

IMPROVED METHOD FOR STEREO VISION-BASED HUMAN DETECTION FOR A MOBILE ROBOT FOLLOWING A TARGET PERSON

B. Ali^{1*}, Y. Ayaz, M. Jamil, S.O. Gilani & N. Muhammad

Robotics and Intelligent Systems Engineering Laboratory
Department of Robotics and Artificial Intelligence
School of Mechanical and Manufacturing Engineering
National University of Sciences and Technology, Islamabad, Pakistan
¹badar.ali@smme.edu.pk

ABSTRACT

Interaction between humans and robots is a fundamental need for assistive and service robots. Their ability to detect and track people is a basic requirement for interaction with human beings. This article presents a new approach to human detection and targeted person tracking by a mobile robot. Our work is based on earlier methods that used stereo vision-based tracking linked directly with Hu moment-based detection. The earlier technique was based on the assumption that only one person is present in the environment - the target person - and it was not able to handle more than this one person. In our novel method, we solved this problem by using the Haar-based human detection method, and included a target person selection step before initialising tracking. Furthermore, rather than linking the Kalman filter directly with human detection, we implemented the tracking method before the Kalman filter-based estimation. We used the Pioneer 3AT robot, equipped with stereo camera and sonars, as the test platform.

OPSOMMING

Die interaksie tussen mense en robotte is 'n fundamentele behoefte vir ondersteunende- en dienslewerende robotte. Die robotte se vermoë om mense op te spoor en dan te volg is 'n basiese vereiste vir die interaksie tussen mense en robotte. 'n Nuwe benadering tot die opsporing en geteikende volg van mense deur 'n mobiele robot word bekendgestel. Die navorsing is op vroeër metodes wat van stereovisie gebruik gemaak het, gegrond. Die ouer algoritme is gegrond op die aanname dat net een persoon teenwoordig is in die robot se teiken omgewing - dié algoritme kon nie meer as een persoon in die robot se omgewing hanteer nie. Die nuwe metode los hierdie probleem op deur die Haar-gebaseerde menslike opsporingstegniek te gebruik. Dit sluit 'n teikenpersoonseleksie stap in voor volging begin. Verder word die volgmetode voor die Kalman filter beraming geïmplementeer. Die Pioneer 3AT robot, wat met stereokameras en sonars toegerus is, is as toetsplatform gebruik.

* Corresponding author

1 INTRODUCTION

Human detection and tracking has been a very popular field of research recently. The applications include surveillance, intelligent monitoring systems, human-machine interaction, virtual reality, and many more. Service and assistive robots are gaining importance due to their ability to serve and assist humans in homes, hospitals, and offices. Human-robot interaction is therefore an emerging field as robots become involved in daily life as assistants, team-mates, care-takers, and companions.

A large amount of research has been carried out on human tracking robots. Generally, a laser rangefinder and camera have been the most commonly-used sensors. Jia et al. [1] presented a human detection and tracking approach using stereo vision and extended Kalman filter-based methods. They used Hu moments of head and shoulder contour to recognise humans, and then used the extended Kalman filter (EKF) to predict the location and orientation of the target. Kristou et al. [2] presented their approach for identifying and following the targeted person using a laser rangefinder and an omnidirectional camera. Their method fused the data of both these sensors by identifying the target using a panoramic image, and then tracking that targeted person using the laser rangefinder. Bellotto et al. [3] also presented their multi-sensor fusion system to track people using a laser range sensor and PTZ camera. They used laser scans to detect human legs and video images to detect the human face. They further used an unscented Kalman filter to generate multiple human tracks. People recognition based on their gestures is presented by Patsadu et al. [4].

Recently, Petrović et al. [5] presented a real-time vision-based tracking method using a modified Kalman filter. They used a stereo vision-based detection method to get the features from 2D stereo images, and then reconstructed them into 3D object features to detect human beings. This technique is suitable when only one person is present in the environment - the target person - and it is not able to handle more than one person in the environment.

In our previous works ([6,7]), we had presented novel approaches for detecting and following humans using multi-sensor data fusion of the laser rangefinder and stereo camera. The laser rangefinder (LRF) had been used to gather the width of legs of all the people in the field of view of the LRF, and the stereo camera was used to detect their bodies and faces and extract their width and height. That technique first detected all the faces in the left camera's field of view, and then asked the user to select a target person on the controlling laptop. After the selection of the target person, the system kept the features of height and width of the face, upper body, and legs, and deleted the data of other human beings. Face-based tracking of the selected target was then done using the camshift tracking method. The system had been successfully tested with three persons present in the environment. The limitation of that system was that it was not able to track the target person if that person was not facing the robot. Moreover, camshift was prone to false tracking when lighting conditions were varied.

The work presented here solves the problem of tracking in the presence of multiple persons by using the Haar-based human detection method, which performs feature-based tracking. The target person selection step is also included before initializing the tracking. We included meanshift based tracking before the Kalman filter-based estimation, rather than linking the Kalman filter directly with human detection.

Furthermore, to handle the problem of false detection under different lightning conditions using meanshift tracking, we replaced meanshift tracking with Lucas Kanade (LK) optical flow-based tracking. But we soon found that its limitation was false tracking when there was movement in the background of the target person. We therefore implemented a particle filter instead of the LK optical flow; and this successfully handled both the problems of illumination and tracking the target person in the presence of multiple persons.

In the end, we compared the results of the three tracking techniques, considering the parameters of background illumination, background color effect, background movements, and processing speed. Evaluating these tracking techniques is done in detail and the results are presented here.

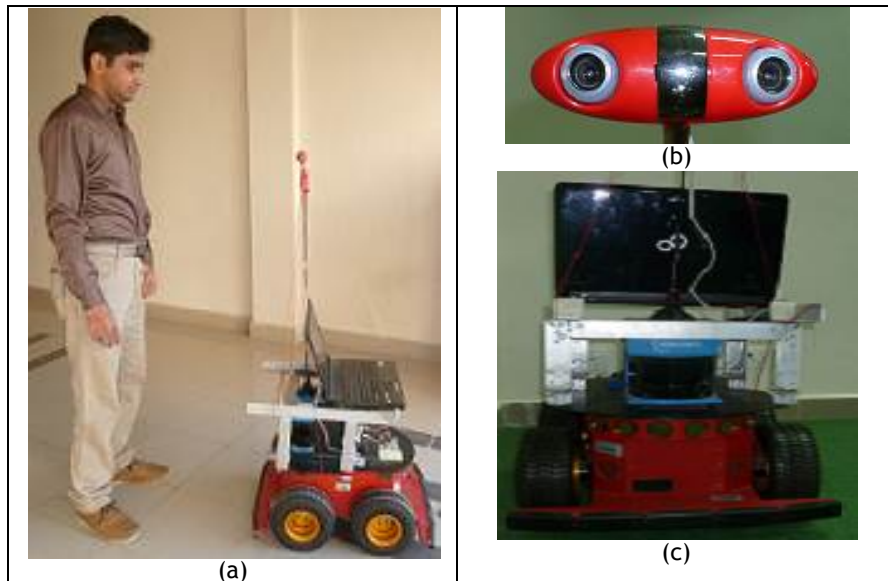


Figure 1: Hardware platform of the system. (a) Person standing in front of robot (b) Stereo USB camera (c) Pioneer 3AT robot with laptop

The article is organized as follows. Section 2 briefly introduces the overall components of the system. Section 3 briefly describes the related techniques. Section 4 presents the proposed method, and explains the details of implementing the tracking techniques. Section 5 presents the experimental results, and the comparison of these results is done in Section 6. Finally, the conclusion and future work is described in Section 7.

