

Research Article

Dynamic Track Management in MHT for Pedestrian Tracking Using Laser Range Finder

**Abdul Hadi Abd Rahman,¹ Hairi Zamzuri,¹ Saiful Amri Mazlan,¹
Mohd Azizi Abdul Rahman,¹ Yoshio Yamamoto,² and Saiful Bahri Samsuri¹**

¹Vehicle System Engineering Research Lab, Universiti Teknologi Malaysia, Jalan Semarak, 54100 Kuala Lumpur, Malaysia

²Department of Precision Engineering, Tokai University, Hiratsuka 259-1292, Japan

Correspondence should be addressed to Hairi Zamzuri; hairi@ic.utm.my

Received 28 August 2014; Accepted 5 November 2014

Academic Editor: Yudong Zhang

Copyright © 2015 Abdul Hadi Abd Rahman et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Real time pedestrian tracking could be one of the important features for autonomous navigation. Laser Range Finder (LRF) produces accurate pedestrian data but a problem occurs when a pedestrian is represented by multiple clusters which affect the overall tracking process. Multiple Hypothesis Tracking (MHT) is a proven method to solve tracking problem but suffers a large computational cost. In this paper, a multilevel clustering of LRF data is proposed to improve the accuracy of a tracking system by adding another clustering level after the feature extraction process. A Dynamic Track Management (DTM) is introduced in MHT with multiple motion models to perform a track creation, association, and deletion. The experimental results from real time implementation prove that the proposed multiclustering is capable of producing a better performance with less computational complexity for a track management process. The proposed Dynamic Track Management is able to solve the tracking problem with lower computation time when dealing with occlusion, crossed track, and track deletion.

1. Introduction

Detection and tracking moving object system (DATMO) is one of the most important and popular research areas in autonomous navigation. Previously, DATMO is widely used for system monitoring using camera either in indoor or outdoor usage under both static and dynamic environments but it is unable to provide accurate measurements for long distance objects. A reliable DATMO system contributes to significant improvement for other scenarios such as obstacle avoidance, path planning, and collision avoidance. The presence of Laser Range Finder (LRF) which is capable of providing accurate range information, wide coverage area, and a low time interval permits implementations in real time system. Pedestrian tracking in urban area is one of the useful implementations of DATMO for autonomous navigation. When dealing with crowded environment or urban area, a DATMO system tends to face difficulties to deal with noisy data provided by LRF. Data for different pedestrians

sometime interlace and disrupt a reliable feature extraction to represent the pedestrians. The identification of each pedestrian needs to be fused with other information such as geometrical shapes and movement histories. In order to realize DATMO in real time implementation, a suitable optimization is needed especially in computational time and detection rate.

LRF placement on vehicle or robot is very crucial to detect either waist or leg portion of pedestrians. Both methods have their own benefits and drawbacks. In leg detection situation, a walking model is used to associate two legs data in order to represent the same pedestrian. Meanwhile the waist part data may contain data of pedestrian's hands which may sometime cause occlusion for full waist data. Experimental works done by Zhao et al. [1] proved that waist data contained better representation for pedestrians. The shape of a target pedestrian that changed for one iteration to another due to changes in orientation and translation is better detected by using waist data compared to leg. The size of associated target pedestrian helps to filter less possible clusters [2].

Most researchers implemented Iterative Closest Point (ICP) for data clustering. New cluster is created when two adjacent beams are separated by a prefix threshold distance. There are two different methods to measure the threshold either using a prefix threshold value or a threshold function. Arras et al. [3] used ICP with prefix threshold for segmentation and adaption of AdaBoost for training a strong classifier to associate features of consecutive laser scans which correspond to pedestrians legs. It worked in a cluttered environment but was applied only from a static position. Some of the researchers used ICP with adaptive threshold value as in [4, 5] followed by Expectation-Maximization (EM) algorithm to separate or merge clusters. The experimental results showed that the true detection rates for all the experiments are very low for a single layer LRF. Klasing et al. [6] and Wenzl et al. [7] used fluctuations Δr to represent the circumcircle transformation effected by upper and lower limb movements. The position of an object is estimated by calculating the distance L between the LRF sensor and the object surface with the measurement error ΔL . The estimated distance between a LRF sensor and the object centroid is $Lr = L + r$, where r denotes the estimated mean radius.

There are numbers of approaches which have been conducted for feature extraction. Zhang et al. [8] proposed a mutation operator (MBPSO) based wrappers to reduce false positive error. It can solve other classification problems such as web-page, text, and image classification. Wrappers can perform better than filters with regard to classification performance indexes. False positive error can be reduced without compromising the sensitivity and accuracy values. Zhang et al. [9] has conducted stationary wavelet transform (SWT) to extract features in order to encountered problem occurring when using discrete wavelet transform (DWT) based classification. By applying DWT, one subject was recognized as two different subjects when the centres of the images are located at slightly different positions. SWT based features also allowed to be implemented in the MR image classification, denoising, compression, and fusion. The finding showed that SWT is superior to classical discrete wavelet transform associated with translation invariant property.

