Develop and Implementation of Autonomous Vision Based Mobile Robot Following Human

Hairol Nizam Mohd Shah, Mohd Zamzuri Ab Rashid and Tam You Tam

*Universiti Teknikal Malaysia Melaka, Hang Tuah Jaya,
76100 Durian Tunggal, Melaka, Malaysia
hnizam@utem.edu.my*

Abstract

This project related to develop and implementation of autonomous vision based mobile robot following human. Human tracking algorithm will be developed to allow a mobile robot to follow a human. A wireless camera will be used for image capturing, and Matlab will be use to process the image captured, followed by controlling the mobile robot to follow the human. This system will allow the robot to differentiate a human in a picture. The foreground and background will be separated and the foreground is used to determine the object whether it's human or not. Then classification algorithm is applied to find the centroid of the human. This centroid is then compared with the center of the image to get the location of the human with respect to the camera, either at the left or right of the camera. If the human is not in the center of the camera view, then corrective measures is taken so that the human will be in the center of the camera view. Data for the centroid of human is shown through the Graphical User Interface (GUI).

Keywords: *classification algorithms, tracking algorithms, wireless camera, graphical user interface (GUI)*

1. Introduction

Human detection and tracking is currently a hot debated topic in the research of computer vision. The literature of human tracking is too vast to be discussed in detail, therefore only a few of those literatures will be reviewed to determine the best possible human tracking algorithm.

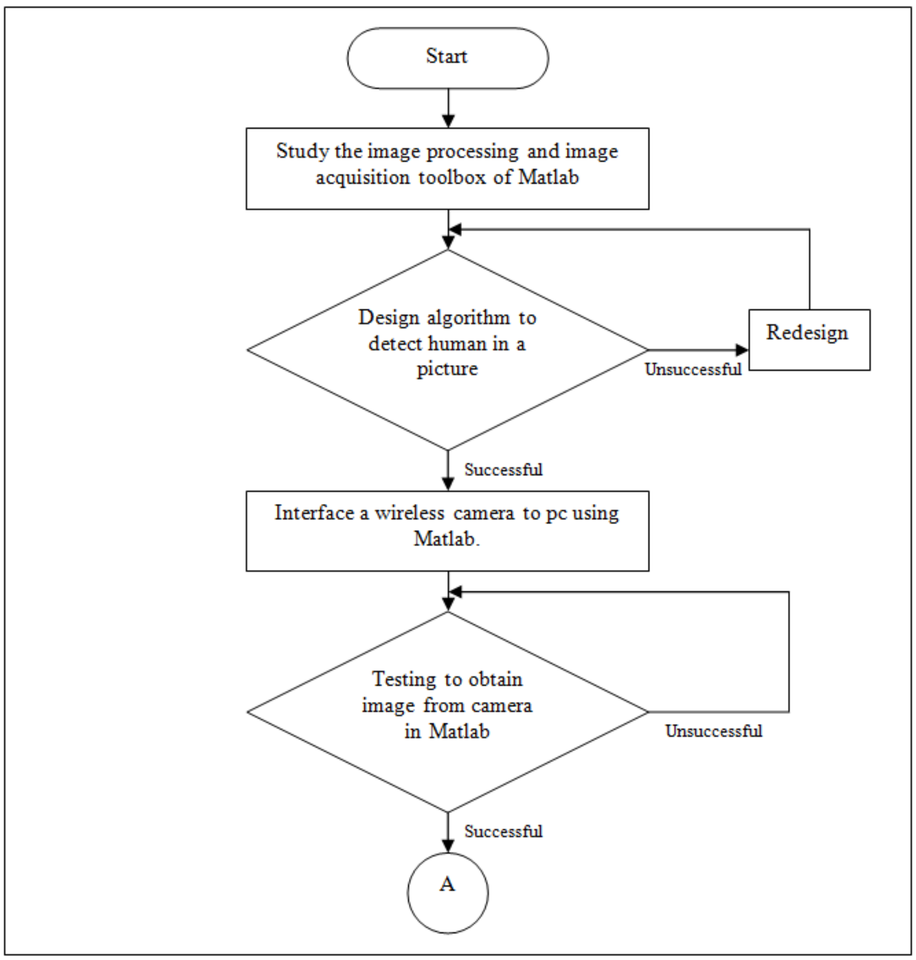
In the previous researchers [1], an algorithm was developed that tracks a human by building a model of appearances from clustering of candidate body segments without comparing with human motion dynamics. Comparing with human motion dynamics is unreliable because humans can move very fast. Frame to frame comparison will show a great deal of acceleration, which will likely be deem as inhuman motion by the comparing algorithm. To solve this, they compare by appearances, because the appearance of a humans does not tend to change between frame, such as color and texture of clothing. In their algorithm, they have managed to include the ability to track multiple humans, and also recover when it loses track, for example when a car passes by and blocks the view to the human.

