MEKATRONIKA
JOURNAL OF MECHATRONICS AND INTELLIGENT MANUFACTURING
E-ISSN: 2637-0883
VOLUME 6, ISSUE 2, JULY, 2024, 1 – 11
DOI: https://doi.org/10.15282/mekatronika.v6i2.10754

RESEARCH ARTICLE

Utilization of Mediapipe Posture Recognition for the Usage in Estimating ASD Children Engagement Interacting with QTrobot

M. F. El-Muhammady, A. S. Ghazali*, M. K. Anwar, H. M. Yusof, and S. N. Sidek

[1] Department of Mechatronics Engineering, International Islamic University Malaysia, 53100 Kuala Lumpur, Malaysia

ABSTRACT - Imitation skills are one of the most important learning skills that are naturally developed by typically developed (TD) children at a young age. Unfortunately, this skill is lacking in special children who are diagnosed with Autism Spectrum Disorder (ASD). To enhance the ASD children's imitation skills for a better social life, this paper proposes to develop and embed a robust gesture recognition system onto a therapy robot called QTrobot. This paper will discuss the utilization of Mediapipe posture recognition as part of estimating the ASD children engagement. MediaPipe posture recognition has the average accuracy of 96% and 60% for both straight facing the camera and 60 degrees away from the camera, respectively. Further enhancements have been done to embed the selected gesture recognition algorithm into QTrobot for developing an efficient Human-Robotic interaction (HRI). Using twenty healthy adult participants, the enhanced algorithm has achieved an average of 94.33% accuracy with an average of 10.5 frame rates per second in recognizing five selected gestures to be imitated by the participants, which are T pose, Strong pose, Super pose, Victory pose, and V pose. Plus, the participants experienced a useful and enjoyable interaction with the robot based on a the 5-point Likert scale of the Technology Acceptance Model (TAM) questionnaire.

ARTICLE HISTORY
Received : 22nd May 2024
Revised : 23rd June 2024
Accepted : 1st Sept 2024
Published : 6th Sept 2024

KEYWORDS
HRI
HCI
ASD
QTrobot
MediaPipe

1.0 INTRODUCTION

Recognition systems are one of the branches of technology that sprout as a result of the rapid development of information technology like Human-Computer Interaction (HCI). With the help of both sensor-based and vision-based technology, the system can be applied in a wide range of applications, namely security, healthcare, and even shopping areas. The recognition system can be divided into three distinct categories namely body gesture recognition, facial recognition, and hand gesture recognition.

Other than that, Human-Robot Interaction (HRI) has become a common scenario in this modern age, especially in industrial and healthcare areas too. Many robots have been custom-built and developed to assist and interact with humans.

Moreover, the combination of both elements, which are the body posture recognition system and HRI, can help enhance the imitation skills of kids with mental impairments like autism. In other words, with the help of HCI and HRI, it can improve the conventional way of doing therapy sessions, especially for Applied Behavioural Analysis (ABA) therapy. Plus, robotic therapy can reduce the dependency on human therapies, and the patient does not need to worry about limited access to therapy centres anymore [1].

The remainder of this paper provides a detailed method of body gesture recognition systems as well as their application in enhancing the imitation skills of special needs children. In addition, this paper will also focus on embedding the developed posture recognition system into a social robot called QTrobot.

2.0 HUMAN-ROBOT INTERACTION (HRI)

Human-robot interaction (HRI) is presently the subject of significant and varied research and development [10]. According to [8], Human-robot interaction is an emerging, multidisciplinary field that includes a branch of Computer Science that links from the field of Artificial Intelligence, primarily in Human-Computer Interaction (HCI), Robotic Engineering, Natural Language Processing, and Computer Vision. It relates to Electrical, Mechanical, Industrial, and Design Engineering as well. In addition, it focuses on the Social Sciences, specifically Human Factors, Psychology, Cognitive Science, Communications, Sociology, and Anthropology. In Humanities, it is also related to Ethology, Ethics, Linguistics, and Philosophy [8].

A system or device is regarded as a robot if it has the following characteristics: sensing, movement, energy, intelligence, and shape [8]. Thus, it can be said that a robot is a programmable device, physically embodied, intelligent, mobile, energy-driven machine with the potential to behave autonomously or be teleoperated in a sensing-capable environment [8].

Meanwhile, the definition of "interaction" in terms of the HRI area refers to the process of collaborating to achieve a common objective. HRI is consequently focused on making the interactions between robots and human beings as natural

***CORRESPONDING AUTHOR | A. S. Ghazali** | ✉ aimighazali@iium.edu.my

as feasible [8]. There are numerous sorts of interactions between humans and robots such as one human-robot team, one human-multiple robot and others as illustrated in Figure 1 [11].