Multiple Hypothesis Tracking (MHT) algorithm preserves multiple hypotheses associating previous observations with targets in multiscan association. A new set of hypotheses is formed based on previous hypothesis after receiving new measurements at each iteration. The algorithm calculates for the highest posterior hypothesis to be chosen as the solution. MHT is categorized as a deferred logic method where a track creation or deletion of existing track is only done when the measurements are completely gathered. Hypotheses-based MHT suffers from the computational complexity to maintain and expand the previous hypotheses. To overcome this problem, several heuristic methods are applied but they sacrifice the Maximum A Posteriori (MAP) estimation property. Ding and Chen [10] successfully verified the MHT technique via simulation for moving object tracking to be applied in unmanned vehicle operation. Coraluppi [11] provided simulation results demonstrating the advantages of multistage MHT processing compared to single-stage MHT, track-while-fuse processing. The simulation results

proved that multistage MHT offers a powerful and flexible paradigm to circumvent limitations in conventional MHT processing. Vasquez and Williams [12] proposed MHT with Integral Square Error Reduction (MISER) to represent the dependency which occurs between targets effected by the joint observation process and applied Integral Square Error (ISE) mixture reduction algorithm to control the hypothesis growth. Mohammed et al. [13] developed Fast-IMM (Fast Interacting Multiple Model) for tracking a single maneuvering target which decreases the computational burden but keeps a high accuracy. In the study by Blackman [14], it is proven that track-oriented MHT can easily maintain several hundred of tracks and expand into new hypotheses for difficult scenarios. Vu et al. [15] implemented an adaptive Interacting Multiple Models filter which consists of 16 motion models coupled with a Multiple Hypothesis Tracker to solve moving objects tracking.

In this paper, a multiclustering method is introduced to solve spurious data obtained from Laser Range Finder in an outdoor environment. For tracking part, a new Dynamic Track Management for MHT (DTM-MHT) is proposed to produce lower computational time for real time implementation in urban area. It is evaluated to deal with track creation and crossed track, as well as track deletion in real time implementation.

2. Multiclustering

Multiclustering is significantly important to reduce a total size of observation at each iteration before data tracking process is done. An efficient multiclustering provides more reliable observation data which will reduce complexity in track management process. Firstly, the laser data at each iteration are clustered using ICP with adaptive threshold. Then, filtering is accomplished by removing clusters with total point numbers less than 5. This simple filtering is important to reduce the computation (DTM) time by eliminating spurious data from measurement noise. Then the measurement of the width for each cluster is done for baseline before feature extraction. The width of the cluster is important to reduce the computation time for unwanted features other than pedestrian which has the width between 40 cm and 70 cm. The next step is to determine the geometrical feature of the detected object as mentioned in [16]. Feature extraction using an ellipse fit and a circle fit was used for all clusters with circle or arc features. The feature extraction results are then transformed into global Cartesian coordinates based on the vehicle position.

The multiclustering process is illustrated as in Figure 1. The Laser Range Finder (LRF) provides the polar coordinate (r, θ) for the body and arm parts of a pedestrian. L represents the threshold distance between body and arm. The rotation and translation of the pedestrians are then calculated according to the feedback from motion model (MM) selection in data association process. MM is attributed to the laser data at $t - 1$. With the selected motion model and the tracking result from the previous iteration, the region area for each pedestrian is computed and multiclustering of all

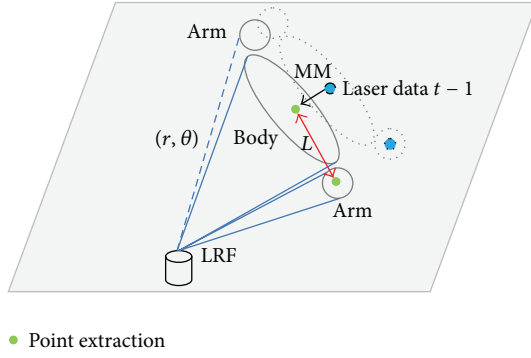


FIGURE 1: Multiclustering technique.

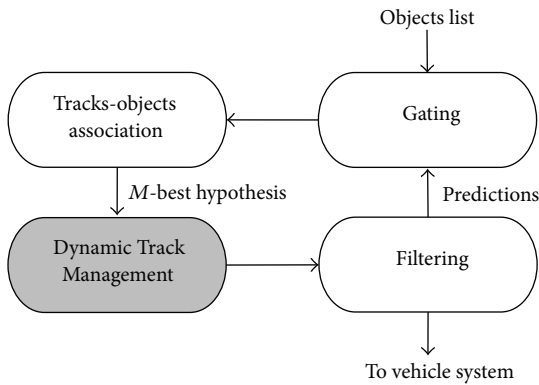


FIGURE 2: MHT process.

the neighbouring clusters is accomplished using the following equation:

$$\text{Nbr}(C_i^t) = \{C_j^{t-1} \mid \text{dist}(C_i^t, C_j^{t-1})\} < \text{Thresh}_{\text{nbr}}, \quad (1)$$

where C_i^t denotes the i th cluster at time t and $\text{dist}(\cdot)$ denotes the function for computing the distance between two clusters. Each pair of clusters C_j^{t-1}, C_j^{t-2} in $\text{Nbr}(C_i^t)$ is tested whether a merge case is generated according to the size of the overlapping area for a pedestrian.

3. Multiple Hypothesis Tracking

The flow for MHT is illustrated in Figure 2. The MHT forms different hypotheses for every possibility which includes detection of new tracks, false alarms, or existing tracks. First, the object list is calculated to determine candidates for association in the gating procedure. The results of the gating process will undergo a data association process to find the best possible track-object association. Next, the tracks are managed to deal with track creation, track deletion, and false measurement. Furthermore, the movement for each object for the next iteration is predicted in the filtering process. These processes are repeated for every iteration. The details of each step of MHT with proposed Dynamic Track Management are described in this section.