The combination of the two main categories of human tracking which are shaped- based human detector and motion-based human detector also can be applied [2]. Shaped- based human detectors suffer from 2 drawbacks; high false detection rates and slow processing time. This is caused because it has to process the whole image at different scale to find human figures. Motion-based human detectors has lower false detection rate because they analyze the motion pattern of an object and compare it to human motion dynamic. The weakness is that it has to detect an object first. By combining these two categories, the shaped-based human detector will detect possible human candidates, then tracked for a sufficient amount of time and finally verified with human motion dynamics.

Next researchers [3] used a method called Distance Transform for shaped- based detection on a moving vehicle. This will be crucial in this project for this deals with what happens when the camera gets nearer to an object of interest. The size will gradually increase as the camera gets closer to the object, causing difference when comparing to existing models of the object.

2. Methodology

This chapter discusses the approach taken to complete the project. Figure 1 shows the flow chart of project. Basically, there will be three parts in implementing this project. The first part will be to develop the detection and tracking algorithm. The second part involves interfacing the camera with the program to obtain real time processing. The third part will be troubleshooting the algorithm combined with the mobile robot control software.



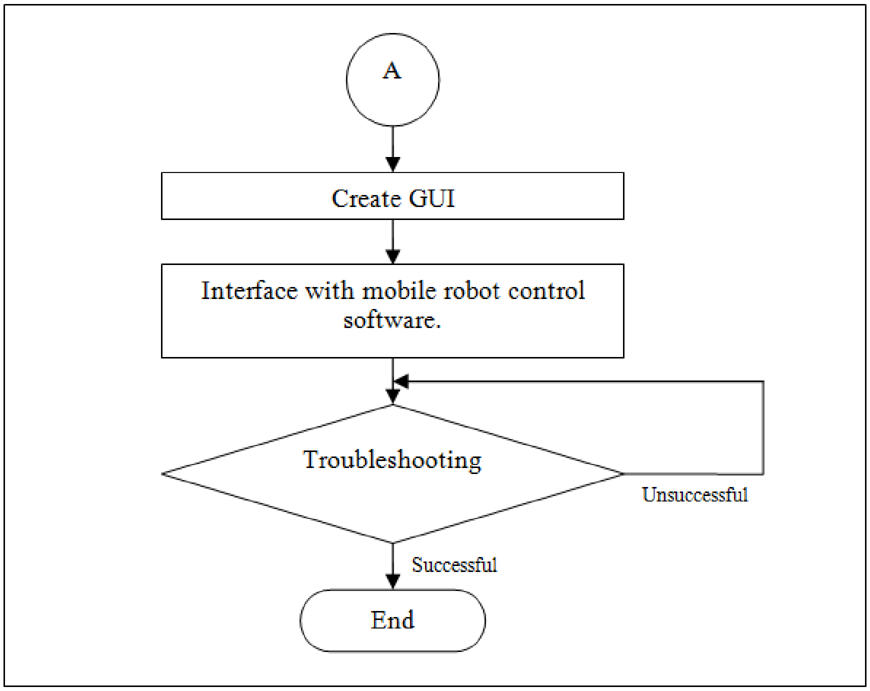


Figure 1. Project Flowchart

Part 1: Develop detection and tracking algorithm

The first step in the algorithm is to recognize objects in an image captured by the camera. Then, the algorithm must determine whether the object segmented is human or not. After determining the object is human, the centroid of the human is determined. This centroid is then compared with the center of the image to get the location of the human with respect to the camera, either at the left or right of the camera. If the human is not in the center of the camera view, then corrective measures is taken so that the human will be in the center of the camera view. Figure 2 shows the flowchart for the detection and tracking algorithm.

Detection is done by using segmentation on the image to try to isolate out the human from the image. Two methods of segmentation were used and compared to determine which better; thresholding and edge detection [4, 5]. After segmentation, the object segmented must be recognized as a human. Two properties are calculate to determine whether the object is human or not; the size of the segmented region and the lowest point of the region. Due to the position of the camera from the human show in Figure 3, it can be estimated that the human will take up a large portion of the image, usually more than half the image.

