Using Pepper as a Domestic Helper
Interim Report
Libby Salter 25276034

Robotics and Artificial Intelligence

Swagat Kumar

CIS3140

3rd March 2025

Contents

Project Background

Over the years there has been an increase in service robots and the variety of tasks they are trusted to undertake. This is seen in a multitude of areas including hospitality, construction, manufacturing and healthcare.

The project aims to develop necessary software capabilities to enable Pepper robot to act as an efficient domestic helper. This would be particularly beneficial to individuals with decreased mobility, giving them more independence. The main focus of the project would allow the Pepper robot to recognise objects using its on-board camera, thereby allowing it pick and place items as required.

The Pepper robot created by SoftBank Robotics (2024) was developed to be a social robot and is given humanoid features to give individuals a sense of safety and trust. SoftBank Robotics describe Pepper as 'friendly and engaging, Pepper creates unique experiences and forms real relationships' (SoftBank Robotics, 2024). Pepper was designed to recognise emotions and carry conversations however there are very minimal solutions available whereby Pepper is performing some sort of physical assistive task.

This limitation is a driving force to explore the undiscovered utilisation of Pepper to perform assistive tasks within domestic settings. To evaluate the implementation of Pepper in this way, the system will undergo a series of experiments testing its robustness and overall effectiveness in completing the task to successfully tidy away objects in its local environment.

To achieve this communication between an external computer or laptop is essential to overcome the dated hardware and software of Pepper. The external computer would take the images from Pepper's cameras to process and perform the object recognition algorithm. Then realise the location of the object relative to Pepper in 3D cartesian space, so that Pepper can localise to collect the object and store it in the correct bin.

The communication between Pepper and ROS can be established in various ways, including ROS (robotic operating system) or sockets. ROS is a 'software ecosystem … facilitating the integration, maintenance, and deployment of new functionalities and hardware from simulations to physical deployment' (Herath, 2022), allowing seamless information passing between devices. Additionally, ROS is equipped with simulation tools including RVIZ, which can be utilised to visualise where the system is locating the object relative to Pepper.

 Sockets are a programming tool which 'allow for inter-process communication (IPC) over networks' (Jennings, 2024). Put simply, sockets allow for different programs or processes to exchange information in real time without interruption. Subsequently this would support the exchange of information between Pepper and the external computer required for this project.

Due to its superior integration and visualisation techniques ROS is the preferred method of communication between Peper and the external computer. However, due to some connection challenges development of the system using sockets will be prioritised unless the challenges can be resolved in a timely manner.

To recognise objects to tidy away in the environment the system will implement an object recognition model provided by MediaPipe trained on the COCO dataset. This model was chosen as 'it is both lightweight and accurate' (Google AI for Developers, 2025) making this model ideal

for this task as it requires minimal computational costs whilst still providing relative accuracy. Additionally, as the COCO dataset consists of 80 labelled objects, modifications can be made so that the model will only identify a select three objects.

Aims and Objectives

The overall aim of the project is to enhance the capabilities of the Pepper robot to become a domestic helper, through the development of software using an external computer. The following objectives have changed from the original plan as the scope of the project has become more focused and specific to the steps necessary to achieve the main aim.

1. Implement MediaPipe tools on an external computer to recognise objects.
   Method: Incremental Methodology
   Data Type: Qualitative
   Milestone: Mid-February

2. Compare and contrast programming tools to transmit camera images from Pepper robot to an external computer.
   Method: Incremental Methodology
   Data Type: Qualitative
   Milestone: Beginning of March

3. Implement and develop the motion commands for the Pepper robot to grab and drop objects.
   Method: Incremental Methodology
   Data Type: Quantitative
   Milestone: End of March

4. To calibrate and visualise the location of the object to the Pepper robot in 3D real world Cartesian Space.
   Method: Incremental Methodology
   Data Type: Quantitative
   Milestone: Mid-April

5. To integrate, test and evaluate the performance of Pepper robot as a domestic helper by tidying away objects in its local environment.
   Method: Incremental Methodology
   Data Type: Quantitative
   Milestone: End of April

Progress Report

Significant progress has been made towards achieving the first and second objectives. The first objective, implementing object recognition using MediaPipe on an external computer, has been successfully completed. Following the guides on the Google AI for Developers, the object recognition model was deployed to identify objects within the camera frame. To streamline the system and reduce complexity, the model was modified detect only three specific objects (TV remotes, forks and bottles) out of the original 80 available in the COCO dataset. Figure 1 illustrates successful identification of these objects.