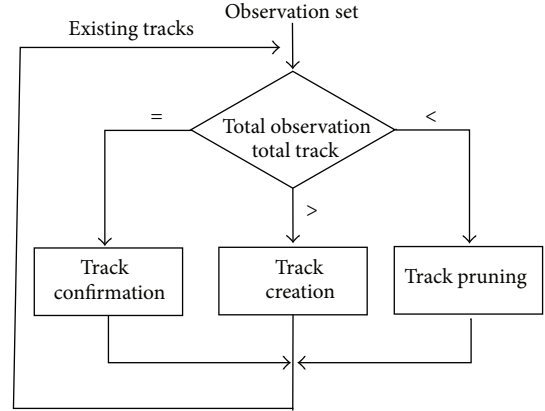


FIGURE 3: Dynamic Track Management.

3.1. Gating. The object list obtained from the feature extraction of raw laser data is used as an input for gating procedure. Gating is necessary for early detection of false measurements and new objects. Global Nearest Neighbour (GNN) algorithm is used to calculate the distance between the tracks and observations, which produces candidates for association. The result of the gating procedure is a set of compatible observation-track pairs of current hypothesis which is processed in data association step.

3.2. Data Association. Reids algorithm [17] is a hypothesis-based MHT implementation which keeps the past different hypotheses in the memory between consecutive time steps. When a new measurement falls within the gate region, it will form a list of possible assignments between measurements and exiting tracks which expanded the existing hypothesis. Each hypothesis contains a set of potential track-object assignments, which cause an inhibitory time consuming process in determining all the possible assignment combinations. If there is no measurement which is compatible with one of existing tracks, then a new track or a false alarm should be formed. The implementation of MHT considered in this paper is similar to running GNN-based multitarget trackers in parallel. Then N -scan pruning and M -best hypotheses techniques are used to reduce the number of hypotheses. M -best hypothesis is able to optimally determine the M -best assignments in polynomial time. It reduces the dimension and thus precludes solving duplicate assignment problems.

3.3. Dynamic Track Management. At this stage, Dynamic Track Management is proposed to manage track creation, confirmation, or deletion as shown in Figure 3. The implementation is based on track-oriented approach. New hypotheses are formed using the updated tracks and the new observation data on each iteration.

The process starts with an early detection to determine which type of management needs to be executed. It is done by subtracting the number of observation set from the number of existing tracks available. The previous probability for

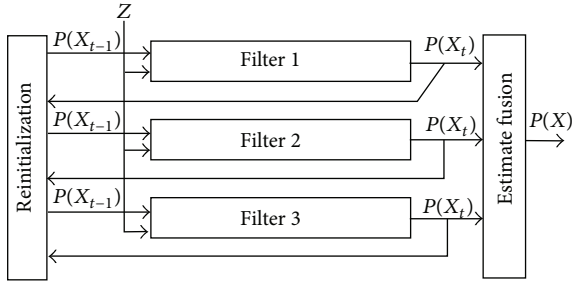


FIGURE 4: Basic IMM steps.

each track is considered in this step. The creation of new tracks only takes place when a new track creation hypothesis appears in the M -best hypotheses. The new tracks are confirmed after detected objects appear along the same track over several consecutive iterations to prevent track creation for spurious measurements. Evaluation of track scores using Sequential Probability Ratio Test (SPRT) is used to perform the track confirmation test. The selection of confirmation and deletion thresholds is associated with the tracking requirements through the parameters α for false track confirmation and β for true track deletion probability. A new hypothesis is reformed by using the new observations based on the tracks which survived after track pruning process. The survive tracks after pruning process are predicted using the new observations obtained and reformed into new hypotheses. Some tracks are deleted based on low probability or N -scan pruning. The track scores containing all the relevant statistical data for track pruning are maintained for the next track management process.

Nondetection hypothesis appears when occlusion of measurement occurs, that is, when an object is hidden by another object or disappears from perception sensor. When this happens, the track measurement for unassociated objects is updated according to the previous associated objects via prediction using Interacting Multiple Models (IMM) in a later step. The track will be deleted and would be unavailable to be occupied for any upcoming observations when a track is not updated for a certain iteration limit. Furthermore, the continued growth of the tracks is controlled by performing another pruning. Typically the growth is controlled by N -scan pruning technique by keeping only the N previous scans in the trees. The numbers of tracks trees were delimited to N which simply the execution of MHT algorithm.

3.4. Filtering. The basic Interacting Multiple Models (IMM) steps are illustrated in Figure 4. It consists of multiple filters which are fused to estimate the next pedestrian's state. The pedestrian's movements were emulated by a set of possible directions using IMM models. The parameters for each motion model were predefined as it represents the corresponding filter. With an assumption of having the same velocities in eight possible directions, motion of an object will fall in one of the 8 motion models as shown in Figure 5.

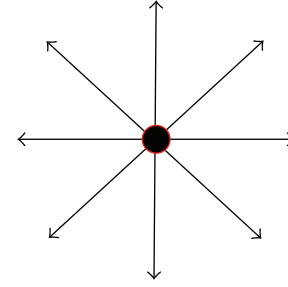


FIGURE 5: Motion models for IMM.

