Ann. For. Res. 65(1): 5572-5586, 2022
ISSN: 18448135, 20652445

ANNALS OF FOREST RESEARCH
https://www.e-afr.org/

DETECTING AND TRACKING MOVING OBJECTS USING STATISTICAL ADAPTIVE THRESHOLDING APPROACH WITH SOM & GMA TRACKING

Bino.T.K[1], Dr.S.Thanga Ramya[2], Dr.K.Ramar[3]
1. Assistant Professor, Immanuel Arasar J J College of Engineering
2. Associate Professor, RMK Engineering College
3. Professor & Principal, RMK College of Engineering and Technology

ABSTRACT

One of the biggest obstacles is finding and tracking moving items in an urban area. This paper offers a standard way for detecting objects in their dynamics so that every object in the scene can be detected in order to handle the challenge. This method uses a static obstacle map and statistical adaptive thresholding. On the basis of their determined threshold and likelihood, those objects were then divided into static and dynamic categories. It is therefore possible to track the resulting dynamic objects using a geometric model-free approach (GMA). When compared to the previous methods, which either used a static obstacle map or statistical adaptive thresholding, the accuracy has greatly increased.

Keyword: Static Obstacle Map, Geometric Model-free, Statistical Adaptive Thresholding

1. INTRODUCTION

Data broadcast from security cameras is managed by video surveillance software, which also controls monitoring equipment. Automatic object detection processes based on statistical and geometric features are used for object detection, which not only demands precise classification of items in images but also accurate positioning of objects. Once the initial position of the target object is known, object tracking refers to the capability to estimate or forecast the position of the target object in each subsequent frame of a video. Both recorded films and a live feed directly from various kinds of cameras can use the object recognition and tracking functionality. Using the camera for live-feed video, using previously recorded video, using an in-out tracker, and object detection are all examples of object detection and tracking. To track movement, a set of bounding boxes for a person and their associated centroids are calculated. Next, the Euclidean distance between any new centroids and associated centroids is computed. Each pixel from the Lidar sensor generates four levels of data. i. Range: The distance of the location from the origin of the sensor, estimated using the laser pulse's time of flight, ii. Signal: The intensity of the light that the sensor received at a specific spot. The number of photons of light detected is the signal for Ouster digital Lidar sensors. iii. Near-IR: The amount of photons that were detected at a specific spot that were not generated by the sensor's laser pulse; iv. Reflectivity: The reflectivity of the surface (or item) that the sensor observed.

***Static Obstacle Map*:** Grid technique and counting algorithm are used to construct the static obstacle map. It determines whether there are any immovable objects or obstacles in the scene and avoids them while object detection is taking place. Any further view of the system and its parts is provided by the 3D model known as the Geometric Model Free Approach. It maintains a

correlation between the non-static spots of the subsequent scans and follows moving objects. Each track is discretized after being forecasted using a discrete time Extended Kalman Filter (EKF). Then, using the Global Nearest Neighbor (GNN) algorithm, allocate the clusters to the predicted track after initialising the tracks with clusters that are not assigned to the predicted tracks. The suggested algorithm successfully locates and follows the item at a high rate of speed and relative

2. RELATED WORK

In traffic light signal systems using radar-based target detection and tracking [1], the problems caused by the YOLOv4 algorithm's insensitivity to small objects and low detection precision can be rectified by using the Improved YOLOv4 method, which uses the shallow feature enhancement mechanism and the bounding box uncertainty prediction mechanism. In order to extract features from the network and improve its capability to locate small objects and colour resolution, the shallow feature improvement technique combines two shallow features at various stages with the high-level semantic features generated after two rounds of upsampling. To boost the precision of the forecast, uncertainty is added to the bounding box prediction process by modelling the output coordinates of the prediction.

The detection of the road ahead in road course prediction using deep learning on radar data is the most fundamental issue with autonomous driving. This is often accomplished using rule-based algorithms [2] and radar data. This study suggests using occupancy grids produced by radar sensors to estimate the ego lane's trajectory using deep learning. The method can also measure the predict driving path's dependability at the same time. Utilizing the known ego stance of the vehicle as determined by a high precision positioning sensor, an automatic labelling process is initiated. The automated labelling technique allows for very cost-effective learning data construction.

Instant object recognition in Lidar point clouds [3] is a cutting-edge technique for categorising things in constantly streamed Lidar point clouds collected from urban environments. The framework takes as input the unprocessed 3-D point cloud sequences from the Velodyne HDL-64 Lidar and it extracts all the pedestrians and vehicles nearby the moving sensor. For the purpose of recognising outdoor 3-D urban objects, a comprehensive pipeline has been designed. The technique is tested using actual Lidar data with 1485 items collected from various metropolitan environments.