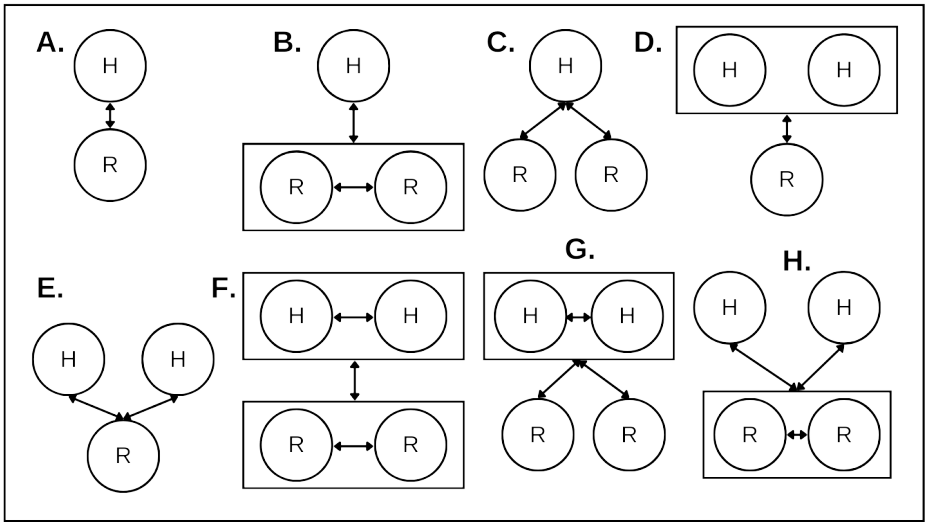

Figure 1. Combinations Involving Humans and Robots Working Alone or Together [11]

Furthermore, the presence of autonomous has aided in raising the productivity and efficiency of the industrial system. Additionally, thanks to industry 4.0, now there are more autonomous and adaptable robots that can work safely side by side with humans, especially on the manufacturing line [2]. According to [4], the percentage of industrial robot installations has increased by 19% every year from 2013 to 2018 and most of the installation happens in 5 countries which are China, Japan, the USA, Korea, and Germany [4].

A pecial category of industrial robots called a "collaborative robot" (Cobot) was created to interact with workers directly [4]. With this technology, Cobots can supplement the advantages of both the human workforce and robot capabilities by offering functionality and effectiveness that can operate in collaboration with human workers to complete the jobs in the production line [4].

3.0 MEDIAPIPE

MediaPipe is Google's open-source, cross-platform tool for constructing perception pipelines that conduct inference on any sensory data [6]. The term "perception pipeline" refers to the sequence of processes from visual input to displayed (output) video. In addition, a perception pipeline can be constructed as a graph of modular components, such as model inference, media processing algorithms, and data transformations [6].

The cross-platform framework enables MediaPipe to run on a variety of platforms, including Desktop/Server, Android, iOS, and embedded devices such as Raspberry Pi and Jetson Nano [5]. In addition, MediaPipe is comprised of a collection of libraries or solutions, pre-trained models, and methodologies for resolving various types of problems. Face detection, position estimation, hand gestures, and holistic analysis are a few examples of the available options [9]. As each of these solutions is based on an open-source framework, customization is possible. According to [12], the MediaPipe can efficiently detect photos or videos because as employs a two-step detector-tracker ML pipeline, which consists of locating the person with Region-of-Interest and detecting within the ROI. Following these operations, the MediaPipe will return the framework-supplied landmarks. As shown in Figure 2.2 the MediaPipe Pose solution uses 33 landmarks to anticipate a person's pose, and the Nose, which is "landmark 0," is typically utilised to find the subject [12].

Therefore, further developments for body gesture recognition algorithms can be expanded based on landmarks [7]. Moreover, MediaPipe employs TensorFlow if a problem requires machine learning to be resolved [9]. Alternatively, recognition conditions also can be done by calculating the angle (Heuristic angle) between the primary body landmark and the condition [12].

There are benefits and also limitations when using the MediaPipe framework. One of the benefits is, MediaPipe has an average accuracy of 95.7% [3]. Plus, MediaPipe also has the highest number of landmarks which is 552 landmarks which include body/foot, hand, and facial landmarks. More landmarks signify more precise results [9]. In addition, MediaPipe can detect the body pose even when some part of the body is hidden [12]. Whereas in terms of frame rate per second (fps), MediaPipe can approach 20 fps on average due to GPU acceleration and multithreading [3].

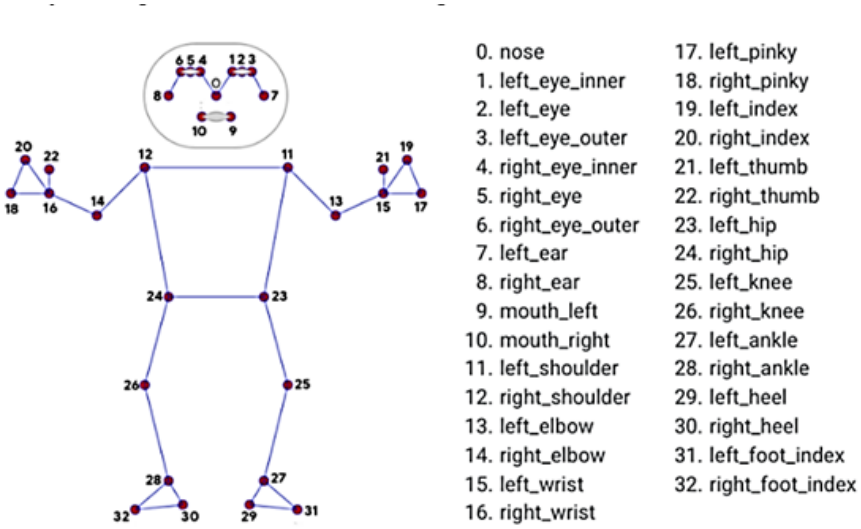

Figure 2. MediaPipe's Body Landmarks.