Figure 1: MediaPipe Object Recognition on External Computer

Progress has also been made toward the second objective, which involves transmitting camera images from the Pepper robot to an external computer. However, this process presented several challenges. First the NAOqi library are only compatible with 32-bit systems, while the external computer operates on a 64-bit. To overcome this a 32-bit conda environment was established, successfully enabling the use of NAOqi tools to communicate with Pepper. The next challenge involved socket implementation issues, whereby only the first camera frame would be displayed, while any subsequent frames failed to load due to premature socket closure. This was resolved through threading implementation, ensuring the socket remained open, facilitating the transmission of frames continuously. As a result, a stable video feed can now be viewed.

With object detection and image transmission established, the next focus is developing motion commands required for Pepper to grasp objects. This involves reducing the Pepper robot's joint stiffness allowing for physical manipulation to determine the movement and positions of each joint, recording them for future execution. Additionally, landmark setting of the object bins and exploration of the environment will aid Pepper's route planning when navigating throughout the environment.

Objective four involves accurately determining the 3D position of objects within Pepper's environment. Using its built-in APIs (application programming interface) two 2D cameras, depth information can be calculated, allowing Pepper to localise objects in Cartesian space. This is a crucial step, enabling the seamless integration between object detection and physical movement. Thus, ensuring that Pepper can navigate, grasp, and place objects correctly

To evaluate Pepper's effectiveness as a domestic helper, all system functionalities must be assimilated and tested under real-world conditions. Pepper's overall effectiveness will be evaluated against numerous metrics, including object detection accuracy, grasping success rate, localisation and navigation precision and overall task completion rate (i.e., number of objects successfully tidied away).

Draft literature review

Over the years, robots have become increasingly integrated into domestic environments, assisting with tasks such as vacuuming and managing smart devices. This project explores the potential of Pepper, a humanoid robot designed by SoftBank Robotics, as an effective domestic helper capable of tidying away clutter. Known for its social and engaging design, Pepper is equipped with features that foster a sense of safety and trust in users (SoftBank Robotics, 2024).

Misaros et al. (2024) explored Pepper's use in an office environment, tasked with receiving and delivering documents upon voice command. In contrast, Zhao et al. (2023) investigated Pepper's role in healthcare, particularly elderly care, where it assisted with more complex, human-centred tasks. Both studies highlight Pepper's ability to enhance productivity and improve user experience by boosting office morale or personalising health guidance. Consequently, both noted limitations in Pepper's physical handling capabilities. Misaros et al. (2024) noted Pepper's inability to traverse multiple floors and handle objects heavier than 300g, whilst Zhao et al. (2023) discussed Pepper's reliance on specific feedback systems in healthcare providing only verbal feedback or assistance. Thus, Pepper is constrained from fulfilling more interactive or tactile roles. These findings suggest that while Pepper holds promise, its practical applications may be constrained by both technical and physical boundaries.

A crucial challenge in Pepper's functionality is its outdated software and hardware, noted by Bauer et al. (2019). The study recommends a combined approach of Pepper's depth readings and deep learning models to improve its 3D perception, which addresses one of the major limitations observed in both the office and healthcare settings. Integration of these advanced technologies could increase Pepper's adaptability to diverse environments, overcoming its current physical and sensory limitations. However, the outdated nature of Pepper's software, particularly the discontinued support for Python 2.7 or later Python versions, remains an issue that could hinder its broader adoption in fast-evolving fields of robotics. Furthermore, these challenges emphasise the importance of addressing hardware and software limitations to improve Pepper's overall functionality and usability.

The assimilation of findings from Misaros et al. (2024), Zhao et al. (2023), and Bauer et al. (2019) illustrates both the promise and challenges of using Pepper as a domestic helper. While studies demonstrate that Pepper can improve efficiency, enhance user interactions, and perform valuable tasks in various settings, limitations in its physical capabilities and outdated technology remain as significant barriers. By combining the strengths and weaknesses across the studies, key areas for development are identified specifically around enhancing Pepper's physical capabilities, improving navigation and depth perception, and updating its software. Addressing these challenges are essential to ensure that Pepper can effectively transition into more dynamic environments, performing task like tidying up clutter.