LIDAR [3],[4] is evolving into one of the most important sensors for the environment perception in autonomous driving applications. In fact, the system's operation relies on distance ranging, which involves scanning the environment with a laser beam to take measurements. LIDAR is the sensor best suited for determining an object's form. But currently, only four measurement layers are available with LIDARs made exclusively for automotive applications. As a result, the objects detection algorithm must rely on a limited number of layers to identify and categorize the exact types of items that are sensed in the road scene. The DATMO algorithm, which classifies objects according to their types based on assumptions,

Moving obstacle identification and determining free space are crucial concerns for autonomous vehicles and driver assistance systems [5] in urban contexts. Uncertainty occurs as a result of ignorance and mistakes when employing Lidar sensors to scan the area in front of the

vehicle. Errors result from inaccurate posture assessment and noisy data, whereas ignorance is brought on by the perception of unfamiliar places. Complexity increases when the Lidar provides information that is both multi-echo and multi-layer. To manage these numerous sources of uncertainty, a framework for an occupancy grid has also been created. This problem can be resolved by using global and local grids projected onto the road surface. Moving objects are handled by the local one, while the mapping is created by the global one. A global fusion with the world-fixed map is carried out once the sensor data is modelled using a credibilist methodology. Comparing this perceptual style to the norm considerably improves performance.

SHOT [6]: The unique characteristics of histograms for surface and texture description are examined together with local 3D descriptors for surface matching. First, we categorise every method now in use into two classes: signatures and histograms. Then, we highlight the crucial problems of the local reference frame's uniqueness and repeatability through both discussions and experiments. Based on these findings, we create a brand-new, all-encompassing approach for surface representation that includes a new 3D descriptor as well as a fresh, discrete, and repeatable local reference frame. Perhaps to strike a better balance between toughness and description, the latter is located at the intersection of signatures and histograms. Experiments employing publicly accessible datasets and range scans obtained using space time stereo provide strong support for our idea.

Intelligent cars can find, follow, and categorise components of a semi-structured outside environments using a sensorial-cooperative architecture [7] based on Lidar and vision technique for vehicle and pedestrian detection and tracking. The data from the in-vehicle employ monocular vision and lidar to complete this assignment. Both the laser space (using a Gaussian Mixture Model classifier) and the vision spaces are used for detection and tracking which can be used for object classification (AdaBoost classifier). By combining the output from two classification algorithms using a Bayesian-sum decision rule, it becomes feasible to classify objects more precisely. Experiments have been done to confirm the suggested architecture's effectiveness.

A overview of the research on tracking, and recognition of vehicles using on-road vision and behaviour comprehension is given in the article Looking at Vehicles on the Road: A Survey of Vision-Based Vehicle Detection, Tracking, and Behavior Analysis [8]. We give an overview of recent research in the literature by putting vision-based vehicle recognition in the context of sensor-based on-road surround analysis. We go into detail on improvements in vehicle detection and investigate monocular, stereo, and active sensor-vision fusion for on-road vehicle identification. We discuss vision-based vehicle tracking in the monocular and stereo-vision domains with an investigation of filtering, estimation, and dynamical models. We discuss the recently developed field of research on intelligent vehicles that focuses on using spatiotemporal data, trajectories, and various features to explain on-road behaviour.

3. SYSTEM ANALYSIS

The current system shows how the integrated perception module, which uses GMFA and SOM construction to detect and track moving objects without a direct correlation between individual points in successive point clouds, is organised. SOM (P(xstatic[k])), xn[k], and Zn[k]

are produced by the integrated perception module from Y[k] and xhost[k]. The probability of static objects in each grid space surrounding the vehicle is referred to as SOM. When the grid is stationary, it is 1, and when it is movable, free, or unknown, it is 0.

Using P(xstatic[k 1]), xhost[k], and xn[k 1], Z n[k 1] is the initial step in predicting SOM. Ymoving[k] is extracted from the current LiDAR point cloud using P(x static[k]), and it is then sent to the GMFA. In comparison to the tracks, the term Z[k] is produced from Ymoving[k] using Euclidean clustering. The clusters are then given to the current tracks for measurement or the development of new tracks. Iterative closest point (ICP) and an extended Kalman filter are used to estimate xn[k] (EKF). According to the estimated speed, the motion condition of the points is divided into Static and Moving categories. The remaining sites, such as those designated as static by anticipated SOM, Yc moving[k], and not included in Z n[k] among Ymoving[k), are regarded as unclassified. A Bayes filter is used to update each grid's static probability, P(x j static[k] = 1), using the motion states of the points that make up each grid. This method's drawback is that it has poor accuracy for high relative speeds of vehicles or low speed pedestrians.