Kalman filters are integrated in each motion model for prediction. The Kalman filter equations are given as follows:

$$\begin{aligned} x_k &= F_k x_{k-1} + G_k u_{k-1} + w_k, \\ Z_k &= H_k x_k + v_k, \end{aligned} \quad (2)$$

where matrices F_k , G_k , and H_k are time dependent and the noises w_k and v_k have covariance matrices Q_k and R_k , respectively. If the prior distribution $p(x_o)$ is also a Gaussian $N(x_o; \hat{x}_o, P_o)$, the distributions resulting from the prediction and update steps are also Gaussian. Therefore, the beliefs of the Kalman filter are completely specified by the first two moments of the distribution. Thus, the distribution $p(x_k | Z_{1:k-1}, u_{0:k-1}) = N(x_k; \hat{x}_{k|k-1}, P_{k|k-1})$ where the mean $x_{k|k-1}$ and covariance matrix $P_{k|k-1}$ are

$$\begin{aligned} \hat{x}_{k|k-1} &= F_k \hat{x}_{k-1} + G_k u_{k-1}, \\ P_{k|k-1} &= F_k P_{k-1} F_k^T + Q_k. \end{aligned} \quad (3)$$

In the same way, the mean \hat{x}_k and covariance matrix P_k resulting from the update step are

$$\begin{aligned} \hat{x}_k &= \hat{x}_{k|k-1} + W_k v_k, \\ P_k &= P_{k|k-1} - W_k S_k W_k^T, \end{aligned} \quad (4)$$

where the innovation v , the predicted measurement $\hat{Z}_{k|k-1}$, the innovation covariance S_k , and the filter gain W_k are

$$\begin{aligned} v &= (Z_k - \hat{Z}_{k|k-1}), \\ \hat{Z}_{k|k-1} &= H_k \hat{x}_{k|k-1}, \\ S_k &= H_k P_{k|k-1} H_k^T + R_k, \\ W_k &= P_{k|k-1} H_k^T S_k^{-1}. \end{aligned} \quad (5)$$

4. Experimental Setup

The developed DATMO algorithm with the DTM-MHT was evaluated on its capability to deal with various important situations in tracking part. Two most important aspects for evaluation were tracking computation time and position accuracy during tracking process. The capabilities of pro-

TABLE 1: Experiment parameters.

Parameter	Setting
Area	40 meters \times 20 meters
Scenario duration	770 iterations
Target number	1 to 6 objects
Sampling rate	0.2 s
Laser distance range	5 meters
Track creation	3 hits
Track kill	8 misses



FIGURE 6: Hokuyo LRF placed on iREV.

posed multiclustering were tested with real time implementation in an outdoor environment where a high level of noise was expected. It becomes more complex when LRF data are obtained from a moving vehicle. The proposed Dynamic Track Management was evaluated on the capability to handle missing observation cases due to occlusion, cross track, track creation, and track deletion. The tracking computation time was computed in every iteration for evaluation.

The details of the experimental setup are shown in Table 1. The experiment was conducted in area of 40 m \times 20 m. A custom-made buggy car called iREV (Instrumented Research Vehicle) developed in [18] was used to provide the trajectory data which contained the vehicle pose state (X , Y and the heading, θ). A Hokuyo Laser Range Finder (LRF) was placed in front of the vehicle at 1.2 m height from the ground to provide multiple pedestrian data on the waist part as shown in Figure 6. The data from LRF covers the area within 5 meters in 180° width. The resolution of the LRF was set to 0.333° to provide more data compared to 0.5° used in most previous works. This simply increases the accuracy in feature extraction process of multiple pedestrians.

Figure 7 shows three different scenarios for evaluations and analysis. Figure 7(a) involves 5 pedestrians appearing from different direction heading across the moving vehicle. There occurs an occlusion scenario where two pedestrians walk in parallel to each other causing blind spot from the laser perception. The second scenario is found in Figure 7(b) where 4 pedestrians are detected and moving closely to each other in the same direction labelled as “B.” The occlusion situations were labelled as label “A” and “C” while “D” represents one of the track deletion cases. Figure 7(c) was set up to deal with high measurement noise from laser data

labelled as “A” while “B” represents crossed track situation. In this experiment, each scenario is separated by a longer time interval. It is considered as a more realistic situation where pedestrians would not appear for a long time as the vehicle moves along its path.

The computational time was computed for evaluation of the overall tracking process which includes multiclustering process and DTM-MHT. The performance of tracking results for position of each pedestrian was then evaluated using Root Mean Squared Error (RMSE).

5. Results and Discussion

This section presents the results of the pedestrian tracking from a moving vehicle based on the experimental setup. As mentioned before, multiclustering is supposed to deal with spurious data from LRF sensor. The first stage clustering contained noise data and cluster candidates. Thus, choice of selecting threshold value is critical in selecting candidates for multiclustering process. The results from multiclustering are used for feature extraction process. The feature extraction results from multiclustering process were represented and labelled as PE in Figures 8(a), 8(b), and 8(c) using an ellipse fit technique. It is observed that some extraction points are fluctuated due to measurement noise. However, in unlike the tracking part, the occlusion problem could not be resolved in these feature extraction results.

The occlusion problems were solved using the motion model estimation. Use of motion models allows the prediction of next pedestrian movement. The selection of motion models are based on the highest probability obtained from previous hypotheses. Next, filtering with Kalman filter on the motion models information was done to complete the pedestrian tracking process. Figures 8(a), 8(b), and 8(c) presented the tracking path produced by the motion models. The motion model selection during *missed detection* situation produced estimation for certain period of time based on track kill threshold set in Table 1 before track deletion is done or maintained the track. Based on this motion model, filtering was done by using Kalman filter to predict the next movement of the entire pedestrians. The tracking results reflect successful implementation for DATMO algorithm in dealing with various scenarios as explained in the experimental setup.