System Diagrams

To Visualise the system's architecture and requirements, three key diagrams were created. The System architecture design diagram (appendix 1) provides a high-level overview of different aspects of the system and the interactions between them. A block diagram (appendix 2) breaks the system down into modular components, aiding in incremental development by highlighting functionalities can be developed and tested independently without dependency on other steps. The flow diagram in appendix focuses on the process of transmitting camera frames from Pepper to the external computer, where the MediaPipe object recognition algorithm is applied. This offers a snapshot of how the second objective is structured and implemented. By employing a modular approach, these diagrams aid prioritising functionalities, ensuring that different parts of the system can be developed, tested and integrated independently. This minimises bottlenecks and enhances project flexibility.

Project Plan

The original project plan in the Gantt Chart (Figure 2) lacked specific steps, making it difficult to track progress effectively.
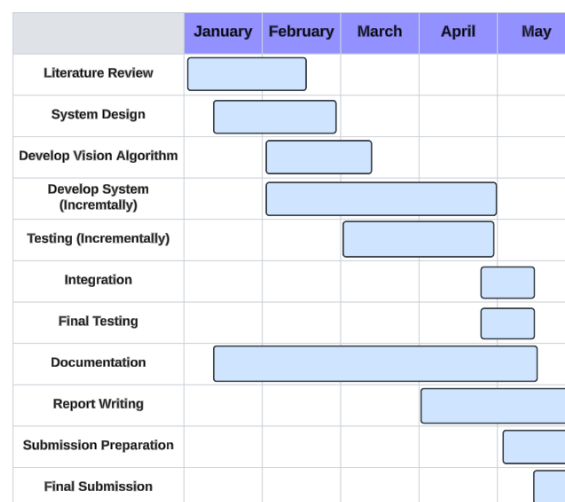

Figure 2: Original Project Plan in a Gantt Chart

To address this the revised project plan (Figure 3) introduces more detailed milestones, revised time allocations and adjusted deadlines.