Advantages

- Accuracy for low speed pedestrians can be improved
- Accuracy for high relative speed vehicle can be improved

Objective

- To detect and track the moving object.
- To improve the accuracy of low speed pedestrians
- To improve the accuracy of high relative speed vehicle

4. SYSTEM DESIGN

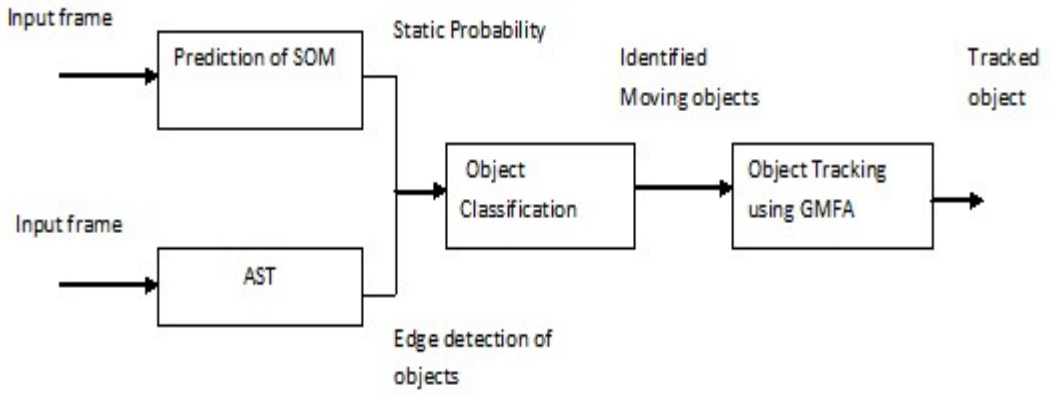


Fig 4.1. Architecture Diagram

Figure 4.1 shows the architecture diagram of the proposed system. From the output of Lidar static obstacles are detected using the prediction of static obstacle map. Initially the probability for static objects is represented within the from 0.05 to 0.95 and the moving objects are separated using Object Classification. Also to the same input apply frame differencing that identify the moving object using Adaptive Statistical Thresholding Method. It uses an automated threshold selection process that uses the standard threshold as the variation in time for each frame. Once the statistical measures of each frame are calculated the threshold is applied to each frame to identify the static and movable object. Once the moving objects were identified track the object using geometric model free approach.

Detection using SOM & AST

SOM is utilised to map the unexplored territory and ascertain the ego vehicle's current state. To represent the immediate area around the ego vehicle, SOM employs a Static and Unknown (including Moving, Occluded, and Free) grid with a static probability range from 0.05 to 0.95. In order to effectively represent static impediments, such
as accident vehicles, scaffolding, and cones, SOM separates the static and moving LiDAR points. If the grids holding the points are a part of a moving object that is close to the ego vehicle, the grids are treated as static without interaction from GMA. Because the points are considered to be moving through contact with GMA, the static probability of the moving object region is decreased.

Local configuration and inertial navigation system predictions are used to configure SOM (INS). It should be mentioned that our study does not put a lot of emphasis on its accuracy. SOM is set to 0.05 during the initialization process.

The steps in constructing the Static Obstacle map are as follows

i. Prediction of Static Obstacle Map
ii. Adaptive Statistical Thresholding
iii. Moving Point Selection
iv. Updated measurement of the static obstacle map.

Prediction of Static Obstacle Map: The ego motion must be rectified in a series of steps based on the condition of the ego vehicle using the approach because SOM reflects the local environment. The distance from the middle of the current j-th grid can be used to identify the eight grids of the previous SOM, l_i where i = 1, 2, 3, 4, 5, 6, 7, 8 while the probability of the j-th grid in the current step was calculated. The anticipated probability of the j-th grid is therefore calculated using the weighted average of the probabilities of the eight grids whose distance is less than 2 d_{grid}. Each grid of SOM has the static probability which is calculated using Bayes filter.

Moving Point Selection: Physically, the j-th grid is similar to the previous grid, if $l_i = 0$. If the object is present in any of the adjacent eight grids in the next state it represents that, it is a moving object or else it is stated as static.

$$L = \sum_{i=1}^{4} l_i^{-1}$$

$$P(\overline{x}^{J}_{static}[k]) = \begin{cases} \sum_i \frac{l_i^{-1}}{L} P(\hat{x}^{i}_{static}[k-1]) & (\mathbf{l_i \neq 0}) \end{cases}$$

$$P(\overline{x}^{j}_{static}[k]) = P(\overline{x}^{i}_{static}[k-1]) \text{ if } (\mathbf{l_i = 0})$$