2 SYSTEM COMPONENTS

The hardware platform of the proposed system, as shown in Figure 1, includes a Pioneer 3AT mobile robot and stereo camera. The mobile robot is equipped with sixteen sonars (eight at the front and eight at the rear) that are used to avoid colliding with surrounding objects. The stereo camera is mounted at a height of 1,400 mm from the ground to place it nearly at the chest level of an average person. The robot is powered up using its built-in batteries, and the stereo camera is connected directly with the controlling laptop.

In order to support the hardware, two software libraries of C++ are used: Player Stage [16], and Open CV [17]. Player Stage is a C++ platform used to establish server client communication with various mobile robots. This library enables the robot's hardware to be controlled using standard C++ commands. Open CV is a C++ library used to perform all the computer vision-related operations. This library has various built-in functions that make it easy to implement various methods for the live streaming of videos. The proposed control program thus makes use of these two libraries to implement all vision-related techniques, and then to steer the robot towards its target.

3 RELATED WORK AND TECHNIQUES

Recently, a large number of human tracking methods have been introduced, consisting of a large number of vision- and other sensory-data-based techniques. Various techniques are used in this paper, as described below.

3.1 Human detection technique

The proposed scheme uses the object-based detection technique. This approach uses a cascade of Haar-like features to detect a particular object in an image. The method was proposed by Viola et al. [8]. Haar-like features are basically adjacent rectangular regions of pixel intensities at specific positions. These Haar-like features are cascaded to detect a particular object in an image.

A detection window that may be used for human face detection is shown in Figure 2. There are various adjacent rectangular regions in the detection window for all possible features in the human face, and all are combined to get the resemblance of a human face. These rectangular features are searched in the grayscale image, and if their combination is found in any sub-region, then that sub-region are regarded as a face.

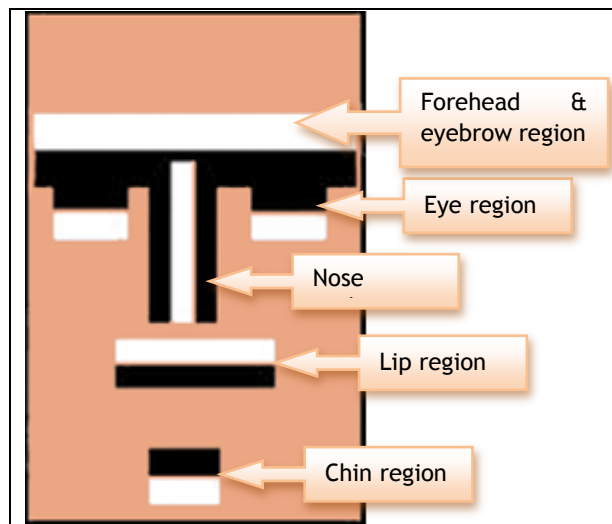


Figure 2: Haar features for face detection

This method creates an integral image of the input image to enable rapid feature detection; it uses a machine learning method called 'AdaBoost' [18] that combines many weak features to create a strong classifier for fast and strong feature detection. The AdaBoost method takes into account many weak classifiers and assigns weight to each of them. The image sub-region is then passed to each classifier filter for classifying as a particular object. Thus each classifier of Haar-like features is cascaded to make a strong classifier for object detection.

3.2 Tracking techniques

The proposed method uses one of three object-tracking techniques based on user input. These techniques are called meanshift, optical flow, and particle filter. A brief description of these techniques is given below.

3.2.1 Meanshift

Various analyses and applications of the meanshift algorithm are discussed in [9, 10]. The idea behind this technique is the iterative calculation of the mean position of all the points (centroid) that lie within the kernel window. The next step is to translate the kernel window towards that centroid. The points are basically the detected features from a region of interest that needs to be tracked. In the proposed work, it is actually the histogram of the Hue component of the HSV image. Figure 3 shows the meanshift vector of a region of interest.

Camshift (continuously adaptive meanshift) [19] is a variant of meanshift that can handle dynamically changing distribution in a video. Camshift handles dynamic changes by

readjusting the search window size for the next frame based on the zeroth moment of the previous frame's distribution. This allows the algorithm to forestall object movement to track the object in the next frame. Even during quick movements of the target, Camshift is still able to track correctly.

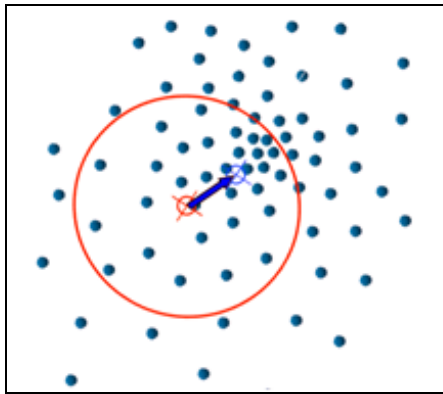


Figure 3: Meanshift clustering-based tracking

3.2.2 Optical flow

Optical flow is the pattern of the apparent motion of objects in a video due to the relative motion between the camera and the scene. Actually, the method tries to compute the motion between two image frames taken one after the other. The optical flow technique that has been used in this paper is based on Kanade tracking [11].

The L-K algorithm for tracking a template $T(x)$ in an image from time $t = n$ to time $t = n+1$, is the extraction of template $T(x)$ (5×5 window) from the image at $t = n$ in the image at $t = n+1$. This method assumes that there is a constant flow in the neighbourhood of the pixel. It then solves the optical flow equations of all the pixels of neighbourhood through the least square method, followed by the combination of the information from the neighbouring pixels to resolve the ambiguity of the optical flow equation. The basic optical flow equation is given as:

$$I(x, y, t) = I(x + \Delta x, y + \Delta y, t + \Delta t). \quad (1)$$

where (x, y) are the points in the image at time t with the intensity $I(x, y, t)$, and $(x + \Delta x, y + \Delta y)$ are points at time $t + \Delta t$ with the intensity $I(x + \Delta x, y + \Delta y, t + \Delta t)$.

3.2.3 Particle filter

The third tracking technique that has been implemented is particle filter-based tracking. The theory and applications of a particle filter are discussed in [12].

A particle filter is the non-parametric sampling-based Bayesian filter that estimates the position by a finite number of parameters. The particle filter uses the random sampling method to model the posterior probability. It uses N weighted discrete particles to estimate the probability by observing the prior data. Each particle has a state vector x as well as its weight w . Resampling and the posterior approximation are then done.

The particle filter method [13] can be divided into four steps: initialization, sampling, estimation, and selection. During sampling, each particle is taken from the distribution, and the weights are calculated based on the probability in the distribution of each particle. Then the posterior density of each particle is approximated, based on its weight. Afterwards, particles are selected according to their approximated posterior density from the previous step to get uniform weight distribution.