Gantt chart key

- 🟢 Completed
- 🟡 In Progress
- 🔵 To Be Completed

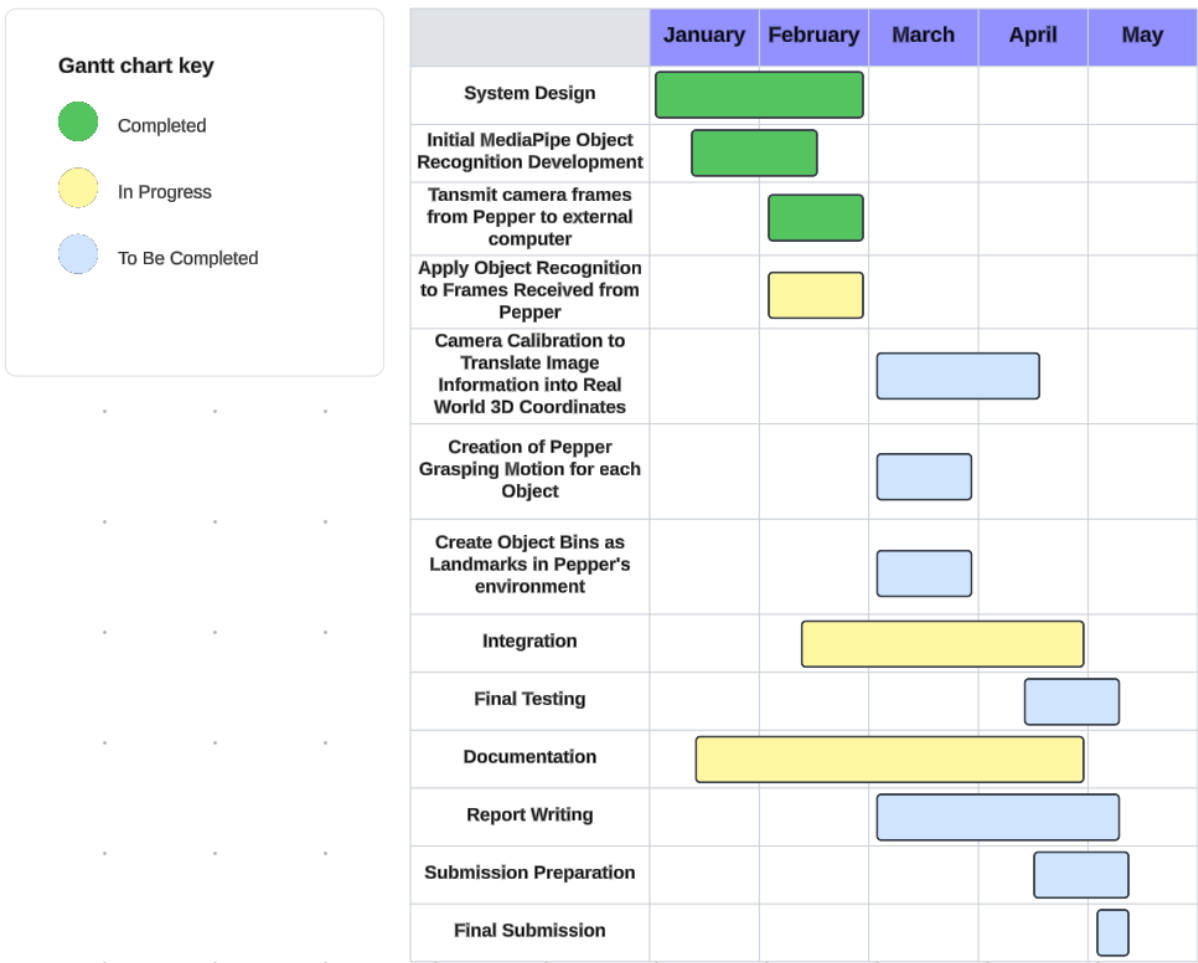|	January	February	March	April	May
System Design	🟢	🟢			
Initial MediaPipe Object Recognition Development	🟢				
Tansmit camera frames from Pepper to external computer		🟢			
Apply Object Recognition to Frames Received from Pepper		🟡			
Camera Calibration to Translate Image Information into Real World 3D Coordinates			🔵		
Creation of Pepper Grasping Motion for each Object			🔵		
Create Object Bins as Landmarks in Pepper's environment			🔵		
Integration		🟡	🟡		
Final Testing				🔵	
Documentation		🟡	🟡		
Report Writing			🔵	🔵	
Submission Preparation				🔵	
Final Submission					🔵

Figure 3: Revised Project Plan in a Gantt Chart

The revised Gantt Chart breaks down development into specific, actionable tasks, ensuring a structured approach. Time allocations have been adjusted to better reflect the complexity of each task, particularly integration, which requires careful blending of different components throughout development. Additionally, the new deadlines align with the final project submission in early May, allowing sufficient time for final testing, submission preparation, and refinement. These revisions create a more structured and realistic roadmap for project completion, ensuring key objectives are met within the given timeframe.

Conclusion

This interim report introduced the project, its motivation, and the key objectives, outlining changes from the initial proposal. It detailed the progress made so far, highlighting completed objectives and the next steps required to achieve the remaining ones. Furthermore, the report evidenced a draft literature review and system design diagrams to support the design process. Finally, the updated project plan was presented, incorporating necessary adjustments to ensure a more achievable outcome.
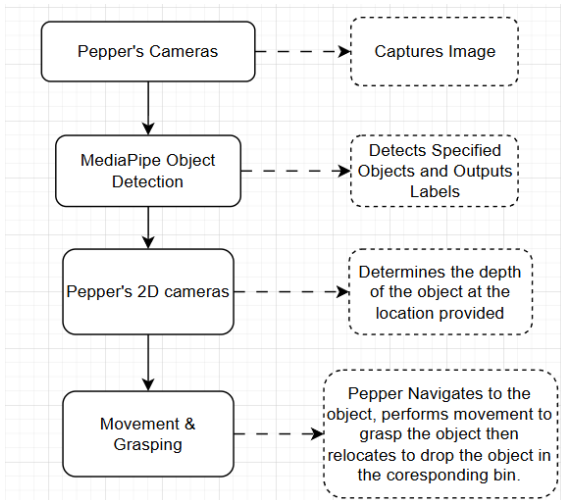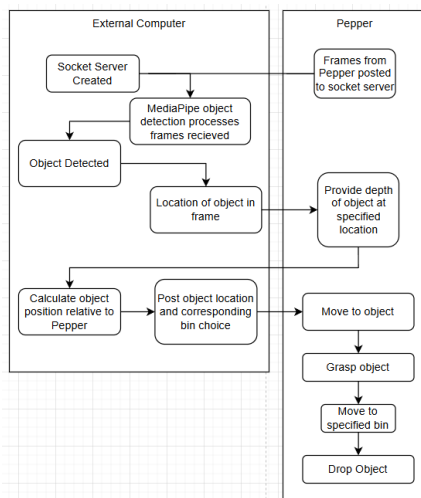
References