In the first scenario, temporarily missed detection labelled as “A” and “B” in Figure 7(a) was solved as illustrated in Figure 8(a). The pedestrian that temporarily disappeared for a certain period of time and then reappeared was tracked via estimation using motion models and labelled as MM2 in Figure 8(a). The tracking results related to crossed track and occlusion situation are labelled as $Est4$ and $Est5$. All the tracks were successfully deleted after they disappeared for a certain period of time or static motion models were selected. The result for the second scenario is shown in Figure 8(b) where 4 pedestrians were tracked moving across the moving vehicle. Occlusion appeared due to pedestrians’ parallel movements. In this scenario, the proposed algorithm was evaluated to deal with close distance between tracked pedestrians. The tracking results for the third scenario are shown in Figure 8(c) to deal

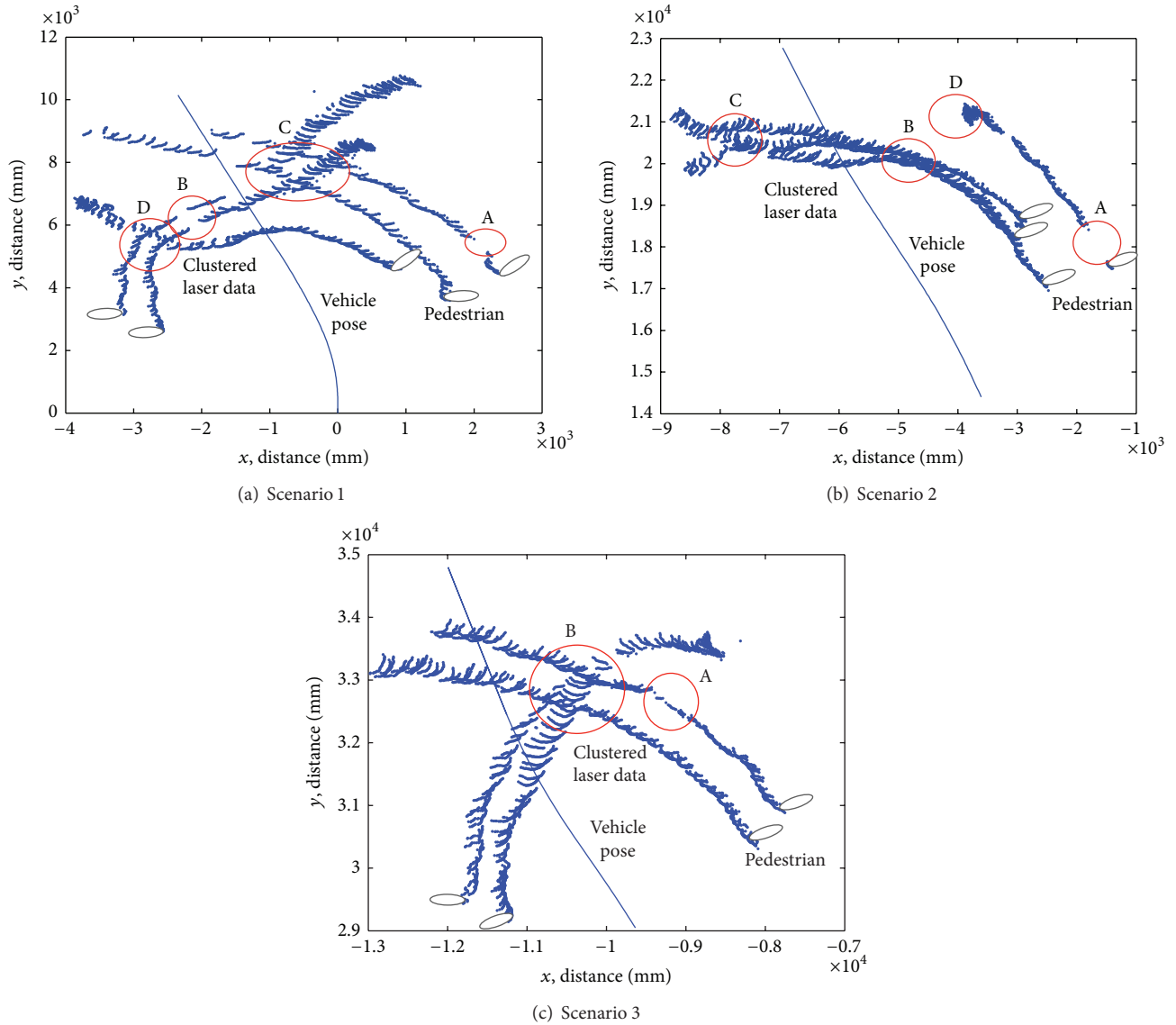


FIGURE 7: Data of multiple pedestrians in three different scenarios.

with data association during crossed track. It is observed that the tracking was having difficulties in solving tracking due to high measurement noise from laser data but still able to associate pedestrian to track correctly.

