

SmartEye: Android Application for Traffic Monitoring

Firoz K, Faizal Sulthan, Muhammed Shaik S,
Bezalel V M
UG Scholar: Department of Computer Science
College of Engineering, Perumon
Kollam, India

Muhammed Nizar B K
Assistant Professor: Department of Computer Science
College of Engineering, Perumon
Kollam, India

Abstract—The basic idea behind the project Smart eye which is an android application is to monitor various traffic parameters including vehicle detection and counting, overall traffic analysis, vehicle type, average speed and helmet detection in real time or offline mode. Information collected from the live feed module/local storage system then passed on to the system for later processing and extraction of required features. Final data from the system is stored into the database by sorting data based on the number plate which is considered as a unique primary key. Later users can sort out the stored data based on certain features such as date, time, number plate etc.. This proposed system can replace conventional methods used for traffic surveillance with increased rate of efficiency and computational speed. .

Keywords— *Traffic Surveillance; Real-time monitoring; Android application; Vehicle Detection and analysis; Image processing.*

I. INTRODUCTION

Traffic surveillance is one of the key elements for traffic management, infrastructure, transportation planning and policy, pollution reduction, and forecasting. Thus, considerable efforts are provided to develop technologies in order to understand and improve the issues[1]. Conventional methods lack several limitations such as reduced accuracy and computational speed. In order to overcome the previous limitations a new system wants to be introduced. Our idea is to create an android application for monitoring and processing of various traffic parameters with less complexity and a higher accuracy and computational speed.

The fast increment in human population led to an increase in the number of vehicles thereby making it hard to control the traffic system manually. As a result, there is a tremendous increase in case of road accidents and traffic monitoring analysts want to complete a lot of pre-processing steps for various feature extraction such as number plate detection, analyzing traffic state etc[3]. The need of the traffic surveillance system is to provide construction engineers and other associates to plan in an economic way and proper decisions were taken based on density of the vehicles and other traffic related problems such as road accidents[4]. Along with it, traffic monitoring also helps for traffic planning, analysis and prediction for a particular area, notifying certain law violations etc. Intelligent traffic monitoring systems are the best adaptive solution for the above problems. but most of the conventional methods lack efficiency, computational performance, accuracy rate and

apparently much more time for data analyzing and processing[5].

The basic idea of the project is to develop an android application for monitoring various traffic parameters. Various parameters include vehicle detection, counting and overall traffic analysis. In addition to this various other sub features such as vehicle type, average vehicle speed, helmet detection and number plate detection are also taken into account for overall monitoring. Information/input data into the system can be accumulated from local storage which can later be processed by adding custom selected features along with downloading and viewing processed data from the database. Final output of the system will be displayed with objects tracked in real time and the processed data will be stored in a database according to user preference and can also be retrieved from it for further references.

Chapter I contains common introduction followed by background and previous technologies used were bounded in chapter II. Chapter III illustrates literature review and in chapter IV proposed system and its methodologies were introduced. Chapter V finalizes with conclusion and future studies.

II. BACKGROUND ON TRAFFIC DETECTION SYSTEM AND APPLIED TECHNOLOGIES

The detection of moving object's regions of change in the same image sequence which captured at different intervals is one of interested fields in computer vision. An important large number of applications in diverse disciplines are employed the change detection in its work, such as video surveillance, medical diagnosis and treatment, remote sensing, underwater sensing and civil infrastructure .One of the video surveillance branches is the traffic image analysis which included the moving/motion vehicle detection and segmentation approaches. Even though various research papers have been showed for moving vehicle detection (background subtraction, frame differencing and motion based methods) but still a tough task to detect and segment the vehicles in the dynamic scenes. It consists of three main approaches to detect and segment the vehicle.