On the opposite side, the limitations of the MediaPipe framework include the difficulties in determining the depth of Z-axis coordinates with the usage of common camera (instead of a depth camera) [7]. Besides, in case the target is out of an optimum range, the MediaPipe framework cannot accurately track the landmarks [7].

For this study, the MediaPipe parameters are set as per Table 1 with including on and off the static image mode, model complexity, min detection and tracking confidence.

Table 1. MediaPipe parameters.

Parameters	Value	Description
Static Image Mode	Off	To make the solution (Pose Estimation Solution) treat the incoming frames as a video stream, we set it to false.
Min Detection Confidence	0.5 (Default value)	For a detection to be deemed successful, the person-detection model must produce a minimum confidence value between 0 and 1.
Min Tracking Confidence	0.5 (Default Value)	The landmark-tracking model must return a minimum confidence value between 0.0 and 1.0 for the pose landmarks to be deemed successfully tracked; else, person detection will be activated automatically on the following input image. By increasing it, the solution's robustness can be improved at the cost of an increase in latency.

4.0 METHODLOGY AND SYSTEM DESIGN

4.1 Overview

The flow of this study is depicted in the flowchart as shown in Figure 3. According to Article 39 of Law Number 11 of 2019 on the National System of Science and Technology, all research operations must adhere to the code of ethics in the relevant scientific profession or subject. Research Approval For researchers, ethics guides in following the ideals of integrity, honesty, and fairness ("Ethical Clearance", nd). Research Ethics Clearance is a tool used to assess how ethically compliant a research process is. Researchers need approval from the Ethics committee before conducting their research involving humans as their test subjects. This Ethics clearance is essential in order to protect the research subject. This research study has secured approval from the IIUM Research Ethics Committee (IREC).

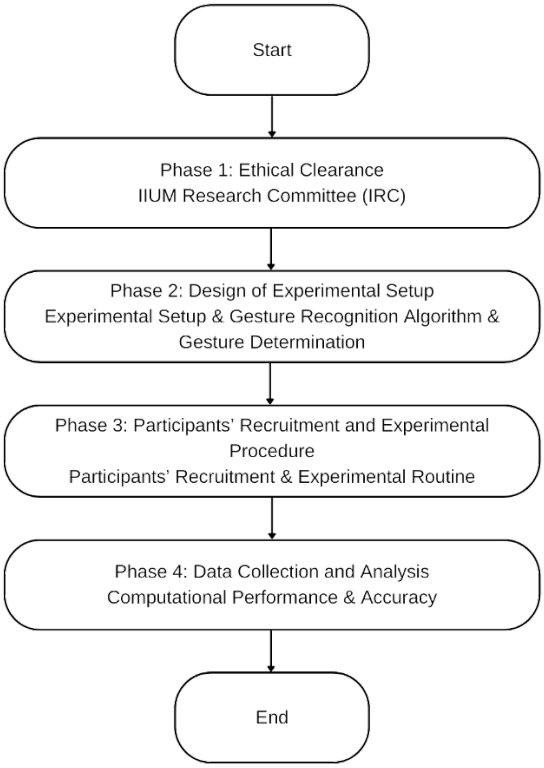

Figure 3. Methodology flowchart

4.2 Predefined Posture

Five gesture that have been selected for this experiment namely the Salute Pose, Strong Pose, T Pose, Waving Pose, and Victory Pose. Figure 4 shows the gestures of these poses that needs to be imitated by the participants. These gestures are chosen with the advice from an occupational therapist (OT) as these gestures can describe the emotion of the perpetrator.

Figure 4. Illustration of selected predefined pose

4.2 Gesture Determination (Angle Calculation and Conditional Statement)

After all the necessary landmarks (upper body landmarks) are obtained, the gestures are recognized by using angle calculation. Firstly, to find an angle, it will associate the coordinate of three related landmarks. For instance, to calculate the angle of the left elbow,the coordinates of three related landmarks which are left shoulder, left elbow, and left wrist are needed. Eventually, there are two ways to calculate the angle between the landmarks.

As shown in the coding below,a function called "calculate_angle" is introduced. As the function's name suggest, it will help to automatically calculate the angle. The parameters that will be passed to the function are the coordinates of the landmarks. Then, the radian angle will be converted into degrees before the function returns this value to the main functions. Importantly, this function will be called every time the code iterates for each frame. Hence, a constant updates on the angles is obtained.

```
def calculate_angle (a,b,c):
    a = np.array(a) #first landmark
    b = np.array(b) #middle landmark
    c = np.array(c) #final landmark

    radians = np.arctan2(c[1] – b[1], c[0] – b[0]) - np.arctan2(a[1] – b[1], a[0]b[0])
    angle = np.abs(radians*180.0/np.pi)

    if angle > 180.0
       angle = 360 – angle

return angle
```

After that, all the angles are combined into a specific conditional statement (if-else) to differentiate the gesture. For this study, only calculation of the angle of neck, left hip, right hip, left shoulder, right shoulder, left elbow, right elbow, left wrist, and right wrist are considered. With these angles, it can be combined to make a logic decision (AND logic).