Where $\bar{x}$ - Estimation of the variable x

k - k^{th} step

L – lack of transition probability

motion, X_{static} (Static = 1, Unknown = 0) SOM state

Updated measurement of the static obstacle map:

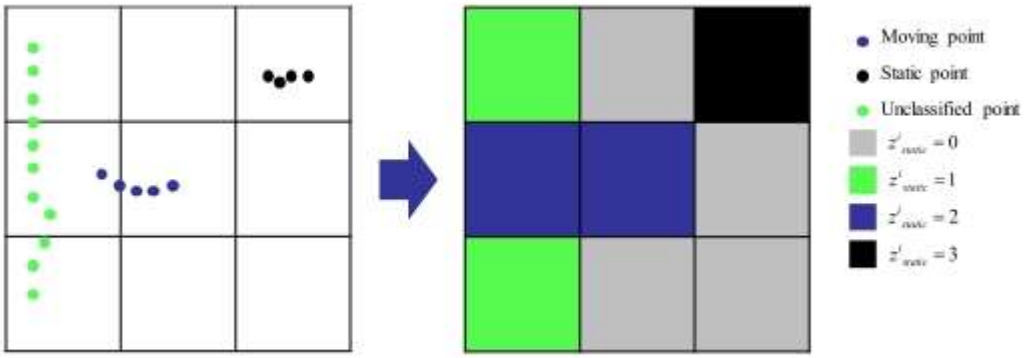


Fig 5.1.1 Measurement Update of motion state of the object

In order to update the likelihood of a stationary object in each grid of the projected SOM, the measurement update uses the motion state of each point recognised by GMA. You can assess the j-th grid using one of the four values provided below:

Free – 0, Unclassified – 1 Moving – 2 and Static – 3

The blue and black dots in the picture represent moving and static object states through GMA, respectively. The unclassified state of motion is indicated by the green dot. Each grid's motion measurement is established as illustrated on the right side when the points on both grids resemble those on the left. The grid's hues of grey, green, blue, and black, respectively, stand for measurements of moving and static motion that are free and unclassified. Each grid can be measured as previously said, and this allows for the calculation of each grid's motion state using

$$P(\hat{x}^{j}_{static}[k] = 1) = P(x^{j}_{static} = 1 \mid z^{j}_{static}[k], \ldots, z^{j}_{static}[0])$$

$$= \frac{p\,z^{j}_{static}[k] \mid x^{j}_{static} = 1)\, p\,\overline{x}^{j}_{static}[k] = 1)}{\sum_{i=0,1} p(z^{j}_{static}[k] \mid x^{j}_{static} = i)\, p(\overline{x}^{j}_{static}[k] = i)}$$

The likelihood is determined in the table below

x^{j}_{static}	z^{j}_{static}			
	Free (0)	**Unclassified (1)**	**Moving (2)**	**Static (3)**
Unknown (=0)	0.30	0.14	0.33	0.23
Static (=1)	0.15	0.40	0.01	0.37

Table: Classification of objects

Adaptive Statistical Thresholding:

The statistical adaptive algorithm's base is the two-frame and three-frame differencing method. Initial parameters for the algorithm are as follows: a starting point or a benchmark frame, Standard threshold: variation in time for each frame

i: the column's index of pixels

j: row pixels' index

k: the RGB pixel index

Using statistical characteristics, we proposed an automated threshold selection process in this method. The applied threshold is applied to each frame of the video sequence once the statistical measures of the mean, standard deviation, and variance have been calculated.

Mean $$\mu_{i,j} = \frac{1}{mn}\sum_{j=0}^{m-1}\sum_{i=0}^{n-1} fm(i,j)$$

Standard Deviation $$\sigma_{i,j} = \frac{1}{mn}\sum_{j=0}^{m-1}\sum_{i=0}^{n-1}\left[fm(i,j) - \mu_{i,j}\right]$$

& Variance $$\sigma^{2}_{i,j} = \left(\sigma_{i,j}\right)^{2}$$

where the indices for the width and height of the fm picture, also known as the current image, are “i” and “j” respectively.

If mean Diff(k) + 2*StdDev Diff(k) > orig video(i,j,k),
then video(i,j,k) = (2^{n}Bits)-1;
otherwise video(i,j,k) = 0;
end

In the subsequent steps, the Geometric Model-Free Approach (GMA) uses $Y_{moving[k]}$ to keep an eye on the moving objects and gauge their conditions.

i. Clustering
ii. Ego-motion Compensation

iii. GMFA Prediction
iv. Track Management
v. Track Initialization
vi. Measurement Update of GMFA

Clustering: Based on Euclidean distance clustering, each point is categorized. The separation between the cluster's mean points and similarity in shape determine the correspondence between the points. Using the Instantaneous Clustering Protocol (ICP), which groups points into single clusters in a parallel manner by first identifying the cluster heads based on cluster head voting and then by minimising the amount of transition, matching is carried out for each cluster after the correspondence has been established. Based on the cluster mean's movement and direction, estimates of the states of moving objects are made by the Extended Kalman Filter (EKF).