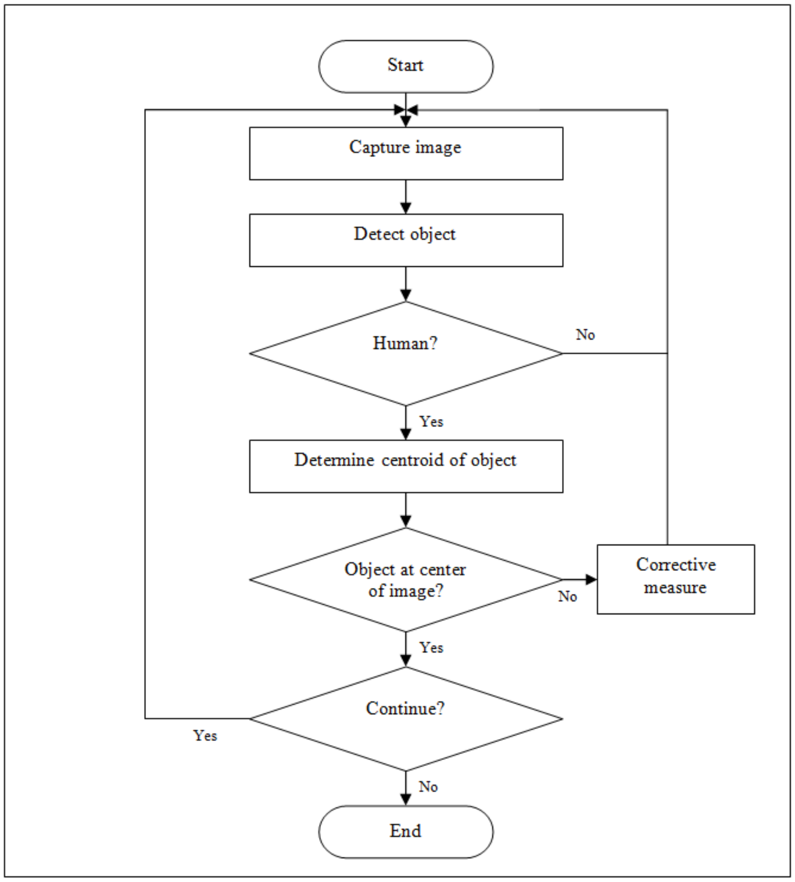


Figure 2. Detection and Tracking Algorithm Flowchart

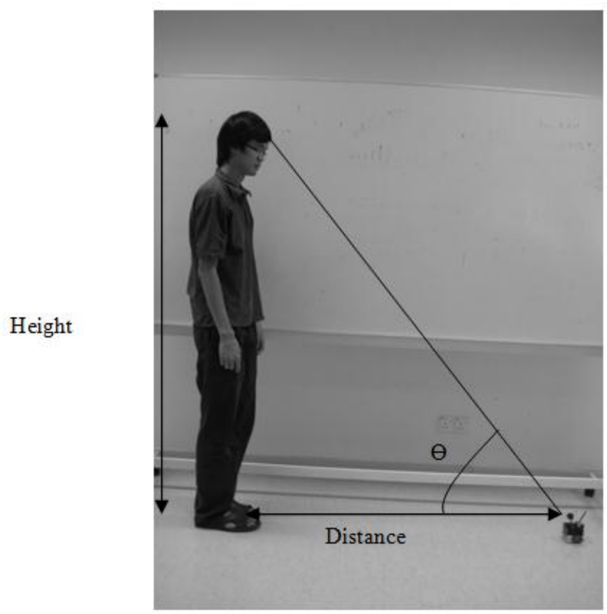


Figure 3. Position of Robot from Human

The next part of the algorithm will be tracking the human. Two properties to selected region determined from recognition by calculated the centroid and the highest point of the region. The centroid is to determine the direction of the human with respect to the camera's view. This same method is used to determine the distance of the human from the camera using the highest point of the segmented region. If the highest point of the region is above the predetermined band, it means that the human is too close to the camera, so the camera must be moved away. Figure 4 show the visual aid for the above explanation.

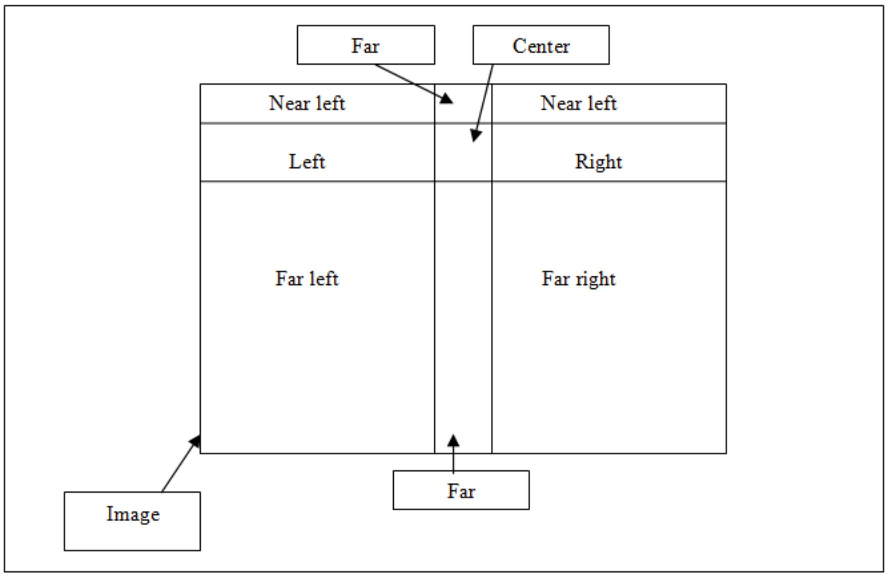


Figure 4. Tracking

Part 2: Interfacing the Camera with the Program

A wireless camera will be used for image acquisition show in Figure 5. Matlab itself contain functions to detect and communicate with any camera and their drivers in Windows through the USB ports. The wireless camera's transmitter uses the normal RG 59 video cable; therefore, a RG 59-to-USB converter is used show in Figure 6. Once interfacing was achieved, a GUI was developed to support a live video feed. Then, the detection and tracking algorithm is applied to the video feed via frames or snapshots obtained from the video feed.

Figure 5. Wireless Camera and Receiver

Figure 6. RG59-to-USB Converter

Part 3: Combining the Algorithm with the Mobile Robot Control

The corrective measure is the mobile robot control software. Once the algorithm is able to detect and track a human, it is merged with the mobile robot control software to enable the mobile robot to position itself so that the human will be centered and distanced in the camera view, in other words making the robot track the human.