The overall performance of the tracking algorithm for the all scenarios is indicated by Root Mean Squared Error of positions for multiple pedestrians as tabulated in Table 2. The results reflected the tracking performance of the proposed algorithm. It is suggested that the algorithm was able to perform correct data association for all 5 pedestrians based in all given scenarios. RMSE values for position of pedestrians labelled 1, 3, and 5 were found lower compared to those for pedestrians 2 and 4. They were affected by consistent detections for all the pedestrians until they disappeared while the latter had missed the detections and thus affected the tracking results. The RMSE values for the positions of all pedestrians were less than 0.5 for all pedestrians in the second

TABLE 2: RMSE for position for each tracked pedestrian.

RMSE	Scenario 1		Scenario 2		Scenario 3	
	x (m)	y (m)	x (m)	y (m)	x (m)	y (m)
Pedestrian 1	0.131	0.281	0.106	0.186	0.125	0.447
Pedestrian 2	0.127	0.112	0.093	0.247	0.231	0.461
Pedestrian 3	0.138	0.224	0.086	0.237	0.231	0.49
Pedestrian 4	0.12	0.129	0.053	0.28	0.187	0.435
Pedestrian 5	0.142	0.134	—	—	—	—

and third scenarios, which reflected successful tracking for all pedestrians.

The computation time of the proposed DTM-MHT was computed to evaluate the complexity of the tracking process. Figure 9(a) shows the relation between track number and

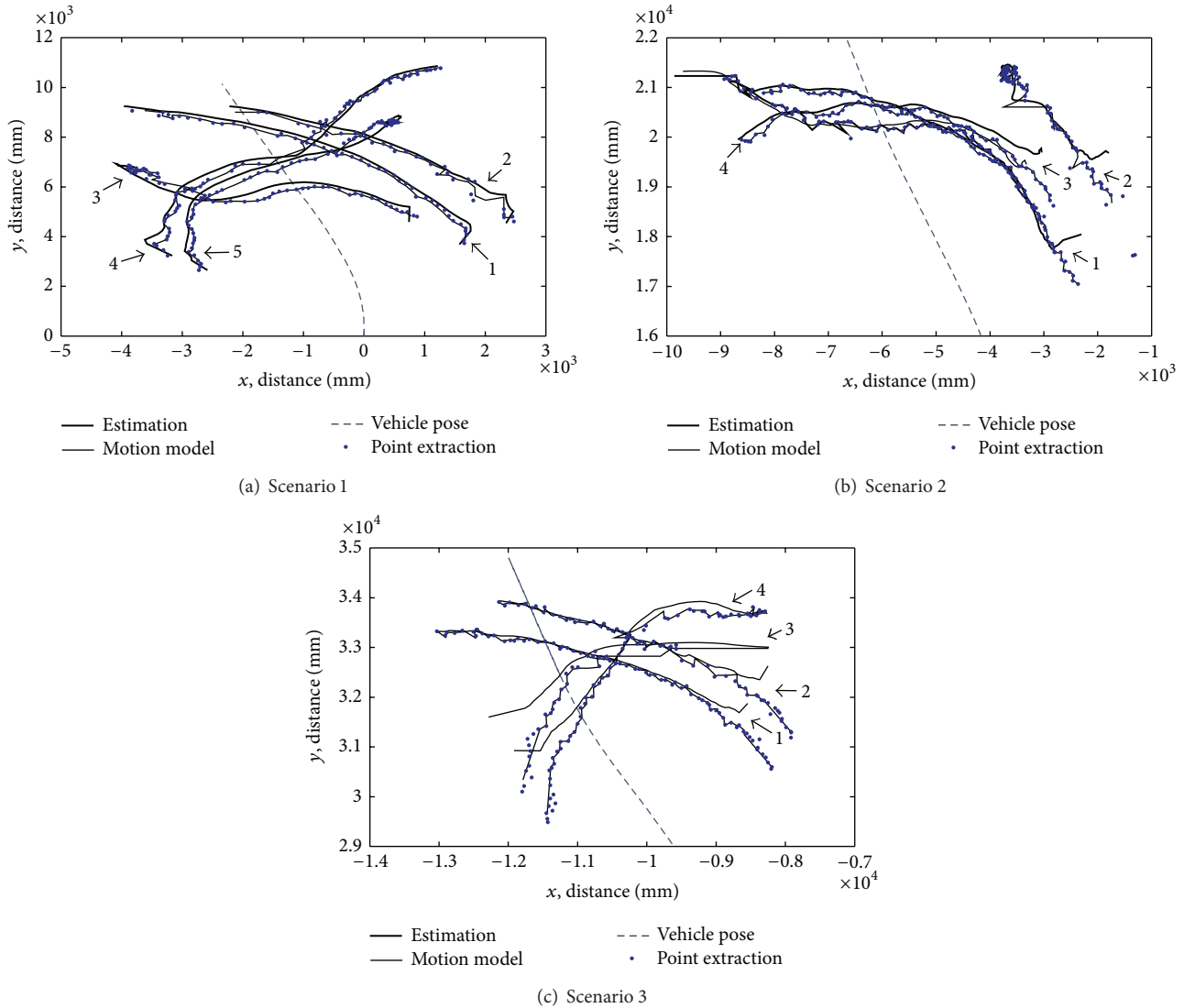


FIGURE 8: Tracking results for all tracks.

the computation time. The average computation time in idle cases where no tracking was performed is 9 ms. At 115th iterations, initialization process started when new tracks are added and started the tracking process. During this process, the computation time was at the peak level at 30 ms. The algorithm processed 5 pedestrians from 125th to 161th iterations and recorded computation time at 14.2 ms. The computational time is fluctuated due to missed detection which requires more computation time. From 162th to 181th iterations, the tracks reduce to four tracks which recorded less computational time at 13.1 ms. The average computation time reduced to 12.9 ms when tracking 2 pedestrians from 182th to 193th iterations. The fact that tracks reduced to one and disappeared after a few iterations means no available pedestrian to track. The computation time and track management second scenarios are illustrated in Figure 9(b). Track creation was performed at the 547th iteration when 2 pedestrians were detected and tracked. Furthermore tracks are added in

the 559th and 581th iterations. Missed detection happened between 584th and 591th iterations but no track deletion was performed as it still did not exceed the value for track deletion. Track deletions were accomplished at iterations 620, 629, and 643. In the 3rd scenario as shown in Figure 9(c), track creation started at the 678th iteration and the tracks were maintained until the 701th iteration before a track deletion was performed. All 4 tracks were deleted at 748th iteration.