Ego-motion Compensation: The fixed global coordinates are converted to the local moving coordinate system through ego-motion compensation.

OeXeYe is a a local moving coordinate system that follows the ego vehicle's rear axle, while OgXgYg is a a fixed global coordinate system.

Seven states are involved: Xn = [Pn,x, Pn,y, n, Vn,x, n, An,x, n]T, Z n,Z n, which expresses the n-th track, and

Pn,x, Pn,y, which denotes the cluster's average location with respect to OeXeYe.

θ_n represents the moving object's yaw angle.

Vn,x represents the direction's speed in relation to OgXgYg.

$A_{n,x}$ denote the acceleration with respect to OgXg Yg, and

γ_n indicates the rate of yaw.

Points earned over more than four steps have been eliminated from the cumulative cluster (Z n, Z n), which is in a queue structure. The identical dynamic states are carried by each point in Z n, Z n. The cluster's geometry in this instance may possibly vary, but the impact would be minimal because LiDAR measures every 80 msecThe prior clusters must be translated to the current step OeXeYe based on the static assumption in order to initialise the tracks and determine the velocities of moving items. This method is known as "ego-motion compensation."

GMFA Prediction: Under the static assumption, Using the speed and yaw rate of the ego vehicle, dead reckoning is used to transform the clusters in the previous step, Z[k], to the current step, Z[k+1].

$$\dot{x}_n = a(x_{n,u}) + q$$
$$= [a1\ a2\ a3\ a4\ a5\ a6\ a7]^T + q$$
$$u = [v_{x,\gamma}]$$
$$a1 = v_{n,x} \cos \theta_n - v_x + P_{n,x} \cdot \gamma$$
$$a2 = v_{n,x} \sin \theta_n - P_{n,x} \cdot \gamma$$
$$a3 = \gamma_n - \gamma\ a4 = a_{n,x}$$
$$a5 = \dot{\gamma}_n\ a6 = -k_a\ a7 = -k_{\dot{\gamma}}$$
$$q \sim (0, Q)$$

Track Management: Track management comprises ceasing the tracks that need to be updated for a predetermined period of time, assigning clusters in the present step to the predicted tracks, and initialising the tracks using clusters that aren't assigned to the expected tracks. Global Nearest Neighbor is used to allocate clusters to the anticipated track (GNN). For each cluster A in the GNN, the feature vector, f, has been previously defined. The clusters' covariance matrices' eigenvalues and a mean point make up the feature vector, a 4D vector. The eigen values show shape information regardless of rotation. In a 4D feature space, the distance between Zn and Yj is characterised as a weighted 2-norm when it is less than a predetermined threshold.

$$f \equiv [x, y\lambda_{MAX}, \lambda_{\min}]^T$$

$$[x, y] = mean(A)$$

$$[\lambda_{MAX}, \lambda_{\min}] = eig(\mathrm{cov}(A)) when \lambda_{MAX} \geq \lambda_{\min}$$

Track Initialization: Once all measurements have been assigned to the projected tracks, the tracks are initialised and discontinued. The track is terminated if it is not updated for three steps in a row or for more than 30% of its lifetime. When a track is initialised, it is done using clusters (Yi Yj) that aren't already associated with a track. A correspondence is created if the distance between Y I and Y j is less than the predetermined threshold, to generate the new track. Accordingly, Yj and Yi become Zm[k] and Zm[k+1].ICP matching is used to initialise the position, yaw, and speed while setting the other variables to zero.

Measurement Update of GMFA: The prediction of the measurement update is performed using the Extended Kalman Filter (EKF) structure. The three measurements made by Zn also contain the yaw angle and position of the moving objects. Zn is expressed as a 3D vector in [hn,1, hn,2, hn,3] T when EKF is measured in Zn.. The moving object's mean position and yaw angle at OeXeYe are each represented by one of the three components of Zn. After Z n and Zn have been matched by ICP, the mean of the matching Z n is calculated using the position of the n-th track. The motion of the item is the direction of the displacement vector from the mean of Z n[k 1] to the mean of matching Z n.