3. Result

Human Detection

To test the detection algorithm, two different pictures are used show in Figure 7 with 640 ×480 size image. The pictures contain a human in front of a white background, but with different illumination.

(a) testpic1.jpg (b) testpic2.jpg

Figure 7. Test Pictures

Two methods were tested to segment out the human; edge detection and thresholding.

Edge Detection for Segmentation

For edge detection, four edge detection operators were tested; Canny, Sobel, Prewitt and Roberts. Figure 8 and Figure 9 show the results of using the Roberts operators on the two images and their resized version with two different pictures.

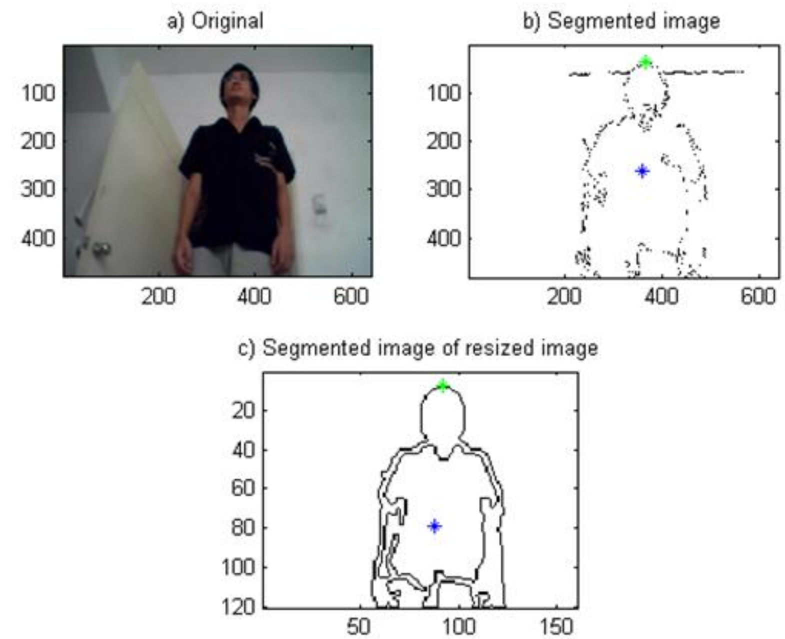


Figure 8. Segmentation of 'testpic1.jpg 'using Roberts Edge Detection

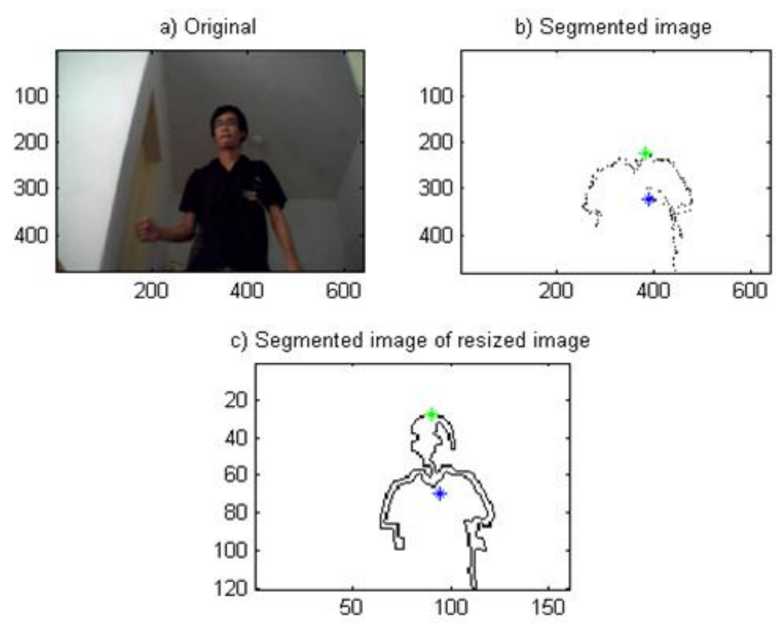


Figure 9. Segmentation of 'testpic2.jpg 'using Roberts Edge Detection

Thresholding for Segmentation

Figure 10-11 show the results using global thresholding. This method did not segment out the human because the background affected the thresholding. Figure 10 had a bright background while Figure 11 had a dark background that caused the calculation of the global threshold value not suitable. Therefore for these two pictures, the threshold value was manually set to 0.2, which in the end allowed the human to be segmented out. Figure 12 show the results of segmentation using threshold = 0.2.