A. *Background Subtraction Methods.*

B. *Feature Based Methods.*

C. *Frame Differencing and Motion Based methods.*

A. Background Subtraction Methods

The process of extracting moving foreground objects (input image) from stored background image (static image) or generated background frame from image series (video) is called background subtraction, after that, the extracted information (moving objects) is resulted as the threshold of image differencing. This method is one of widely change detection methods used in vehicle regions detection. The non-adaptively is a drawback which is raised due to the changing in the lighting and the climate situations. So, several researchers work to resolve this drawback by proposed methods on this field. A significant contribution suggested the statistical and parametric based techniques which are used for background subtraction methods; some of these methods used the Gaussian probability distribution model for each pixel in the image. After that, the pixel values updated by the Gaussian probability distribution model these pixel values which are updated from new image in the new image series. Then, each pixel (x,y) in the image is categorized either be a part of the foreground (moving object or called blobs) or background according to adequate amount of knowledge accumulated from the model.

A new method for vehicle detection based on shadows underneath vehicles information. This method extracts the size features of vehicles from information that gathered from the distance between ends of front and rear tires for underneath shadow of vehicles to distinguish the existence of vehicles on the lanes. In this paper, the information represented as traffic movement images which obtained from a camera assembled on a low position such as the roadside, sidewalk, and etc. Moreover, this method has accurate vehicle detection because it is used the functions to generate and improve a background image, in addition, approximate and modernize the value of threshold of background subtraction images binarization automatically.

B. Feature Based Methods

Several approaches can discriminate the object from the background by using its features, a trainable object detection approach. This approach based on learning which employs a set of labeled training data which used for labeling the extracted objects features. In addition, it uses a Haar wavelets technique as feature extraction method and also uses support vector machine classifier for classification process. Moreover, face, people and cars static images datasets have tested on this approach. A sub region is a technique used to locate the local features which used for recognition non occluded and partially occluded vehicles. Principal components analysis (PCA) weight vector used to pattern the low-frequency components and an independent component analysis (ICA) coefficient vector used to pattern the high-frequency components, these two vectors were generating by sub regions. This approach represents a novel statistical method which dependent on local features of three sub regions for detecting the vehicles.

A new traffic criterion detection approach based on Epi-polar Plane Image (EPI). This method treats the noise sensitivity and existence of the rough edge on edge detection through developed a new sobel operator which overcomes the

traditional sobel shortcomings, and the Gabor operator texture edge detection is also used for extracting the features. Low resolution aerial image used as dataset for detection vehicles system, uses the edges of the car body, the edges of the front windshield and the shade as the features for the similarity process. The gathered extracted features knowledge is shaped in the structure of the Bayesian network that will use for integration Signal & Image Processing.

C. Frame Differencing and Motion Based methods

The frame differencing is the process of subtracting two subsequent frames in image series to segment the foreground object (moving object) from the background frame image. Also, the motion segmentation process is another fundamental step in detecting vehicle in image series which is done by isolating the moving objects (blobs) through analyzed and assignment sets of pixels to different classes of objects which based on orientations and speed of their movements from the background of the motion scene image sequence. An intra frame, inter frame and tracking levels are suggested framework to recognize and manipulate occlusion vehicles. This paper showed by quantitative evaluation that the inter frame and inter frame could be used to manage and manipulate mostly of partial occlusions images, and tracking level could be used to manage and manipulate full occlusions images effectively.

Visual-based dimensional approximation is an approach that used to extract motion vehicles from traffic image series and adjust them with a simple disfigured vehicle pattern. In this approach, shadow removing technique is used, in addition to, the experimental tests show an effective performance and sufficient accuracy for general vehicle type classification within the approach mentioned above works on traffic vehicles motion images. A new method based on versatile movement histogram technique for detection of moving vehicles. In this method, two procedures involved to segment and detect the vehicles in video sequence. The first step, a novel background changing method will use for bright changing in video scene. The second step, adaptable movement histogram-based vehicles detection is used, supported and modernized corresponding with movement histogram in the dynamic view.