From the results in all three scenarios, the average computation time for each number of pedestrians is recorded between 11.8 ms and 17.23 ms as in Table 3 to deal with up to 5 pedestrians. The average incremental time needed to process every new pedestrian is 1.3575 ms. It allows process for larger number of pedestrians for each scene. The track reinitialization adopted in the track management process between two consecutive scenes reduces the hypotheses complexity problem which could be enlarged as a response to the incremental number of pedestrians tracked.

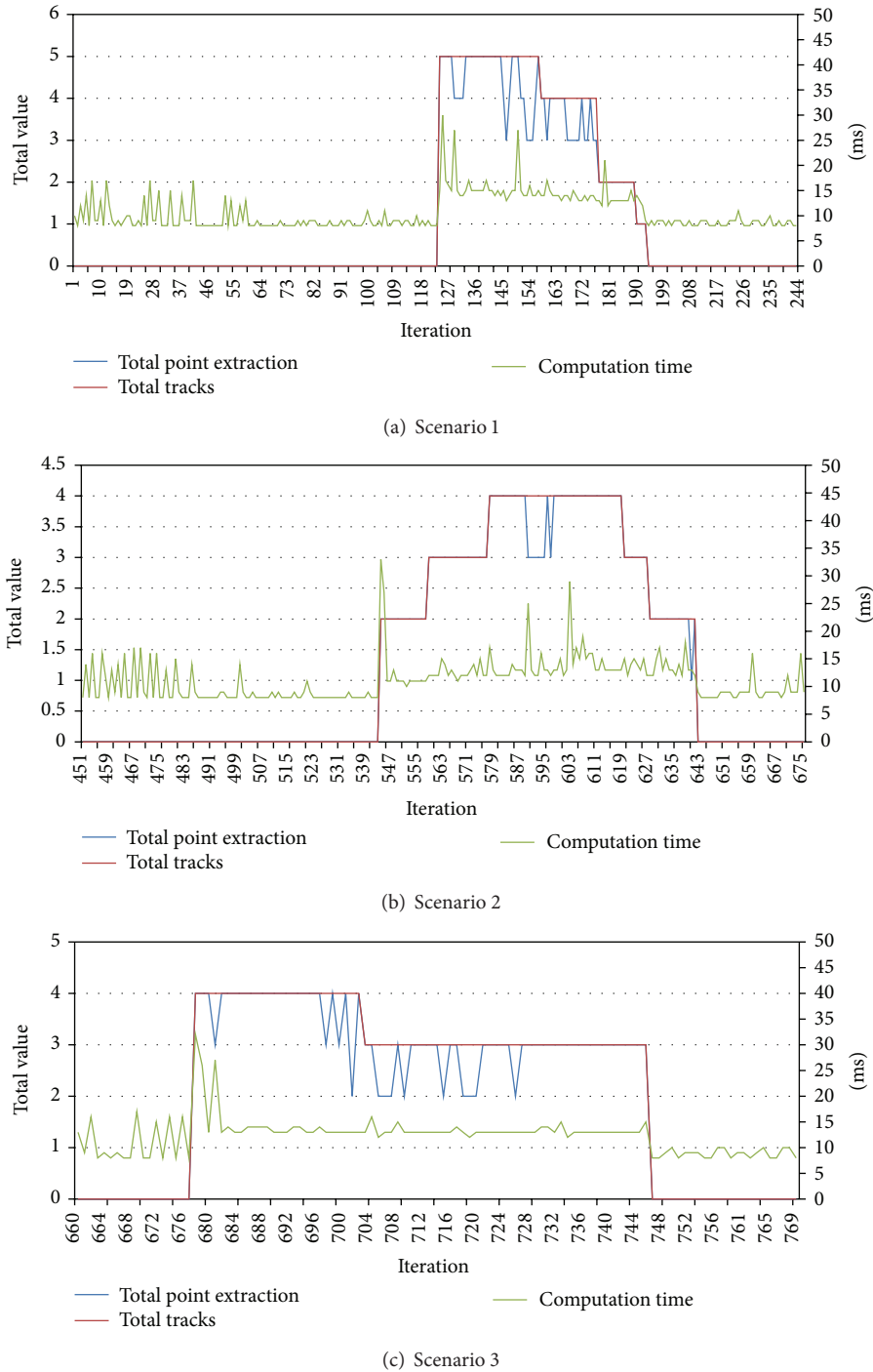


FIGURE 9: Computational time and track management.