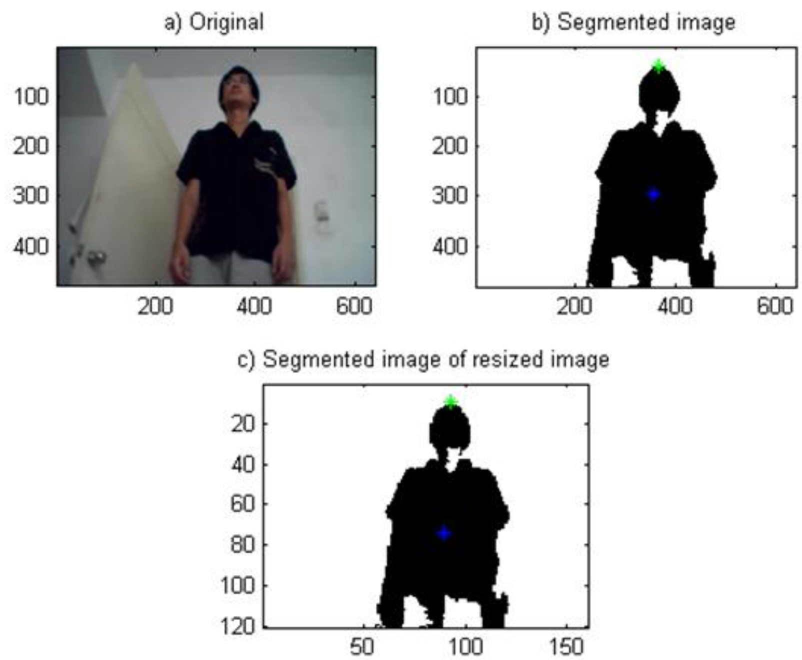


Figure 10. Segmentation of ‘testpic1.jpg’ using Global Threshold

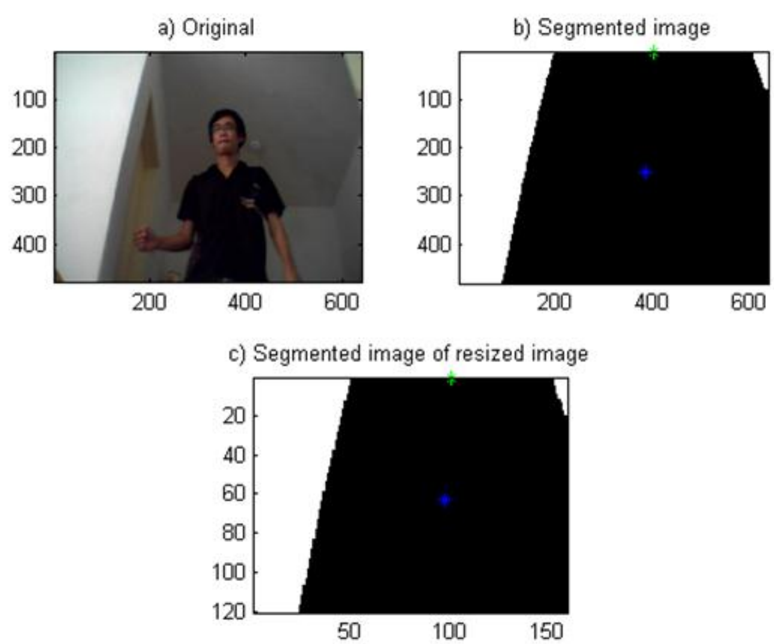


Figure 11. Segmentation of ‘testpic2.jpg’ using Global Threshold

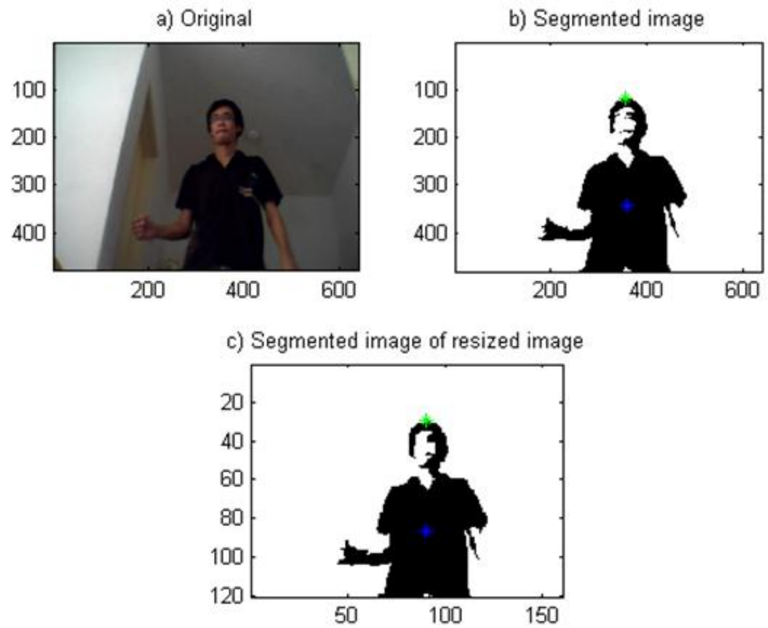


Figure 12. Segmentation of 'testpic2.jpg' using Threshold = 0.2

Tracking and GUI

Graphical User Interface (GUI) was developed to support a live feed from the camera for real time processing and tracking. Figure 13 shows the GUI for real time human detection and tracking.. The centroid of the region and the highest point of the