III. LITERATURE REVIEW

Bouvi'e C, Scharcanski J, and Barcellos P proposed a new system for tracking and counting vehicles in traffic video sequences [13]. Here a new module for vehicle detection and counting based on Particle Filtering is proposed.

A. Obtaining the Background and Particles

It uses videos of 240x320 pixels divided into blocks of 3000 frames. As throughout these blocks no significant changes occur in the background, to detect the background the temporal median at each pixel was used. The particles are obtained by the Minimum Eigen value Method, which is based on identifying corners in the image

For removing particles that belong to background locations does not require any information about background, since the particles that remain static in their spatial positions

according to a threshold are removed as particles associated with the background. The second stage of particle elimination takes into account background information. The color histograms taken from 9x9 pixel windows are compared with the known background colors. Particles with histograms very similar to the background histogram are discarded. For each particle resultant is estimated a motion vector, by applying block matching with exhaustive search. That is, the window centered on each particle is compared with the windows centered on the pixels within a search region in the next frame.

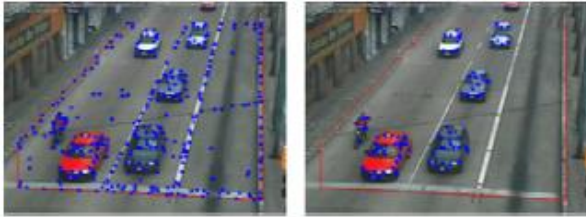


Fig 1 Eigen Value Categorization

B. Grouping Particles

Particles are grouped using the k-means algorithm, and as k-means requires the initial number of clusters, a stage was added to define the number of clusters. Cluster splitting and cluster merging happens similarly.

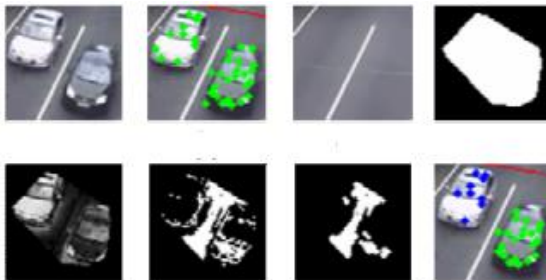


Fig 2 K-means Classification

C. Detection and Tracking of Vehicles

The vehicle detection is based on the shape of the detected clusters, and also uses background information. The vehicle tracking is performed based on the similarity of the color histograms in computed for 9x9 pixel windows centered at the particles belonging to the convex regions formed by the vehicles in the previous frames, shifted by their motion vectors to the current frame. These local color histograms of the particles in the current frame are then compared with the local color histograms of the particles in the previous frame, using the Bhattacharyya coefficient. The particles that have similarity greater than 0.8 with some particles of the previous frame are kept. However, the particles with similarities below this threshold are eliminated, and new convex regions are computed based on the remaining particles.

D. Vehicle Counting

Vehicle counting is performed on virtual loops – areas drawn by the user, where counting occurs – and is based on the intersection of convex regions formed by the particles

tracked with the regions of the virtual loops. The virtual loop that has the largest area of intersection with a given group of tracked particles has its counter incremented.

This paper presented a viable alternative for tracking and counting vehicles based on traffic video sequences. Because this method uses particle motion information, situations in which the traffic flow is interrupted can downgrade the results, but subsequently vehicle tracking and counting is resumed when motion restarts. Occlusion imposes another limitation to our proposed method, but this problem can be handled using another method of particle tracking that is adaptive to vehicle occlusions.

IV. PROPOSED SYSTEMS

In this proposed method an android application for traffic monitoring is introduced for monitoring various parameters such as vehicle speed, type, two wheelers helmet detection, vehicle count and traffic analysis. The overall idea is to identify certain selected traffic parameters in the video stream from an online feed or local storage and thereby storing final processed data which can be manipulated using a unique ID. Here the number plate is considered as the Unique primary key and we can sort out further features which will later be available either in download or viewable format. The user Interface and layer connections, i.e. the front end of the application will be implemented in Android studio platform, software for developing android applications and Java language is implemented for UI development. Different parameters such as vehicle count, traffic analysis, vehicle type, speed and helmet detection will be converted into different sub modules in the programming section using Java Programming language in Open computer vision library (Open CV) and later these modules were integrated into android studio using java Programming.