4 ADOPTED HUMAN DETECTION AND TRACKING METHOD

In this section, the proposed method for human detection and target person tracking is explained. An overview of the method is given in the flowchart of Figure 4.

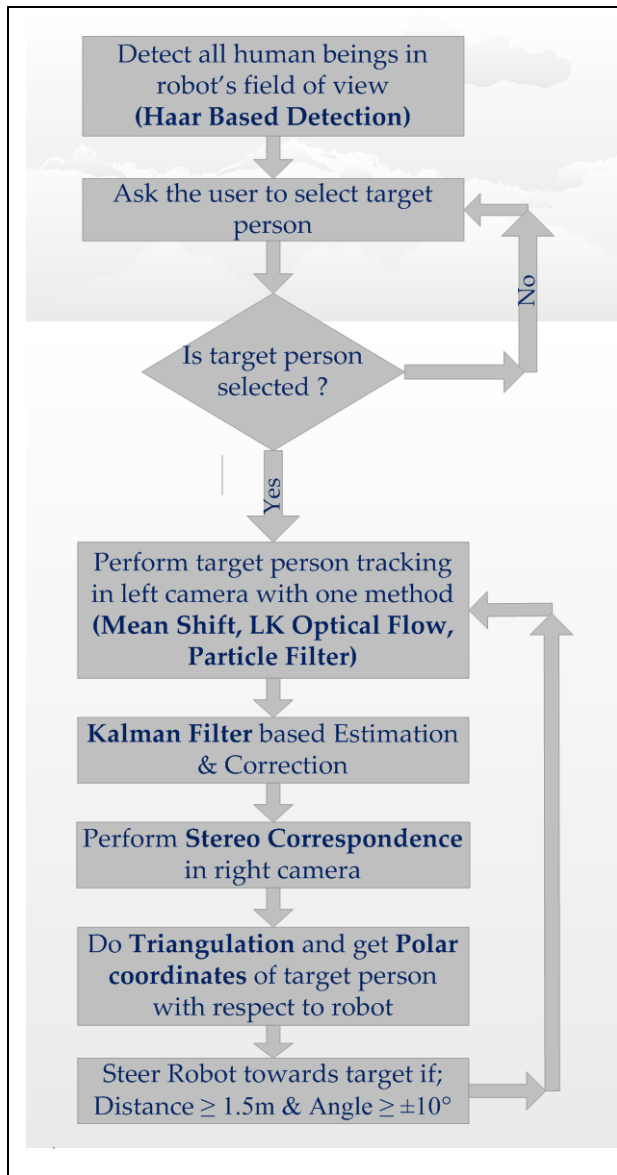


Figure 4: Adopted method for Target Person Following by Mobile Robot

The proposed method initially looks for all the humans in front of the robot. This is done by using Haar feature-based detection. When humans are detected, the control program asks the operator to select one target person; the robot then starts following the target by maintaining a distance of 1,500 mm and at an angle of $\pm 10^\circ$ from him. The tracking is done using the three tracking techniques one by one: meanshift, LK optical flow, and particle filter. These tracking techniques are also implemented independently to enable comparisons between each of them. Furthermore, these techniques are facilitated with the Kalman filter-based estimation to assist more accurate localization of the targeted person with respect to the robot. Afterwards, stereo correspondence is done using template

matching to get the position of the target in the right camera. Using linear triangulation, the position of the target in space is then calculated; this is used to steer the robot towards the target. With all the computation on a single laptop, the position data is sent to the robot's computer, which actuates the robot toward the calculated position.

The implemented methodology of all the processes involved during target selection and tracking is explained below.

4.1 Human detection

As mentioned earlier, the proposed scheme uses Haar-like features for human detection. To implement Haar-based object detection in Open CV, the classifier is trained with a large number of samples of positive and negative examples of the target objects. Then it can be applied to any image to detect that target. This is done by moving the classifier window to the whole image to detect a particular object. A trained set of classifiers is therefore employed to detect the human upper body and face [14].

To perform people detection, only the left camera is used. Initially, only upper body detection is done using a trained set of classifiers; and if an upper body is detected, the faces are searched inside the detected bodies on the input stream of images. This check of face detection inside the detected upper body is done only to minimize the false detection of bodies. The algorithm of this scheme of detection is shown in Figure 5.

4.2 Target person selection

After human detection, the next step is the selection of one target person out of all those detected. All the detected humans are marked with a serial number that starts from the left-most detected person. The control program then asks the operator to enter the serial number of one target person from all the detected humans so that the tracking process can start.

4.3 Target person tracking

As mentioned earlier, three tracking techniques have been used in this work. These methods are implemented independently so that each can be compared with the others. The implementation methodology of each of them is described below.

4.3.1 *Meanshift based tracking*

Meanshift based tracking was the first of the three tracking methods implemented in Open CV. This method requires the initial position of the object in order to track it. Therefore input to meanshift is the location of the selected target person during the target selection stage. The algorithm for the meanshift is explained in Figure 6.

In the first step, the algorithm takes the back projection of the initial position of the target person's face / body. Then it finds the 2D color probability distribution of the target area; and the target center is found by computing the mean of that 2D color probability distribution. After finding the target center, the search window is moved towards that mass center. This position is used to track the target iteratively in the stream of images. The search window size and orientation does not change in this method of tracking.

4.3.2 *Lucas Kanade optical flow-based tracking*

In order to perform LK optical flow-based tracking, the image is converted into grayscale and twelve points - mostly at the contours of the selected target person's upper body, face and shirt - are selected. After storing these points in a vector, iterative pyramid-based optical flow is used to find the corresponding features in the next frame. Then a bounding box - a rectangle containing all the features having the flow - is generated.

This whole implementation is summed up in Figure 7.

- a. **Classifiers initialisation**
 - i. Load Body Classifier: Cl_{body}
 - ii. Load Face Classifier: Cl_{face}
- b. **Image initialisation**
 - i. Get input image: I_{in}
 - ii. $I_{gray} = Convert(I_{in}, grayscale)$
 - iii. $I_{gray} = hist_equalise(I_{gray})$
- c. **Body detection**
 - i. $vec\ det_body = detect_multiscale(I_{gray}, Cl_{body})$
- d. **Face detection inside bodies**
 - i. $vec\ det_face = detect_multiscale(I_{gray}, Cl_{face})$
- e. **Body-face check & data storing**
 - i. *for all* (det_body)
 - if* (det_face \triangleright det_body)
 - body [i] = det_body
 - face [i] = det_face
 - next* i
- f. **Target person selection**
 - i. *if* (selection_key == Pressed)
 - input (k)
 - target_id = k
 - else
 - ii. goto (Step. c)