6. Conclusion and Future Work

This paper mainly presents the experimental results of the developed DATMO algorithm with DTM-MHT and multiclustering for tracking multiple pedestrians in real time implementation. It was validated via real time experiment in terms of its computational time and accuracy. It was observed that low computational time is achieved in all

scenarios given and capable of associating all the detected object with their tracks and predicted their movements. The multiclustering method successfully filtered unwanted data and produced reliable observations for tracking purpose. Furthermore, this proposed algorithm will be applied in more complex scenarios involving more pedestrians. The detection accuracy could be improved using fusion of various sensors and configurations. The tracking technique could be

TABLE 3: Average computation time for each tracked pedestrian.

Total pedestrian	Average computation time
1	11.8 ms
2	12.63 ms
3	13.41 ms
4	15.34 ms
5	17.23 ms

expanded to other fields such as image processing and other intelligent surveillance applications.

Conflict of Interests

The authors declare that there is no conflict of interests regarding the publication of this paper.

Acknowledgments

The authors would like to thank the Malaysia-Japan International Institute of Technology (MJIIT) in Universiti Teknologi Malaysia and Ministry of Education Malaysia (MOE) under FRGS (vote: 4F370) and Research University Grant (vote: 00G64) for funding and supporting this research.

References

- [1] T. Zhao, R. Nevatia, and B. Wu, "Segmentation and tracking of multiple humans in crowded environments," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 30, no. 7, pp. 1198–1211, 2008.
- [2] F.-M. Chang and F.-L. Lian, "Polar grid based robust pedestrian tracking with indoor mobile robot using multiple hypothesis tracking algorithm," in *Proceedings of the 51st Annual Conference on of the Society of Instrument and Control Engineers of Japan (SICE '12)*, pp. 1558–1563, Akita, Japan, August 2012.
- [3] K. O. Arras, Ó. M. Mozos, and W. Burgard, "Using boosted features for the detection of people in 2D range data," in *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA '07)*, pp. 3402–3407, IEEE, April 2007.
- [4] A. Carballo, A. Ohya, and S. Yuta, "Reliable people detection using range and intensity data from multiple layers of laser range finders on a mobile robot," *International Journal of Social Robotics*, vol. 3, no. 2, pp. 167–186, 2011.
- [5] P. Kondaxakis, S. Kasderidis, and P. Trahanias, "A multi-target tracking technique for mobile robots using a laser range scanner," in *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS '08)*, pp. 3370–3377, September 2008.
- [6] K. Klasing, D. Wollherr, and M. Buss, "Realtime segmentation of range data using continuous nearest neighbors," in *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA '09)*, pp. 2431–2436, May 2009.
- [7] K. Wenzl, H. Ruser, and C. Kargel, "Performance evaluation of a decentralized multitarget-tracking algorithm using a LIDAR sensor network with stationary beams," *IEEE Transactions on Instrumentation and Measurement*, vol. 62, no. 5, pp. 1174–1182, 2013.
- [8] Y. Zhang, S. Wang, P. Phillips, and G. Ji, "Binary PSO with mutation operator for feature selection using decision tree applied to spam detection," *Knowledge-Based Systems*, vol. 64, pp. 22–31, 2014.
- [9] Y. Zhang, S. Wang, Y. Huo, L. Wu, and A. Liu, "Feature extraction of brain MRI by stationary wavelet transform and its applications," *Journal of Biological Systems*, vol. 18, no. 1, pp. 115–132, 2010.
- [10] R. Ding and W.-H. Chen, "Moving object tracking in support of unmanned vehicle operation," in *Proceedings of the 19th International Conference on Automation and Computing (ICAC '13)*, pp. 1–6, IEEE, London, UK, September 2013.
- [11] S. Coraluppi, "Multi-stage multiple-hypothesis tracking," *Journal of Advances in Information Fusion*, vol. 6, no. 1, pp. 57–68, 2011.
- [12] J. R. Vasquez and J. L. Williams, "Improved hypothesis selection for multiple hypothesis tracking," in *Signal and Data Processing of Small Targets*, vol. 5428 of *Proceedings of SPIE*, 2005.
- [13] D. Mohammed, K. Mokhtar, O. Abdelaziz, and M. Abdelkrim, "A new IMM algorithm using fixed coefficients filters (fast-IMM)," *AEU—International Journal of Electronics and Communications*, vol. 64, no. 12, pp. 1123–1127, 2010.
- [14] S. S. Blackman, "Multiple hypothesis tracking for multiple target tracking," *IEEE Aerospace and Electronic Systems Magazine*, vol. 19, no. 1, pp. 5–18, 2004.
- [15] T.-D. Vu, J. Burlet, and O. Aycard, "Grid-based localization and local mapping with moving object detection and tracking," *Information Fusion*, vol. 12, no. 1, pp. 58–69, 2011.
- [16] J. Xavier, M. Pacheco, D. Castro, A. Ruanot, and U. Nunes, "Fast line, arc/circle and leg detection from laser scan data in a player driver," in *Proceedings of the IEEE International Conference on Robotics and Automation*, pp. 3930–3935, IEEE, April 2005.
- [17] D. B. Reid, "An algorithm for tracking multiple targets," *IEEE Transactions on Automatic Control*, vol. 24, no. 6, pp. 843–854, 1979.
- [18] M. A. Zakaria, H. Zamzuri, R. Mamat, and S. A. Mazlan, "A path tracking algorithm using future prediction control with spike detection for an autonomous vehicle robot," *International Journal of Advanced Robotic Systems*, vol. 10, article 309, 9 pages, 2013.



Hindawi

Submit your manuscripts at
<http://www.hindawi.com>