4.3 Humanoid QTrobot

QTrobot is a commercially accessible humanoid robot designed by LuxAI S.A. It possesses a child-like appearance and serves as an interactive and socially engaging companion. With its broad range of applications, QTrobot is utilized for diverse purposes such as emotional training for children with autism, aiding in post-stroke rehabilitation, and supporting cognitive and physical rehabilitation for the elderly. Figure 5 and Table 2 show the design of QTrobot and its specifications.

QTrobot is equipped with a wide range of capabilities, including the ability to perform various gestures. These gestures are designed to enhance their interactive and expressive qualities, allowing for more engaging and meaningful interactions with users. With these features, the QTrobot can perform customized gestures clearly for people to follow easily. Besides, QTrobot offers a user-friendly interface that simplifies the utilization of essential robot functionalities, also known as the QTrobot Interface. This interface employs a collection of ROS interfaces, enabling convenient access to the robot's features. By utilizing ROS publish/subscribe and Service/Client interfaces, users can interact with the robot's capabilities in both blocking and non-blocking modes.

In addition, the QTrobot interfaces can be accessed by using ROS Publisher/Subscribers to allow non-blocking calls to the interfaces. This is because implementing non-blocking calls with ROS Publisher/Subscribers means that the robot's functionalities can be accessed without causing delays or blocking the execution of other parts of the code. Developers can send commands or data to the robot using ROS Publishers and receive responses or updates through Subscribers, all while allowing the program to continue executing other operations concurrently. As results, this communication mechanism helps to ensure responsiveness and efficiency in the overall system.

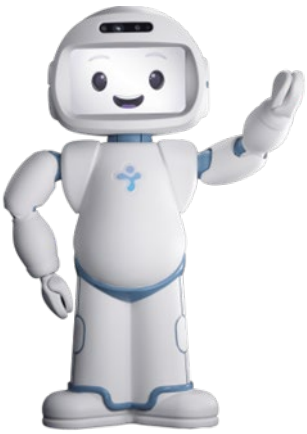

Figure 5. QTrobot.

Table 2. Qtrobot specifications

Main Hardware	Specification
Computing Board	8th Gen quad-core Intel Core i5/i7 Processor
3D Camera	Intel RealSense Depth Camera D435
Microphone	4 High-performance digital microphones

4.4 Research Participants

For system test purposes, there authors used 20 healthy adults. 70% (14 participants) were male and the other 30% (6 participants) were female. The participant's age is between 23 and 24 years old. The duration of the experiment took about 3 to 4 minutes on average.

5.0 EXPERIMENTAL DESIGN AND RESULTS

5.1 Controlled Environment and Experimental Setup

The experiment is conducted under a controlled environment to ensure the collected data is not affected by other extraneous variables that could impact the results of the experiment. The Table 3 below shows the elements that were kept controlled in this experiment.

Table 3. Controlled Parameters.

Controlled Parameters	Value
Temperature	Room Temperature (25 – 28 degC)
Height of the Camera	141 cm
Distance Between the Participants and the QTrobot	149 cm
Video Resolution	480p
Video Colour	RGB
Camera	Logitech C922 Pro Webcam