Zn[k] = Hnxn[k] + vn[k]

$$\mathbf{v}_n[k] \sim N(0, \mathbf{V}_n[k])$$

5. EXPERIMENTAL RESULTS:

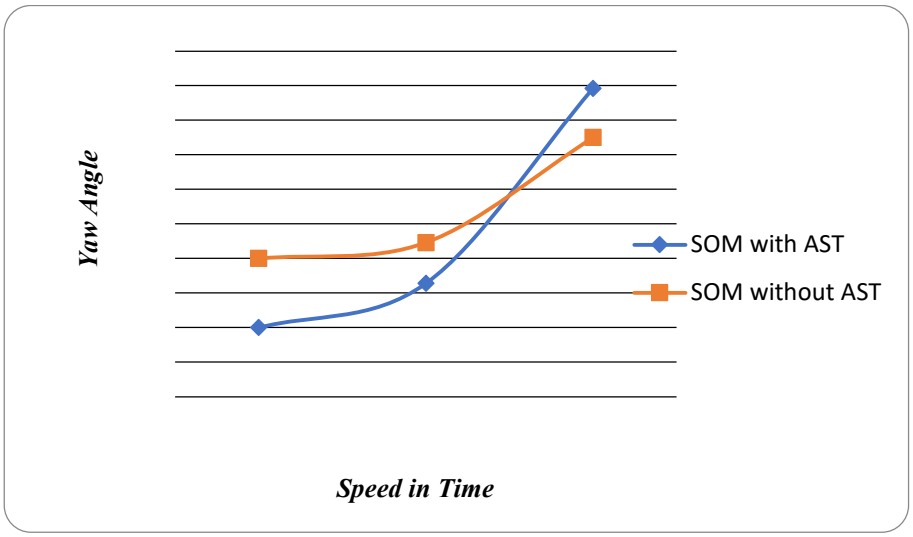
However, the yaw angle estimated in the cluster movement direction relative to the subsequent cluster, which accounts for the better accuracy as compared to GFA as shown in the accompanying table. Because an object was deemed to be moving if it moved more than 1.5 grids at an average distance between consecutive scans of 0.08 seconds, SOM had a grid size of 0.2 metres and was classified as a moving object with a speed of 13.5 kph or greater. The test route featured a range of urban settings, including crosswalks, speed bumps, pocket roads, and intersections. However, the adaptive statical thresholding makes the system as robust by identifying the objects in various acceleration.

The precision and recall of the suggested approach are both boosted, making it around 25% faster than Model Based Tracking (MBT). In this regard, Precision and Recall both improve by 0.109 and 0.242, respectively. A large improvement in recall predicts a decline in the frequency of false-negative results, while an increase in precision suggests a drop in the number of false alarms.

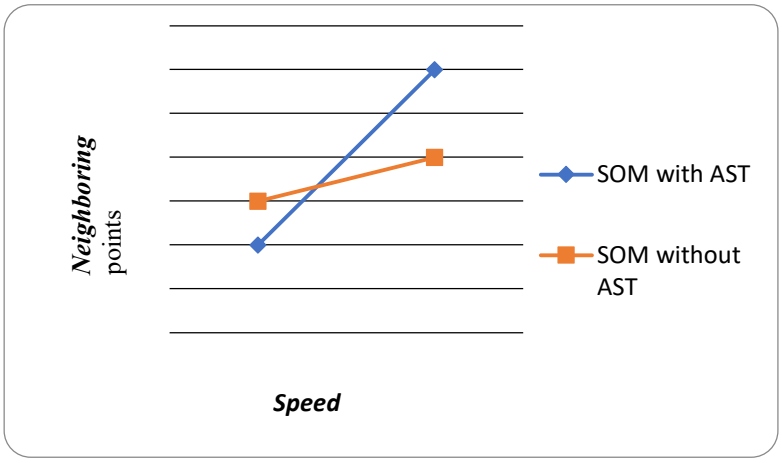
The yaw rate can be calculated using front and rear wheel speeds, front wheel steering angle and some vehicle parameters. Such vehicle parameters can be obtained using speed scheduled Kalman filter. The graph 5.1 shows the error estimation in yaw angle and graph 5.2. shows the neighbouring point detection based on vehicle speed.

Method	Yaw Rate	Lane keeping	Lane changing	Total
GMFA with SOM & AST	Yaw angle(deg)	1.64	2.23	1.81
	Speed(kph)	0.4	0.42	0.4
GMFA with SOM	Yaw angle(deg)	4.46	3.75	4.28
	Speed(kph)	0.48	0.44	0.47