Our application “Smart eye” which is developed mainly for monitoring various traffic parameters consists mainly of a login tab, feature selection tab and tab for additional processing of selected features. The login tab consists of username, user id and password field which redirects the user into the next feature selection tab. An additional feature is also added in this tab for new user registration. In the feature selection tab, various buttons are provided for various feature selection such as vehicle count and analysis, helmet detection, vehicle type, average speed detection and database checking based on certain inputs. The user can feed live or offline data into the system by enabling or disabling the toggle button. Users can also store data into the database according to his/her preference using this toggle button feature. In the next page, for processing the selected data consist of additional features like setting parameters, custom selection of multiple features, showing database results etc... There is also a database searching and sorting feature for selecting a particular data based on particular input. User can logout from his account using the logout button and the application can be closed using the Exit button. System compatibility includes Android studio (4.0) and above, Windows 10 (4 GB RAM), Open CV modules (new version), Tensor flow 2.5.0, Tensor flow object detection API, Flask 2.0.1 and Numpy 1.19.4. The work is being implemented by using python as backend and java for frontend android app.

Particular user is identified by providing login credentials and output retrieval and storage is limited to that user.

- Initially, users want to upload selected video into the cloud server. Later the url from the server is passed into Flask, which is a python web framework.
- Feature extraction is based on the command selection from the user, which is mainly integrated using android studio.
- Each selection involves APIs which directly invoke methods defined in Tensor Flow which is trained using COCO dataset.
- After successful processing, output will be reflected in the UI and from there user can download or view processed result.

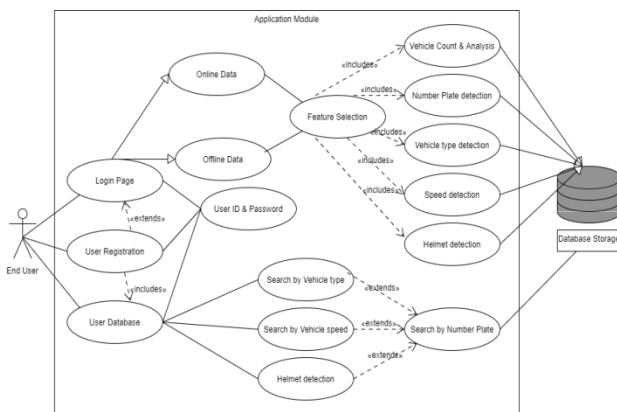


Fig 3 Use Case Diagram

UML Use case diagram is a representation of a user’s interaction with the system that shows the relationship between the user and the different cases in which the user is involved. Here the actors are End user and Database and the system is the application module and the link shows the association of the user with each use case. User linked with a login page which extends user registration including user database along with user ID and password. An extended relationship specifies how the behavior of the extension use case can be inserted into behavior defined for best use case Data insertion then linked with feature selection which includes vehicle count and analysis, number plate detection, vehicle type detection, speed detection and helmet detection. An include relationship defines that a use case contains the behavior defined in another use case. The result of features selected finally stored into the database and database is then linked with searching of vehicles which extends selection based on vehicle type, speed and number plate. The result of searching and classification is finally linked with the end user through a user database for data manipulation and future reference.