Figure 5: Implemented algorithm for human detection and target person selection

- Target initialisation**
 - i. $g = get\ target_ID$
 - ii. $roi = image(target.rectangle)$
- a. **Calculation of probability distribution**
 - i. HSV = $rgb2hsv(roi)$
 - ii. Hist = $imhist(HSV:1)$
 - iii. Distribution = $histogram_backproject(Hist)$
- b. **Meanshift calculation**
 - i. Mean_Calculation = $\Sigma(Distribution) / n$
 - ii. Move_Centre to Mean_Calculation
- c. **Camshift**
 - i. Recalculate Area(Densed_Distribution)
 - ii. Adjust Rectangle (Area)
- d. **Target correction & estimation**
 - i. $K = Kalman_Predict(pre_state, post_state)$
 - ii. Target = $Kalman_correct(K, Move_Centre)$
- e. **Target person following**
 - i. Calculate Position 'P' in 3D Space using Target
 - ii. Robot Navigation towards 'P'

Figure 6: Implemented meanshift algorithm

- a. **Tracking initialisation**
 - i. $g = get\ target_ID$
- b. **Points specification**
 - i. $I_{gray} = Convert(I_{in}, grayscale)$
 - ii. $P_{(1-12)}^n = Locations[body(g) \& face(g)]$
- c. **Optical flow**
 - i. $P_{(1-12)}^{n+1} = apply_lk_flow_equation(P_{(1-12)}^n)$
 - ii. $op_flow = rectangle[P_{(top_left)}, P_{(bottom_right)}]$
- d. **Target correction & estimation**
 - i. $K = Kalman_Predict(pre_state, post_state)$
 - ii. Target = $Kalman_correct(K, op_flow)$
- e. **Target person following**
 - i. Calculate Position 'P' in 3D Space using Target
 - ii. Robot Navigation towards 'P'

Figure 7: Implemented LK optical flow shift algorithm

4.3.3 Particle filter-based tracking

Particle filter is a recursive Bayesian filter-based tracking method that is non-parametric. After the selection of target person, sampling of selected area is done by constructing a motion model. These samples are drawn on the basis of proportionality of Gaussian weight at any location inside the region of interest. Weights are then calculated for each sample, based on the similarity function of the correlation of histograms for RGB channels of the selected target area. The next step is resampling according to the calculated weights. Finally, the bounding window is moved towards the cumulative distribution of weights of the latest samples.

The implemented algorithm for particle filter-based tracking is shown in Figure 8.

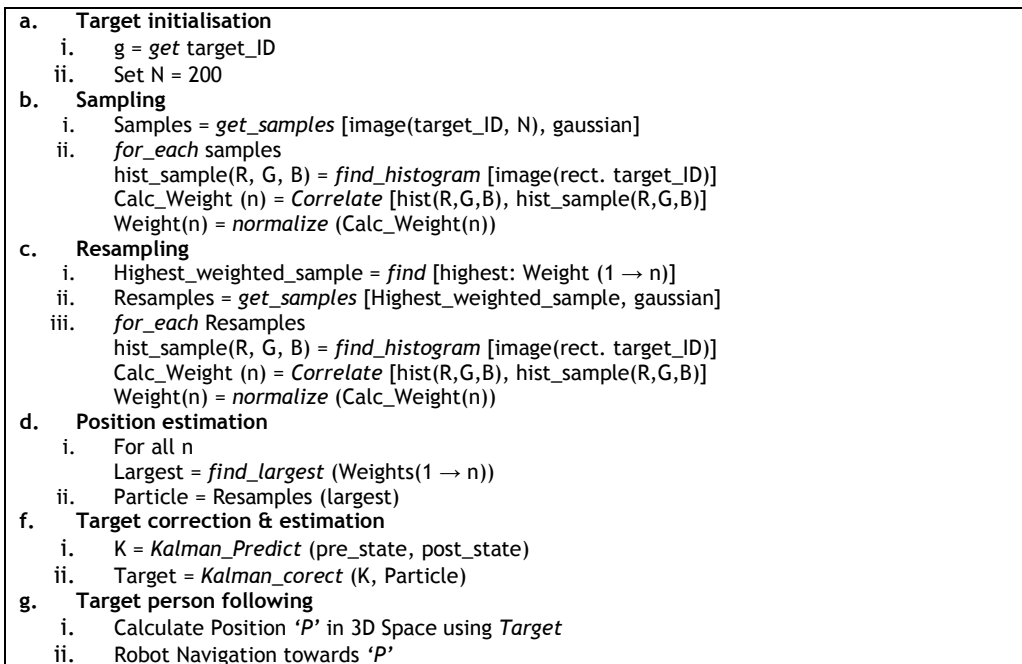


Figure 8: Implemented particle filter algorithm

4.4 Kalman filter-based estimation & correction

A Kalman filter is an optimal estimator that minimizes the mean square error of the estimated parameters. It has two basic parts: prediction and correction. The X and Y coordinate of the tracked target from the previous step is sent to the first step of the Kalman filter - i.e., the prediction part. The second step is the measurement correction based on the distance and velocity model. This step is implemented to localize the target person in the left camera more accurately.

4.5 Stereo correspondence

Stereo correspondence is a technique to find the correspondence between two images in the left and right cameras of a stereo system. This is done by using the iterative location of the target person in the left camera to find the corresponding location of the target in the right camera. Thus the template of the target person from the result of the Kalman filter corrected position in the left camera image is searched in the right camera image to find the best match for its corresponding location. The normalized correlation method is used for searching. The template window is moved across the complete image to find the best match. Some other techniques can also be used - squared difference, normalized squared difference, cross correlation, normalized cross correlation, and correlation coefficient. The results are shown in the right camera windows in the results section.

4.6 Robot following target person

After finding the stereo correspondence for the target person, the next step is to get the exact location of the target person with respect to the robot's origin. Linear triangulation [15] is used for this purpose. It requires the camera projection matrix and the location of the corresponding point in both cameras. As shown in Figure 9, P and P' are the 3×4 camera projection matrix; and z and z' are the points to be located in the left and right camera images; then Z is the 3D location of the point with respect to the camera/robot's origin.

We have used the center point of the target person in both cameras to get the exact location with respect to the robot. This 3D point is used to get the angle and distance of the target from the robot, and the robot is then steered to maintain the distance of 1.5m and the angle of $\pm 10^\circ$ from the target. Furthermore, sonars are used to avoid colliding with other objects in the environment, as this would stop the robot and generate an error message. This can be further used to perform motion planning to avoid static or dynamic obstacles in future.

