from mind to brain - Google Search’. Accessed: Oct. 04, 2023. [Online]. Available: https://www.google.com/search?q=Tulving%2C+E.+(2002).+Episodic+memory%3A+from+mind+to+brain&sourceid=chrome&ie=UTF-8

[107] B. J. Baars, *A Cognitive Theory of Consciousness*. New York: Cambridge University Press, 1988.

[108] B. J. Baars, N. Geld, and R. Kozma, ‘Global Workspace Theory (GWT) and Prefrontal Cortex: Recent Developments’, *Front. Psychol.*, vol. 12, 2021, Accessed: Oct. 01, 2023. [Online]. Available: https://www.frontiersin.org/articles/10.3389/fpsyg.2021.749868

[109] M. E. Raichle, ‘The Brain’s Default Mode Network’, *Annu. Rev. Neurosci.*, vol. 38, no. 1, pp. 433–447, 2015, doi: 10.1146/annurev-neuro-071013-014030.