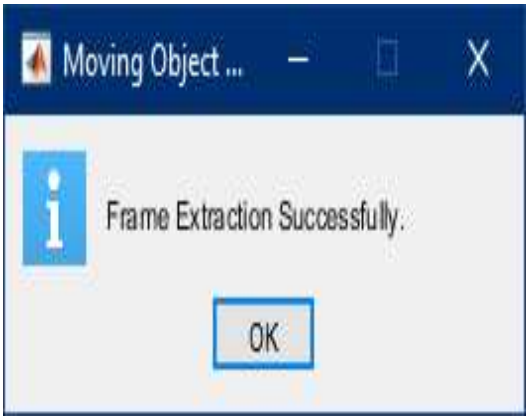
Table 5.1. Yaw Rate Calculation

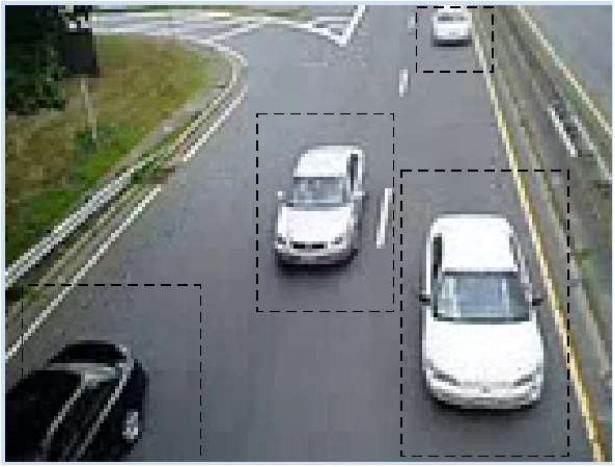


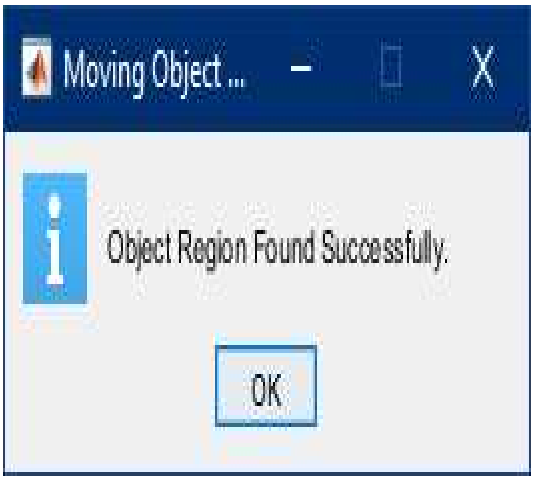
Graph 5.1. Error Estimation in yaw angle



. Graph 5.2. Neighboring points

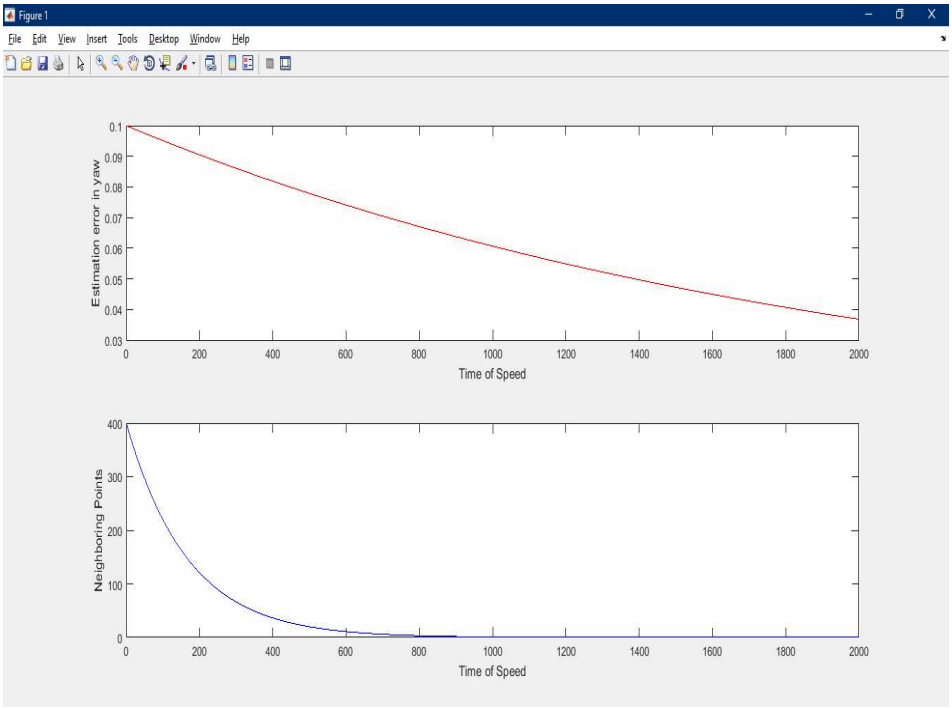

Moving Object ...
Frame Extraction Successfully.
OK




Moving Object ...
Object Region Found Successfully.
OK


Moving Object ...
Static Obstacle Map Successfully.
OK

Fig 5.1. (a) Upload video (b) Frame Extraction (c) Moving object detection (d) Tracking of moving objects