BAUER, Z., ESCALONA, F., CRUZ, E., CAZORLA, M. and GOMEZ-DONOSO, F., 2019. Refining the Fusion of Pepper Robot and Estimated Depth Maps Method for Improved 3D Perception. *IEEE Access*. 7. Available from: https://edgehill.on.worldcat.org/oclc/9791048408 http://dx.doi.org/10.1109/ACCESS.2019.2960798.

GOOGLE AI FOR DEVELOPERS., 2025. *Object detection task guide*.

HERATH, D. and ST ONGE, D., 2022. *Foundations of robotics : a multidisciplinary approach with Python and ROS*. Singapore: Springer. Available from: https://edgehill.on.worldcat.org/oclc/1346429930 https://doi.org/10.1007/978-981-19-1983-1.

JENNINGS, N. and REAL PYTHON., 2024. *Socket Programming in Python (Guide)*. Available from: https://realpython.com/python-sockets/ [Accessed 10/02/25].

MISAROȘ, M., STAN, O. P., ENYEDI, S., STAN, A., DONCA, I. and MICLEA, L. C., 2024. A Method for Assessing the Reliability of the Pepper Robot in Handling Office Documents: A Case Study. *Biomimetics (Basel, Switzerland)*. 9 (9). Available from: https://edgehill.on.worldcat.org/oclc/10382907119 http://dx.doi.org/10.3390/biomimetics9090558.

SOFTBANK&NBSP and ROBOTICS. *For better business just add Pepper*. Available from: https://us.softbankrobotics.com/pepper [Accessed 30/12/24].

ZHAO, D., SUN, X., SHAN, B., YANG, Z., YANG, J., LIU, H., JIANG, Y. and HIROSHI, Y., 2023. Research status of elderly-care robots and safe human-robot interaction methods. *Frontiers in Neuroscience*. 17, pp. 1291682. Available from: https://edgehill.on.worldcat.org/oclc/10129100351 http://dx.doi.org/10.3389/fnins.2023.1291682.
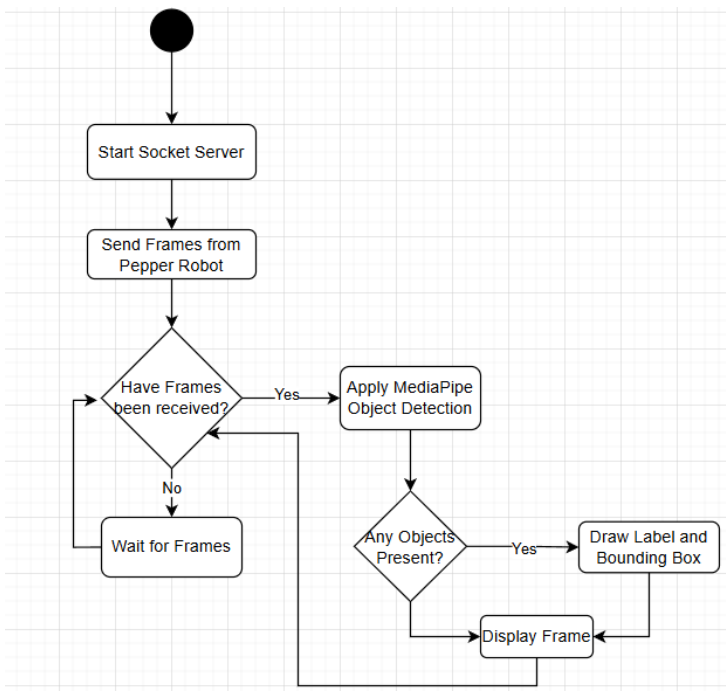
Appendix

Appendix 1: System Architecture Diagram

Appendix 2: Block Diagram

Appendix 3: Flow Diagram for Frame Processing

Appendix 4: List of Figures