region will display in the GUI. These two points are used for tracking. The centroid is used to determine whether the human is located left or right of the camera and the highest point is used to determine the distance of the human from the camera.

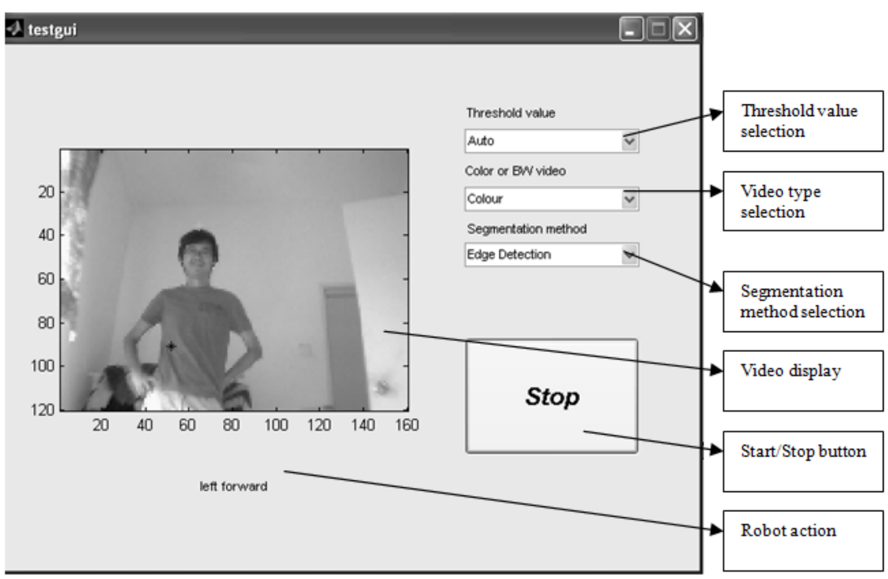


Figure 13. GUI for Real Time Human Detection and Tracking

Completed Human following Robot and its GUI

The completed GUI has additional functions for the mobile robot controls, such as COM port select to select the COM port used by the transmitter for the wireless mobile robot show in Figure 14.