Graph 5.2.Error estimation in yaw angle and neighboring points

CONCLUSION

Adaptive Thresholding approach and the interface between SOM and GMA has been intended to increase the effectiveness of moving object recognition and tracking in the urban environment. By employing SOM to express static objects, including transient stationary impediments, in the urban environment, all points were handled well. The SOM was swiftly and precisely constructed in the local coordinate system by eliminating the interference of the moving objects in conjunction with GMA. Adaptive Thresholding was used to classify the points of

moving objects, whereas SOM and GMA were used to manage and track static impediments. Effectively it was possible to find and follow various moving objects in an urban setting

REFERENCES

1. *Co-operative Systems in Support of Networked Automated Driving by 2030*, AutoNet2030, Amsterdam, The Netherlands, 2014.
2. M. C. Hutchison, J. A. Pautler, and M. A. Smith, "Traffic light signal system using radar-based target detection and tracking," Google Patents 7 821 422, Oct. 26, 2010.
3. T. Giese, J. Klappstein, J. Dickmann, and C. Wohler, "Road course estimation using deep learning on radar data," in *Proc. 18th Int. RadarSymp. (IRS)*, Jun. 2017, pp. 1–7.
4. A. Borcs, B. Nagy, and C. Benedek, "Instant object detection in lidar point clouds," *IEEE Geosci. Remote Sens. Lett.*, vol. 14, no. 7,pp. 992–996, Jul. 2017.
5. V. Magnier, D. Gruyer, and J. Godelle, "Automotive LIDAR objects detection and classification algorithm using the belief theory," in *Proc.IEEE Intell. Vehicles Symp. (IV)*, Jun. 2017, pp. 746–751.
6. J. Moras, V. Cherfaoui, and P. Bonnifait, "Credibilist occupancy grids for vehicle perception in dynamic environments," in *Proc. IEEE Int.Conf. Robot. Autom.*, May 2011, pp. 84–89.
7. S. Salti, F. Tombari, and L. Di Stefano, "SHOT: Unique signatures of histograms for surface and texture description," *Comput. Vis. ImageUnderstand.*, vol. 125, pp. 251–264, Aug. 2014.
8. C. Premebida, G. Monteiro, U. Nunes, and P. Peixoto, "A lidarand vision-based approach for pedestrian and vehicle detection and tracking," in *Proc. IEEE Intell. Transp. Syst. Conf.*, Sep. 2007, pp. 1044–1049.
9. S. Sivaraman and M. M. Trivedi, "Looking at vehicles on the road: A survey of vision-based vehicle detection, tracking, and behavioranalysis,"*IEEE Trans. Intell. Transp. Syst.*, vol. 14, no. 4, pp. 1773–1795, Dec. 2013.
10. S. Ren, K. He, R. Girshick, and J. Sun, "Faster R-CNN: Towards real-time object detection with region proposal networks," IEEE Trans.Pattern Anal. Mach. Intell., vol. 39, no. 6, pp. 1137–1149, Jun. 2017.
11. Kumar, V. Ashok, and N. Sathishkumar. "Effect of fin geometry for various tubes shape using CFD simulation on multi-channel heat exchanger in mobile air conditioning (MAC)." International Journal of Mechanical and Production Engineering Research and Development (IJMPERD) 8 (2018) 2632 (2018).
12. Umesh, K., V. Pravin, and K. Rajagopal. "An approach (performance score) for experimental analysis of exhaust manifold of multi-cylinder SI engine to determine optimum geometry for recreational and commercial vehicles." International Journal of Automobile Engineering Research and Development (IJAuERD) 4 (2014): 2277-4785.
13. Srivastava, Ashish Kumar. "Statistical optimization of wire-edm during processing of hybrid MMC." International Journal of Mechanical and Production Engineering Research and Development (IJMPERD) 8.6 (2018): 783-792.

14. Kaur, Navneet, Meenakshi Sharma, and Pankaj Sharma. "A content based image retrieval using statistical technique." International Journal of Computer Science Engineering and Information Technology Research (2014): 199-202.
15. Kulkarni, A. N. I. T. A., and Elizabeth Rani. "Kalman filter based multiple object tracking system." International Journal of Electronics, Communication & Instrumentation Engineering Research and Development 8.2 (2018): 1-6.
16. DESNITSKY, VLADIMIR VLADIMIROVICH, LYUDMILA VLADIMIROVNA DESNITSKAYA, and I. A. Matveev. "Foundry design scheme." International Journal of Mechanical and Production Engineering Research and Development 10.1 (2020): 657-664.