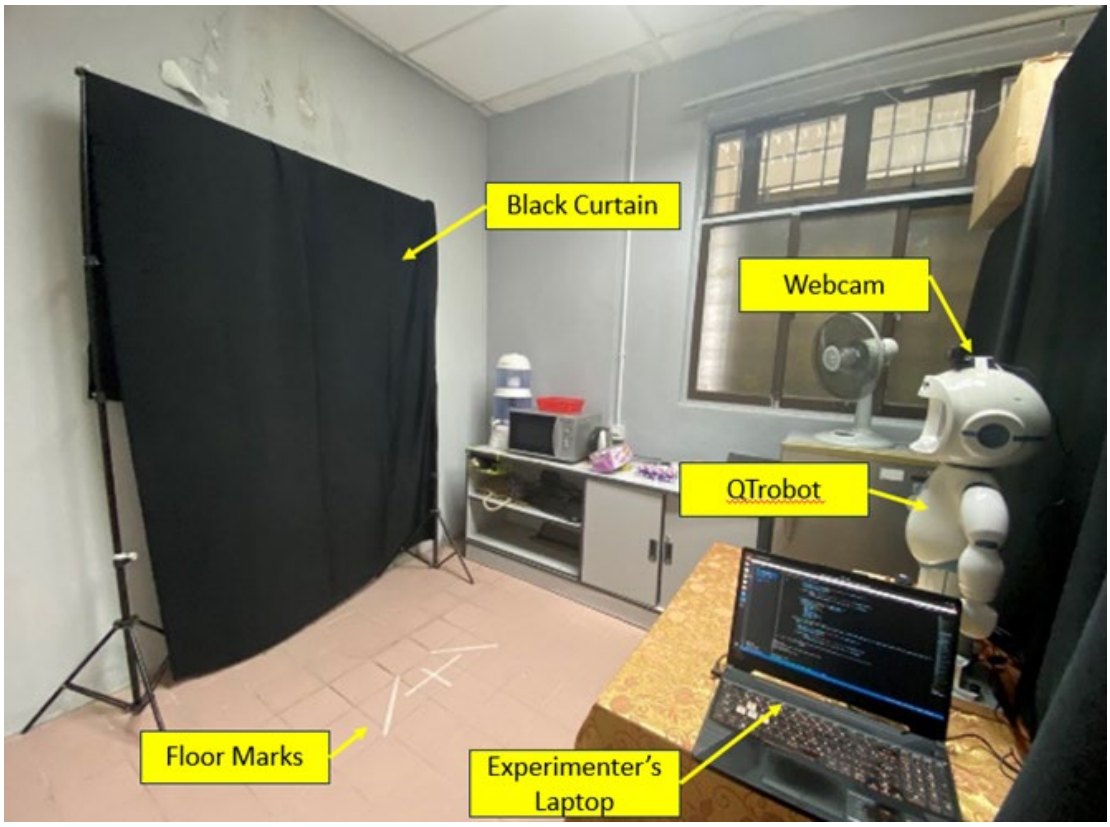

Figure 6. Experimental Setup.

5.2 Experimental Routine

In the experiments, the participants need to imitate the gestures made by QTrobot. Plus, the experiment has two sessions, which are Session 1 (participants facing straight to the camera) and Session 2 (participants facing 45 degrees away from the camera).

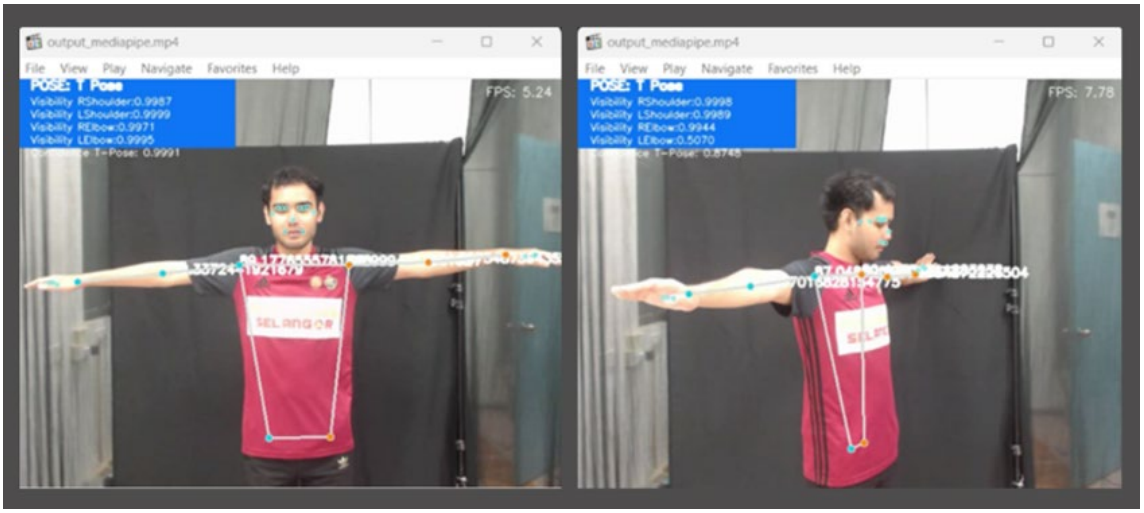

Figure 7. Session 1 and 2 experimental routine.

5.3 Results and Analysis

The collected data will be analysed to determine whether the integration of the QTrobot Interface and Posture Recognition System can work properly together, and whether the participants can do the correct body gestures with QTrobot as their guidance. The Table 4 below shows the type of recorded data from the experiment.

Table 4. Recorded Data

No.	Extracted Data
1.	Experiment Duration
2.	Successful Attempt (Straight)
3.	Successful Attempt (45˚)
4.	Total Attempt
5.	Accuracy (Straight)
6.	Accuracy (45˚)
7.	Accuracy (Overall)
8.	Average FPS
9.	Average Computational Time

Table 5. Participants' Data

ID	Successful Gesture (straight + angled)	Unuccessful Gesture (straight + angled)	Overall Accuracy	Average FPS (fps)	Average Computational Time (s)
P01	10	0	1	8.3809	0.0664
P02	10	1	0.9091	6.7859	0.0636
P03	10	1	0.9091	6.8443	0.0623
P04	10	1	0.9091	8.7795	0.0665
P05	10	2	0.8333	8.3641	0.0627
P06	10	1	0.9091	21.3172	0.0455
P07	10	0	1	22.6505	0.0412
P08	10	0	1	17.0465	0.0461
P09	10	1	0.9091	6.7941	0.0645
P10	10	0	1	8.6167	0.0611
P11	10	0	1	6.7595	0.0643
P12	10	1	0.9091	10.5737	0.0456
P13	10	1	0.9091	13.7068	0.0422
P14	10	0	1	10.5207	0.0586
P15	10	1	0.9091	22.6822	0.0461
P16	10	0	1	5.3782	0.0911
P17	10	0	1	5.958	0.0869
P18	10	3	0.7692	5.8366	0.0944
P19	10	0	1	5.7912	0.0871
P20	10	0	1	8.2847	0.0785

Based on the results, all participants successfully performed all five gestures in both Session 1 (facing QTrobot directly) and Session 2 (facing QTrobot at a 45-degree angle). Notably, half of the participants achieved 100% accuracy, while the remaining half achieved accuracy ranging from 0.7692 to 0.9091. These outcomes indicate the effective functioning of our integrated system in delivering the gestures and accurately detecting the participants' attempts.