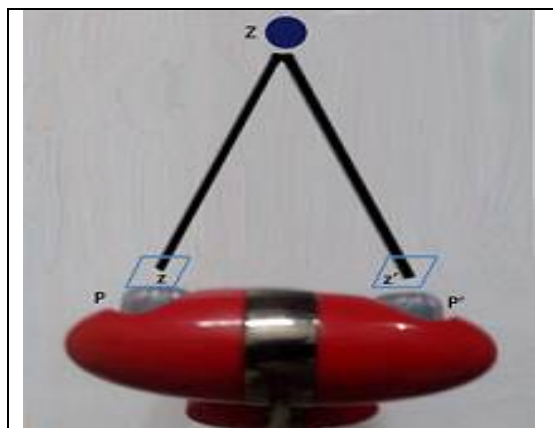


Figure 9: Stereo camera-based triangulation

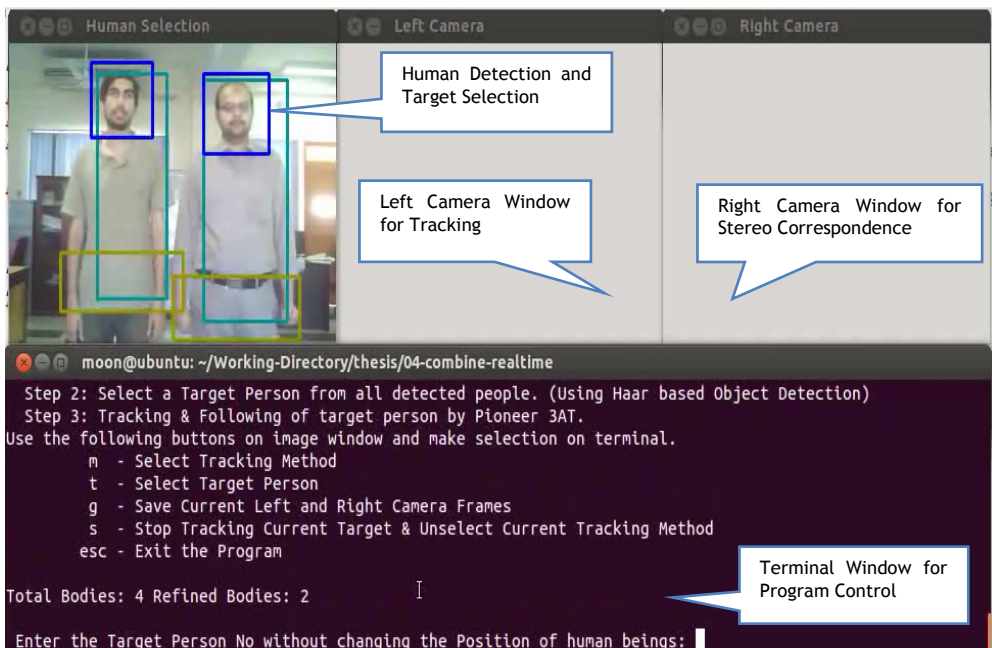


Figure 10: Initialization window for the robot's screen, with labeling

5 EXPERIMENTAL RESULTS

In this section, the results of the proposed human detection and tracking method are presented. The whole test is done using more than ten different benchmarks. Here we present the tracking results on three benchmarks. The human detection and tracking results are presented below.

5.1 Human detection & target person selection results

Figure 10 shows the laptop's screen (initialisation window) with labelling of the left and right images and the terminal window. Figure 11 presents the three benchmarks (indoor & outdoor) for which the results are presented.

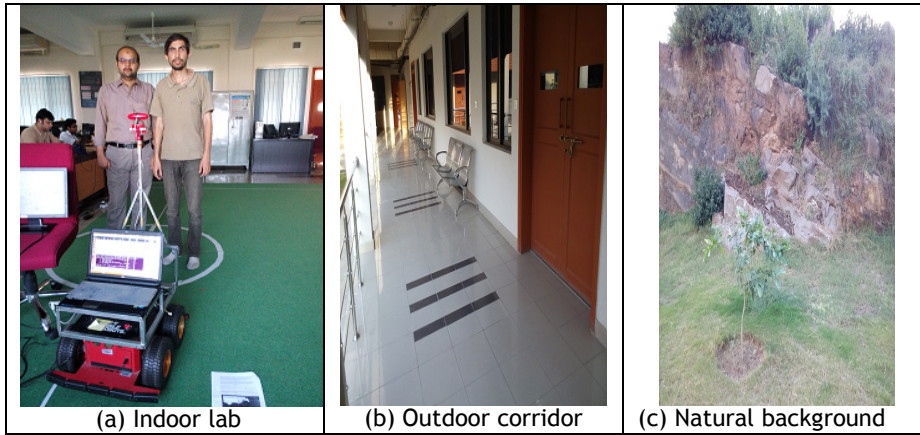


Figure 11: Benchmarks for testing

5.2 Results of target person tracking and stereo correspondence

Figure 12 presents the tracking results of the camshaft based tracking for all three benchmarks. Sequential images of various screenshots are presented. The camshaft can be prone to false tracking in conditions when the background of the environment does not differ much from the target because of the lighting. Furthermore, the tracking results of the meanshift method are not very good in images with really good lighting (i.e., $I = 0.7$).

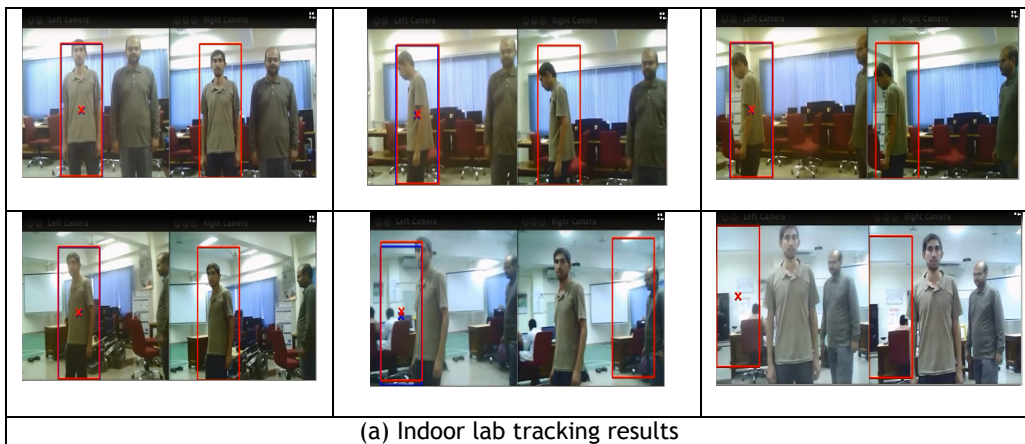




Figure 12: Testing results for camshift based tracking. (a), (b), (c): Sequential frames of all three benchmarks

Figure 13 presents the tracking results of the LK optical flow based tracking for all three benchmarks. Sequential images of various screenshots are presented. As mentioned earlier, twelve points are selected that are inside the contour of the target person. The optical flow of these points is used to track the person iteratively in the left camera image. It is observed that tracking efficiency decreases with a lot of movement in the environment near the target person ($D = 0.75$). It also undergoes a loss of tracking points when the target has moved sufficiently far from its initial position.