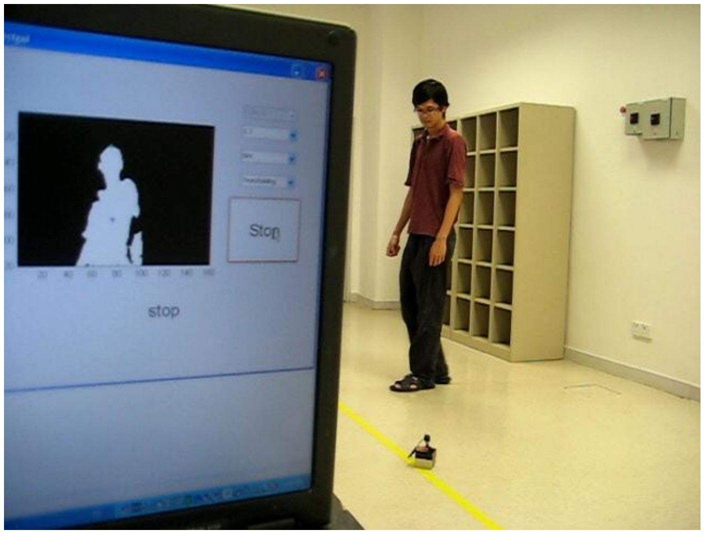


Figure 14. The Robot following a Human and its GUI

4. Discussion

The human following robot performs well in a plain background. But if the background is uneven, there might be a possibility that the robot will not follow the human. This is because of the segmentation of the human from an uneven background or segmentation of other objects. To solve this, the largest segmented object is chosen to be the human because of the position of the camera from the human; normally the human will be the largest object in the picture. But this sometimes produces unwanted results where the algorithm segments out the wall that has a darker shade due to lighting in the room. There are instances that it segments out the ceiling due to lighting in the room. This was solved by calculating the lowers point of the segmented object. Because the camera will be pointed upwards, the segmented human must touch the bottom of the image. Anything else that is not touching the bottom of the picture is not considered for further processing. Also there is a problem that the human moves to fast until he/she moves out of the camera view.

5. Conclusion

A classification and tracking algorithm was developed using Matlab’s image acquisition and image processing library. Real time image processing was achieved but there are still issues such as illumination and rapid movement that cause image processing to be inaccurate when applied onto a mobile robot. A GUI was also developed for the human following robot.

Acknowledgements

This project is sponsored by Short Terms Grant (PJP) through Universiti Teknikal Malaysia Melaka (UTeM)

References

[1] D. Ramanan and D. A. Forsyth, "Finding and tracking people from the bottom up", in Proceedings of Computer Vision and Pattern Recognition (CVPR), Madison, Wisconsin, **(2003)** June.

[2] M. Hussein, W. Abd-Almageed, M. Abdelkader and L. Davis, "Real-Time Human Detection and Tracking from Mobile Vehicle", in Proceedings of the 2007 IEEE Intelligent Transportation Systems Conference, Seattle, WA, USA, **(2007)** September 30 - October 3.

[3] D. M. Gavrila and V. Philomin, "Real-time object detection for smart vehicles", in IEEE International Conference on Computer Vision, **(1999)**.

[4] R. C. Gonzalez and R. E. Woods, "Digital Image Processing Second Edition", Prentice Hall, New Jersey.

[5] T. Svoboda, J. Kybic and V. Hlavac, "Image Processing, Analysis, and Machine Vision", A Matlab Companion, Thompson, **(2001)**.