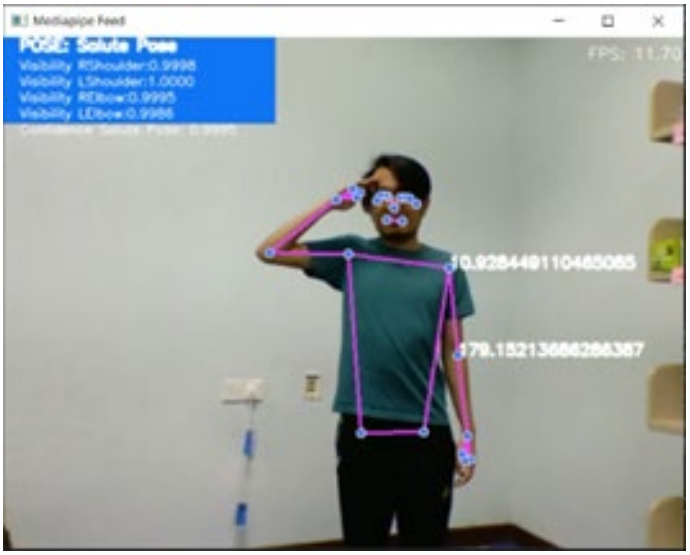

Figure 8. P02 during experiment session

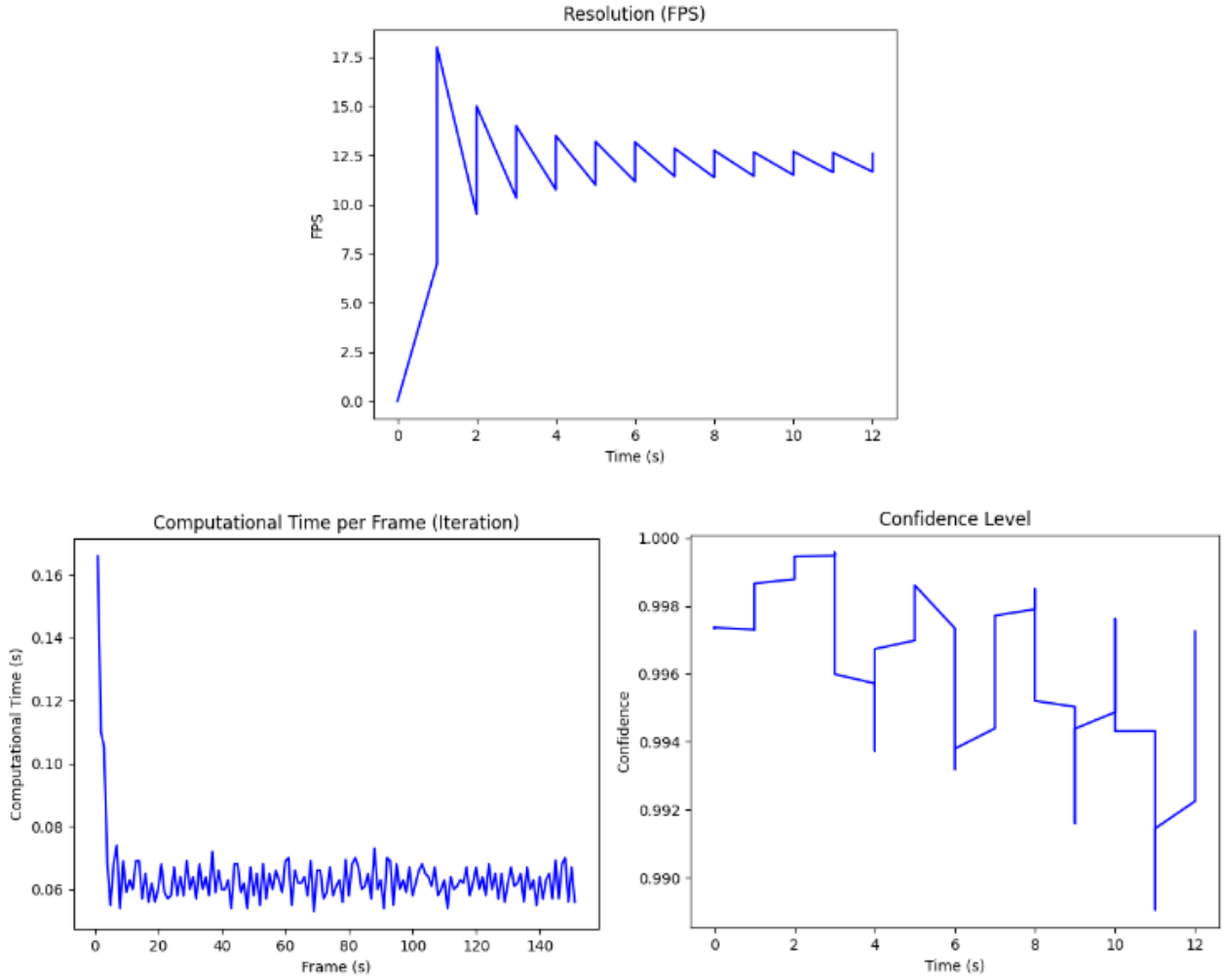

Figure 9. FPS, Iteration, and Confidence chart from Salute Pose of P02

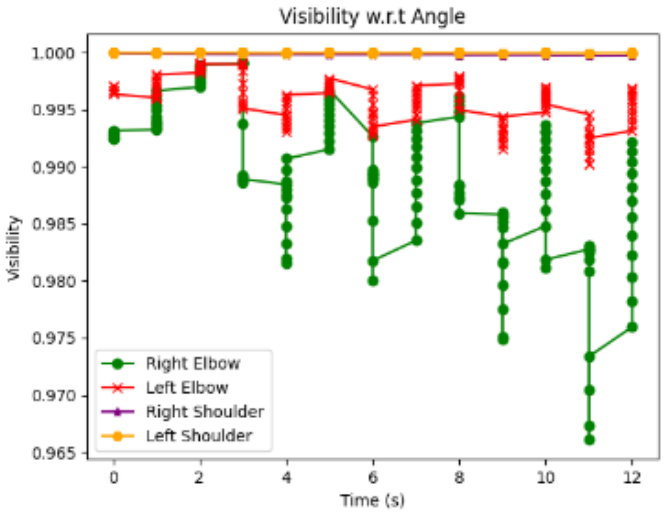

Figure 10. Landmarks visibility

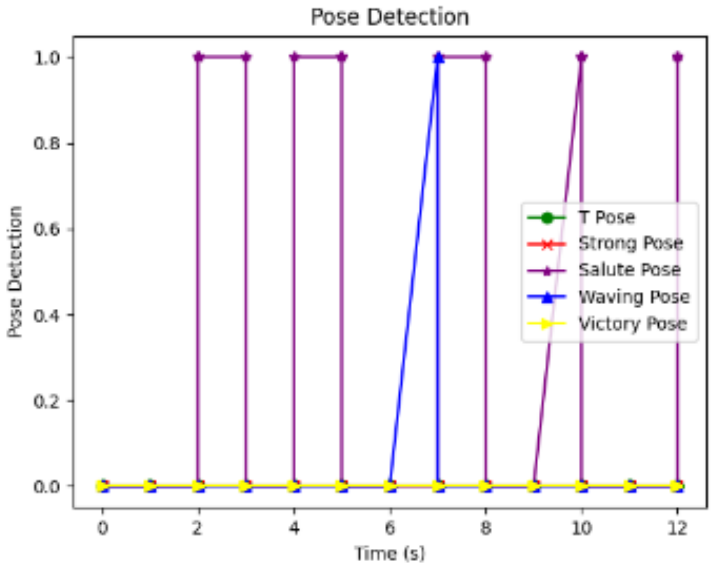

Figure 11. Pose detection confidence

All successful and unsuccessful attempts are calculated based on the mathematical formula mentioned in 4.2 Gesture Determination (Angle Calculation and Conditional Statement) with plus minus 10 degrees. Then, the accuracy is determined by the number of gestures recognized correctly and the confidence level of MediaPipe. In MediaPipe, a confidence level (or confidence score) refers to the probability or certainty associated with the detection or prediction of a particular feature, such as a landmark, pose, or hand gesture. This score is typically a value between 0 and 1, where 0 indicates no confidence and 1 indicates maximum confidence which is 100%. Apart from that, the computational time for processing a task (such as an image operation or video frame processing) is typically calculated by measuring the time taken between the start and end of the task. This calculation can be done using time module in Python, 'getTickCount()', and 'getTickFrequency' fucntions to calculate difference between the start and end tick counts and dividing by the tick frequency to get the elapsed time in seconds as the computational time. Lastly, FPS is calculated by measuring the time it takes to process a single frame and then calculate the number of frames that could be processed in one second. FPS is also calculated using 'getTickCount()' and 'getTickFrequency functions where it calculates the number of clock cycles (ticks) since a reference point and the clock frequency.

Additionally, two reasons can account for the participants not attaining 100% accuracy. Firstly, there is a minor oversight in the coding, whereby participants are unable to input new actions while QTrobot is still in the process of executing its current sequence of actions. Consequently, if the examiner mistimes pressing the keyboard button, QTrobot fails to perform the subsequent specific gestures, leading to participants' attempts being considered unsuccessful. This issue predominantly affected participants who did not achieve 100% accuracy.

Secondly, the current gesture recognition system is unable to identify certain gestures unless the participants mirror them. For instance, in the case of the victory pose, where QTrobot bends its right hand while keeping the left hand straight

(as illustrated in the figure), the system only recognises the pose if participants mirror it, meaning they need to bend their left hand while keeping their right hand straight. Although the examiners provided instructions regarding the mirrored gestures before the experiment, some participants still encountered difficulties in performing the mirrored gestures correctly, resulting in their attempts being deemed unsuccessful.