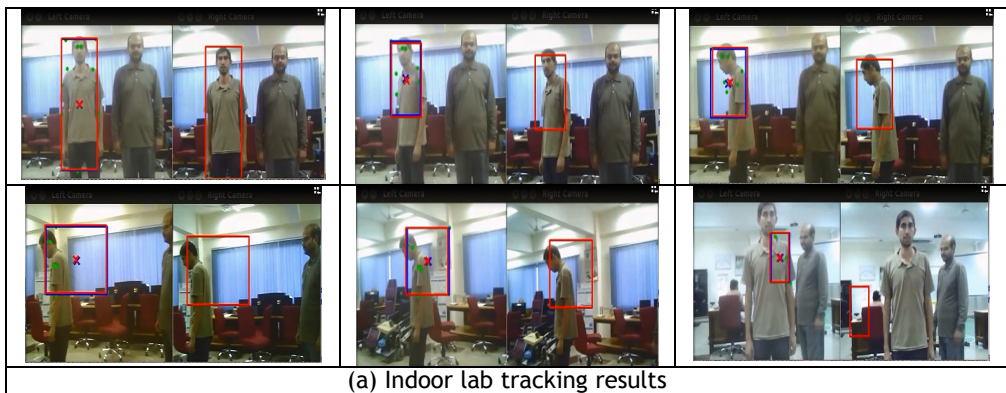




Figure 13: Testing results for LK optical flow-based tracking. (a), (b), (c): Sequential frames of all three benchmarks

Figure 14 presents the tracking results of the particle filter-based tracking for all three benchmarks. Sequential images of various screenshots are presented. The particle filter provided the best results of all three, as it is a non-parametric method. Only weights are assigned to each sample, based on the correlation of the RGB histogram. The rest of the samples are drawn based on random number generation. However, due to the non-parametric approach, it is sample-dependent.

6 COMPARISON OF TRACKING TECHNIQUES

On the basis of the results presented in the previous section, comparison of these techniques is based on various parameters. These parameters are tracking methodology, effect of illumination, effect of background colors, effect of movement in the background, and the processing time of each method. A detailed comparison based on these parameters is described below.

6.1 Methodology

The meanshift method uses the target's color histogram to generate the probability distribution. LK optical flow uses the pyramid-based flow of points computed with reference to the previous frame; whereas the particle filter is a sampling-based probabilistic approach that selects the highest weighted sample. Samples are generated randomly, and weights are assigned on the basis of correlation.

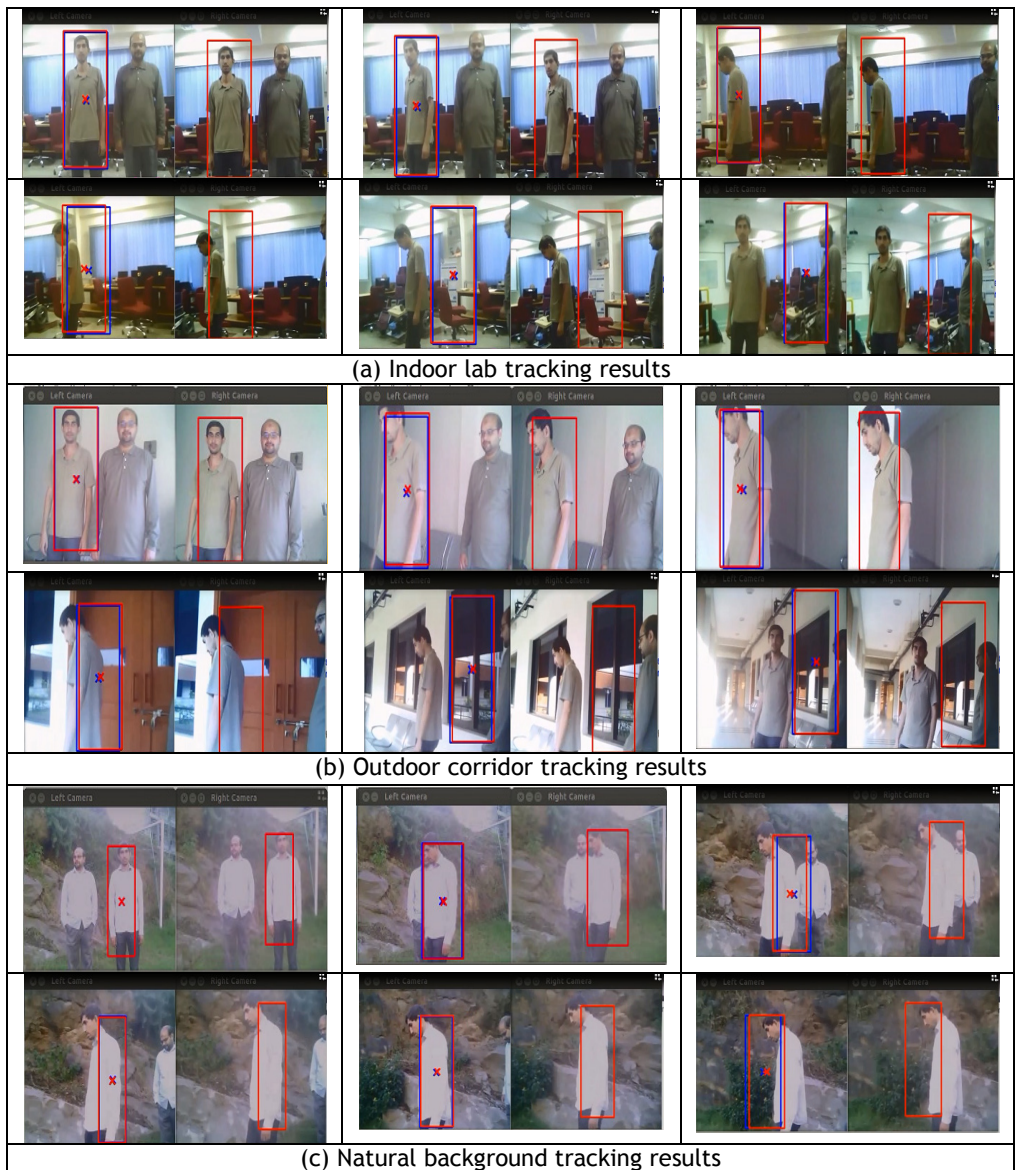


Figure 14: Testing results for particle filter-based tracking. (a), (b), (c): Sequential frames of all three benchmarks

6.2 Illumination

Illumination is measured based on the percentage of near-to-white pixels in the grayscale histogram of the image. The graph of Figure 15 presents the grayscale histogram of two benchmarks, and calculates the percentage of near-to-white pixels.

As the results of Figure 12 show, the meanshift method results in false tracking in all benchmarks having illumination of 55 per cent and 60 per cent respectively. On the other hand, the LK optical flow and particle filter show no influence of illumination in both benchmarks (Figures 13-14).

Table 1: Comparison of tracking techniques

Parameters	Meanshift	LK optical flow	Particle filter
Tracking methodology	Mean of target's color histogram	Flow of points in the contour of target	Probability-based weighted samples
Illumination in background (60% illumination, Figure 18)	False tracking under high illumination (Figure 12)	No background illumination influence (Figure 13)	No background illumination influence (Figures 14)
Effect of background colors (65% red, 70% green, 70% blue) (Figure 19)	False tracking when background colors are similar to the target (Figure 12)	No influence from background colors similar to the target (Figure 13)	No influence from background colors similar to the target (Figures 14)
Effect of movement in background	No effect of movement in background (Figure 12)	False tracking under movement in background (Figure 13)	No effect of movement in background (Figures 14)
Processing time	Average processing time per cycle is 0.02891 sec	Average processing time per cycle is 0.02712 sec	Average processing time per cycle is 0.02694 sec