6.0 LIMITATION

When using the MediaPipe-based pose detection method with ASD kids in real-life therapy settings, there may be some problems and restrictions. These problems can affect how well and reliably the system works, so it needs to be carefully thought through and maybe even fixed. One big problem is that external factors change all the time. Different settings, like therapy rooms, classrooms, or homes, can affect how well the system works. In the real application setting, it's not always possible to set up consistent and controlled settings like the ones used in the first tests with healthy participants. So, for the system to work with backgrounds that are changing and being different, it needs strong algorithms. The different lighting situations are another major problem. The system's accuracy rests on being able to find and follow body landmarks with consistent and enough lighting. Natural light changes, artificial lighting, and shadows can all cause lighting to change a lot in real life, which could hurt the system's performance. It is important to make sure there is enough light or use advanced algorithms that can adjust to different lighting situations. The kids with ASD might not be as compliant or move in different ways than the healthy kids who were used in the first tests. The pose recognition system might have trouble with them because they might move in strange ways, do the same things over and over, or have trouble following directions. The system needs to be strong enough to handle these changes and make sure that detection is still accurate even though the behaviours are different.

7.0 CONCLUSION

This paper reports the first attempt to integrate the MediaPipe Posture Recognition algorithm into a humanoid QTrobot for the usage of estimating the engagement of ASD children during the interaction with the QTrobot. The study reported the average accuracy of 94.33% at the the FPS of 10.5.

One key area of improvement is to develop a fully automated gesture recognition system within QTrobot, building upon the current semi-automated system. This entails automating the sequence of gestures performed by QTrobot and refining the communication cues, starting from the introduction to the closing speech. Additionally, there may be additional features introduced along the development process, a fully autonomous QTrobot system that ensures robustness and efficiency in its operation.

Finally, another aspect for improvement is enhancing the logic for gesture recognition, particularly by enabling QTrobot to recognise mirrored gestures. This improvement will significantly enhance the efficiency and accuracy of the current recognition system.

8.0 ACKNOWLEDGEMENT

The authors extend their heartfelt appreciation to the Ministry of Higher Education Malaysia (MOHE) for their generous financial support, which made this research study possible under the Fundamental Research Grant Scheme (FRGS) [Ref. No FRGS/1/2022/TK07/UIAM/03/5]

9.0 REFERENCES

[1] Alabdulkareem, A.; Alhakbani, N.; Al-Nafjan, A. A Systematic Review of Research on Robot-Assisted Therapy for Children with Autism. Sensors 2022, 22, 944. https://doi.org/10.3390/s22030944

[2] Chiurco, A., Frangella, J., Longo, F., Nicoletti, L., Padovano, A., Solina, V., Mirabelli, G., & Citraro, C. (2022). Real-time Detection of Worker's Emotions for Advanced Human-Robot Interaction during Collaborative Tasks in Smart Factories. Procedia Computer Science, 200(2019), 1875–1884. https://doi.org/10.1016/j.procs.2022.01.388

[3] Datta, A. K., Datta, M., & Banerjee, P. K. (2015). Face detection and recognition techniques. Face Detection and Recognition, Icces, 45–66. https://doi.org/10.1201/b19349-8

[4] Kopp, T., Baumgartner, M., & Kinkel, S. (2020). Success factors for introducing industrial human-robot interaction in practice: an empirically driven framework. International Journal of Advanced Manufacturing Technology, 685–704. https://doi.org/10.1007/s00170-020-06398-0

[5] Kukil. (2022, November 18). Building a Poor Body Posture Detection and Alert System using MediaPipe. LearnOpenCV. Retrieved October 11, 2023, from https://learnopencv.com/building-a-body-posture-analysis-system-using-mediapipe/

[6] Lugaresi, C., Tang, J., Nash, H., McClanahan, C., Uboweja, E., Hays, M., Zhang, F., Chang, C.-L., Yong, M. G., Lee, J., Chang, W.-T., Hua, W., Georg, M., & Grundmann, M. (2019). MediaPipe: A Framework for Building Perception Pipelines. http://arxiv.org/abs/1906.08172

[7] Ma, J., Ma, L., Ruan, W., Chen, H., & Feng, J. (2022). A Wushu Posture Recognition System Based on MediaPipe. 10–13. https://doi.org/10.1109/tcs56119.2022.9918744

[8] Olaronke, I., Oluwaseun, O., & Rhoda, I. (2017). State Of The Art: A Study of Human-Robot Interaction in Healthcare. International Journal of Information Engineering and Electronic Business, 9(3), 43–55. https://doi.org/10.5815/ijieeb.2017.03.06

[9] Savin, A. V., Sablina, V. A., & Nikiforov, M. B. (2021). Comparison of Facial Landmark Detection Methods for Micro-Expressions Analysis. 2021 10th Mediterranean Conference on Embedded Computing, MECO 2021, 7–10. https://doi.org/10.1109/MECO52532.2021.9460191

[10] Sheridan, T. B. (2016). Human-Robot Interaction. Human Factors, 58(4), 525–532. https://doi.org/10.1177/0018720816644364

[11] Yanco, H. A., & Drury, J. (2004). Classifying human-robot interaction: An updated taxonomy. Conference Proceedings - IEEE International Conference on Systems, Man and Cybernetics, 3, 2841–2846. https://doi.org/10.1109/ICSMC.2004.1400763

[12] Zhang. (2022). Application of Google MediaPipe Pose Estimation Using A Single Camera. California State Polytechnic University, Pomona. https://scholarworks.calstate.edu/downloads/n009w777f