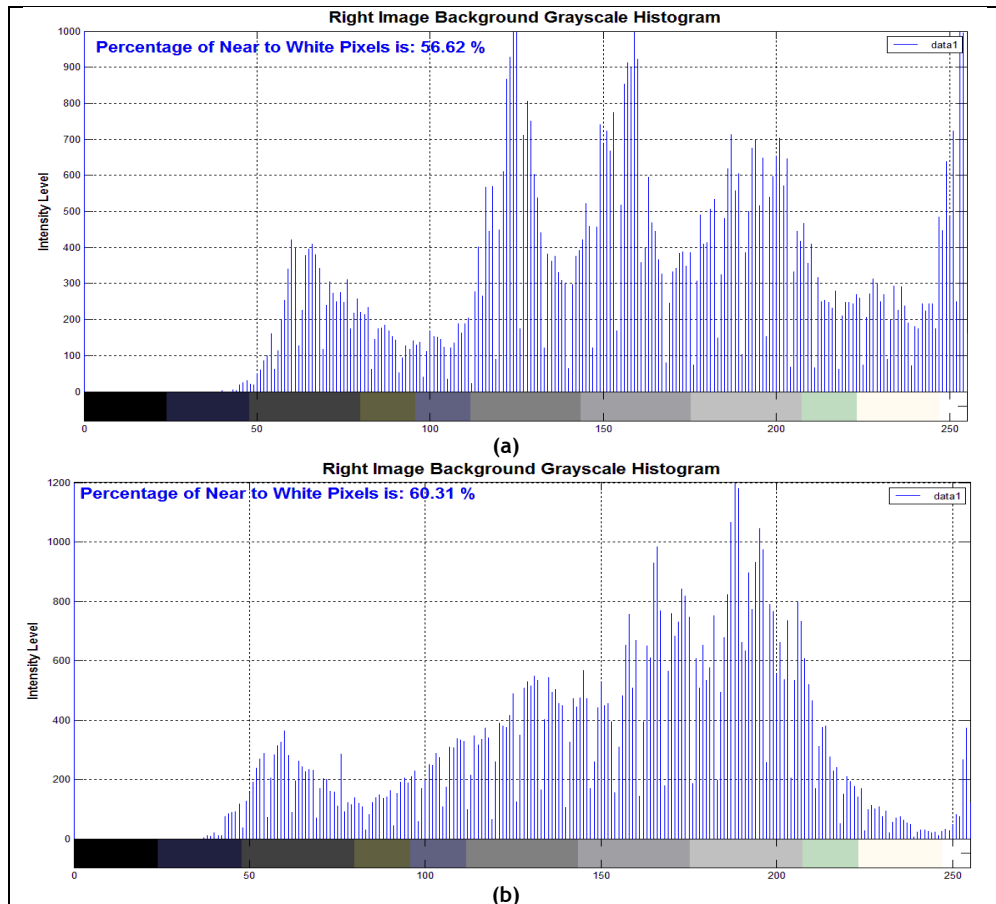


Figure 15: Grayscale histograms for near-to-white pixel percentage calculation. (a) Benchmark No 1 (b) Benchmark No 2

6.3 Background colors

Background colors are measured based on the percentage of near-to-red, -green and -blue pixels in the three-channel RGB histogram of image. The graph of Figure 16 presents the three-channel RGB histogram of two benchmarks, and calculates the percentages of pixels of RGB colors in the background.

As the results of Figure 12 show, the meanshift method results in false tracking in both benchmarks having background colors of 66 per cent red, 70 per cent green, 69 per cent blue and 58 per cent red, 51 per cent green, 54 per cent blue respectively. On the other hand, the LK optical flow and particle filter show no influence of similar background colors in both benchmarks (Figures 13-14).

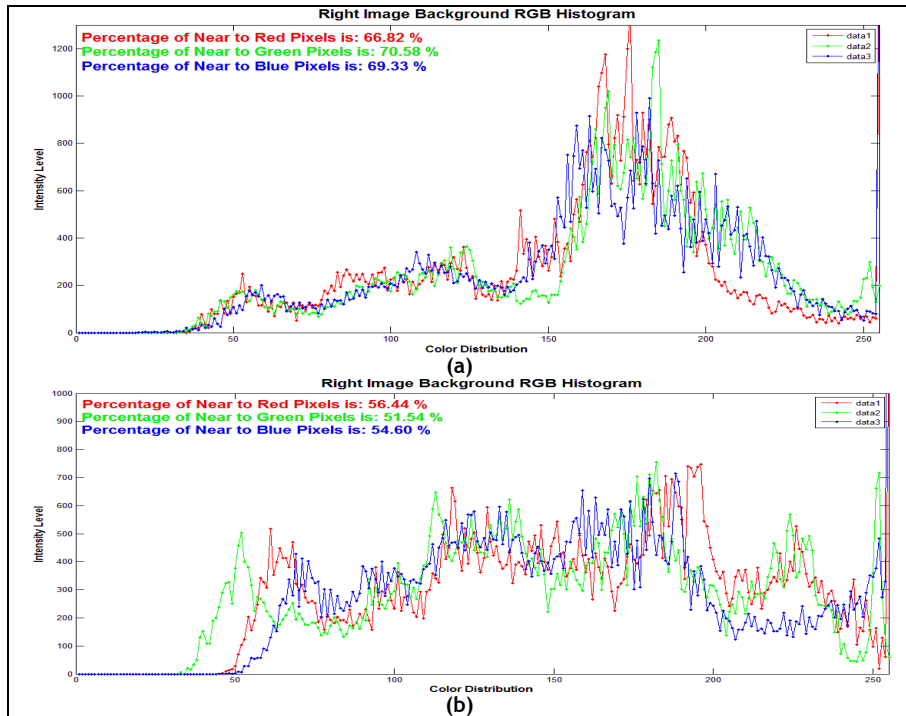


Figure 16: Three-channel RGB histograms for near-to-red, -green & -blue pixel percentage calculation. (a) Benchmark No 1 (b) Benchmark No 2

6.4 Movement in background

Meanshift shows no effect of movement in the background. In Figure 12, another person is moving in the background of the target person. Whereas the LK optical flow does false tracking when there is movement in the background (Figure 13), a particle filter like meanshift shows no influence of background movement (Figure 14).

6.5 Processing time

All computation has been done on a Core i3 laptop with a 2.4 GHz processor and 4GB of Ram. The average processing time per cycle for meanshift based tracking is 0.02649 sec, whereas the average processing time per cycle for the LK optical flow-based tracking is 0.02712 sec. The average processing time per cycle for the particle filter-based tracking is 0.02891 sec. All the processing has been done on a single unit laptop, while the position data is sent to the robot's computer to actuate the robot through a server client configuration. The processing time for all computation is a great deal faster than in other studies because the system is only running the tracking algorithm at any one time, rather than getting human detection results in each iteration.

Based on the above comparison, it can be said that all these tracking techniques have their own merits and demerits. When there is no problem of illumination, and background color is significantly different from the target, the meanshift based method is a good choice. However, if there is no movement in the background or if the environment is not dynamic, the LK optical flow will be the best one to use. On the other hand, the particle filter is not significantly influenced by any of these issues as it is a non-parametric uniform random sampling method; so it could be the most suitable for optimal results. The whole comparison has been summed up in Table 1.

Graphs for the target person distance and the angle from the robot with respect to time are shown in Figure 17. The graphs show that the robot maintains a distance of 1,500 mm and an angle of $\pm 10^\circ$ from the target.

7 CONCLUSION AND FUTURE WORK

In this paper we have presented an original approach to people detection and tracking in both indoor and outdoor environments. The contribution of this approach lies in its inherent ability to handle more than one person in the environment, and to distinguish the target person from other people nearby. Moreover, this technique is independent of the lighting conditions of the environment. The comparison of the three tracking techniques has also been presented.

In our future work, we shall focus on developing a motion planning technique that will be able to navigate in an environment with dynamic obstacles. One of the practical applications of this method might be the navigation of luggage cart carriers in malls, airports, or railway stations, etc. We can also use this technique in hospitals as an application for a service robot.

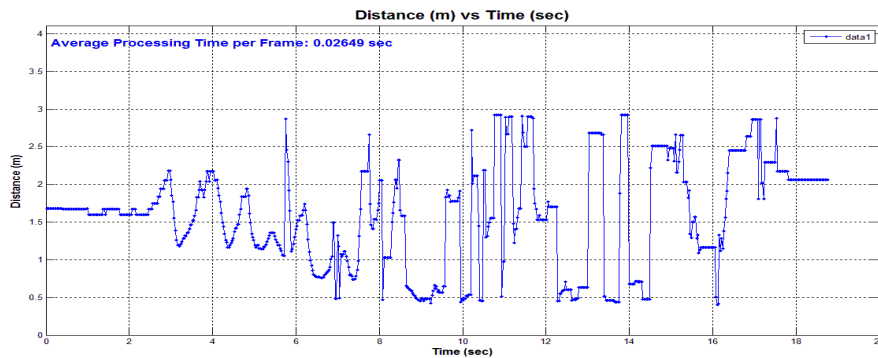


Figure 17a: Target position w.r.t. robot. - Distance vs time

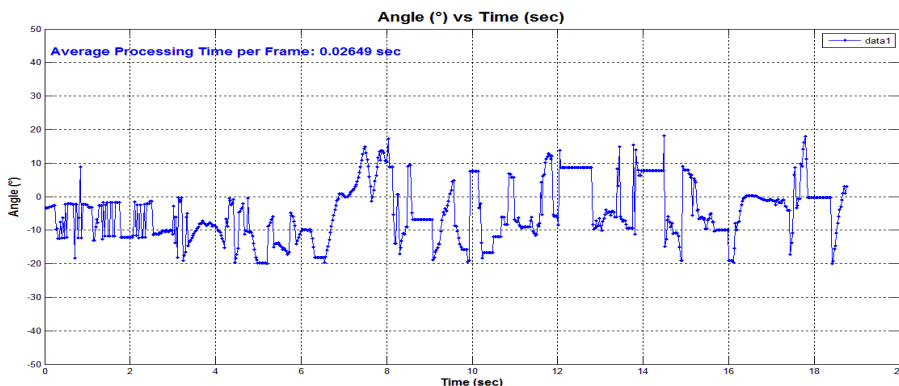


Figure 17b: Target position w.r.t. robot. - Angle vs time

REFERENCES

- [1] Jia, S., Zhao, L., & Li, X. 2012. Robustness improvement of human detecting and tracking for mobile robot. In *Mechatronics and Automation (ICMA), 2012 International Conference* .pp. 1904-1909. IEEE.
- [2] Kristou, M., Ohya, A. & Yuta, S. 2011. Target person identification and following based on omnidirectional camera and LRF data fusion, *20th IEEE International Symposium on Robot and Human Interactive Communication*. pp.419-424.
- [3] Bellotto, N. & Hu, H. 2006. Vision and laser data fusion for tracking people with a mobile robot, *Proceedings of the 2006 IEEE International Conference on Robotics and Biomimetics*. pp.7-12.
- [4] Patsadu, O., Nukoolkit, C. & Watanapa, B. 2012. Human gesture recognition using kinect camera, *International Joint Conference on Computer Science and Software Engineering*. pp. 28-32.
- [5] Petrović, E., Leu, A., Ristić-Durrant, D. & Nikolić, V. 2013. Stereo vision-based human tracking for robotic follower *International Journal of Advanced Robotic Systems*. 10:230 DOI: 10.5772/56124
- [6] Ali, B., Iqbal, K., Muhammad, N. & Ayaz, Y. 2013. Human detection and following by a mobile robot using 3d features, *Proceedings of IEEE International Conference on Mechatronics and Automation (ICMA)*, pp. 1714-1719.
- [7] Ali, B., Qureshi, A., Iqbal, K., Ayaz, Y., Gilani, S., Jamil, M., Muhammad, N., Ahmed, F., Muhammad, M., Kim, W-Y. & Ra, M. 2013 Human tracking by a mobile robot using 3D features, *Proceedings of the IEEE International Conference on Robotics and Biomimetics (ROBIO)*. pp. 2464-2469.
- [8] Viola, P. & Jones, M. 2001 Rapid object detection using a boosted cascade of simple features, In *Computer Vision and Pattern Recognition, 2001. CVPR 2001. Proceedings of the 2001 IEEE Computer Society Conference on (Vol. 1, pp. 1-511)*. IEEE.
- [9] Comaniciu, D. & Meer, P. Meanshift Analysis and Applications. *The Proceedings of the Seventh IEEE International Conference on Computer Vision* Vol. 2, pp. 1197-1203. IEEE.
- [10] Dementhon, D. Spatio-temporal segmentation of video by hierarchical meanshift analysis, Technical Report: LAMP-TR-090/CAR-TR-978/CS-TR-4388/UMIACS-TR-2002-68, University of Maryland, College Park, 2002.
- [11] Bouguet, J. Y. (2001). Pyramidal implementation of the affine lucas kanade feature tracker description of the algorithm, *Intel Corporation*, 5, 1-10.
- [12] Gustafsson, F. 2010. Particle filter theory and practice with positioning applications, *IEEE Aerospace and Electronic Systems Magazine*, pp. 53-82.
- [13] Li, P., Zhang, T. & Pece, A.E.C. 2003. Visual contour tracking based on particle filters, *Image and Vision Computing*, 21(1), pp. 111-123.
- [14] Castrillón, M., Déniz, O., Hernández, D., & Lorenzo, J. (2011). A comparison of face and facial feature detectors based on the Viola-Jones general object detection framework, *Machine Vision and Applications*, 22(3), 481-494.
- [15] Hartley, R. I., & Sturm, P. (1997). Triangulation, *Computer Vision and Image Understanding*, 68(2), 146-157.
- [16] Whitbrook, A *Programming mobile robots with Aria and Player*, Springer London Dordrecht Heidelberg, New York.
- [17] Bradski, G. & Kaehler, A. 2008. *Learning Open CV: Computer vision with the OpenCV library*, 1st edition. O'Reilly Media, Inc.
- [18] Nock, Richard & Frank Nielsen. A Real generalization of discrete AdaBoost, *Artificial Intelligence* 171, no. 1 (2007): 25-41.
- [19] Salhi, A. & Jammoussi, A. 2012 Object tracking system using camshift, meanshift and Kalman filter, *World Academy of Science, Engineering and Technology*, 64. pp. 674-679.