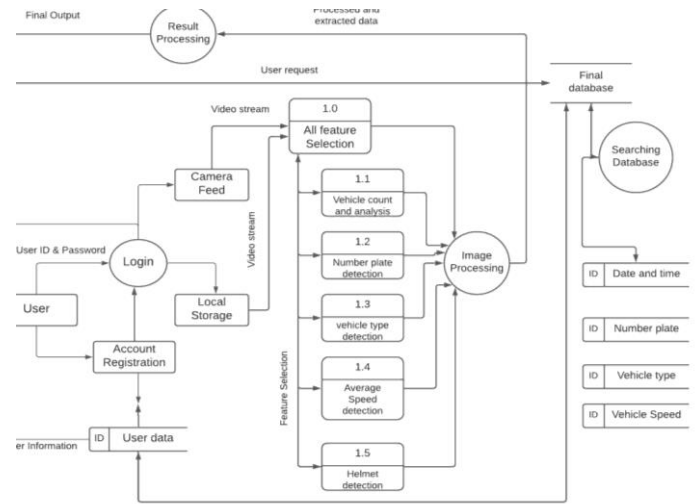


Fig 4 DFD Diagram

In the above DFD diagram each module was categorized into different processes and user and camera feed is categorized into external agents. User enters into the system using login ID and password and user identity is processed with the user database and is confirmed. At the same time user registration processes new user data into the database. Input into the system either through live camera or local storage then forwards into feature selection. Feature selection is then subcategorized into 5 modules such as vehicle count and analysis, Number plate, vehicle type and speed detection and selected data's will be passed on for later processing such as image processing, background subtraction, object tracking etc. After the result processing stage, final data is stored into a database which is linked with the user database for further reference for users and for identification and categorization of vehicles through number plate, date and time, vehicle type and average speed.

Input data can be feed to the system through any cloud medium which then passes the input to the main server for processing of required feature . With use of various ML algorithms each require involves invoking of API s which then turns final processing. Later processed result passed onto the user interface for storage and other manipulation purposes.



Fig 5 Vehicle counting ,type categorization and average speed calculation

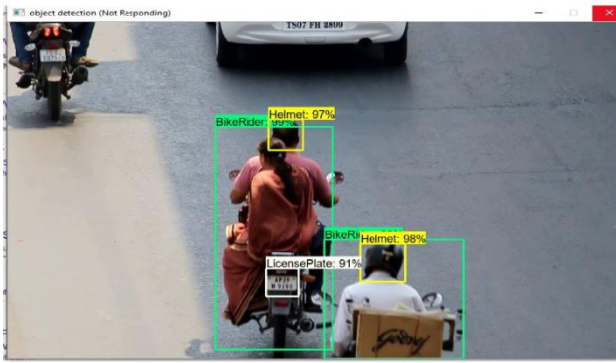


Fig 6 Helmet Detection



Fig 7 Number Plate Detection

V. CONCLUSION AND FUTURE STUDY

Smart eye mainly aims to monitor traffic parameters with more accuracy and high computational speed. Therefore, algorithms used in each module will be much better than from which is taken into consideration. Our application can detect and track a moving vehicle in real time based on unique identity of the vehicle such as number plate by providing it as an input parameter for database searching which is very helpful for vehicle theft detection, crime analysis etc...In this application includes several features such as helmet detection, speed detection which are helpful for notifying certain law violations and can particularly identify the vehicle using number plate. The traffic data stored in the database of our application for a particular area can be later referenced for town planning, road construction, advertisement purposes etc. Apart from conventional methods, smart eye is a mobile application by which we can operate it from anywhere using a smartphone which is also helpful for traffic analysts to pre-process and retrieve data far better than previous methods which include complex steps. Here the vehicle count and vehicle type detection models constitute approximate 82% accuracy. Vehicle speed detection models constitute approx 83% accuracy and vehicle direction and color detection constitute about 89% accuracy. Helmet detection and Number Plate detection constitute about 86% accuracy.

REFERENCES

- [1] C. Toth, W. Suh, V. Elango, R. Sadana, A. Guin, M. Hunter, and R. Guensler. Tablet-based traffic counting application designed to minimize human error. TRB annual meeting, 2013.
- [2] R. Bhatt, M. Lala, A. Deshmukh, S. Lodha, and P. Patil Real time vehicle counting and mapping on Android app. International journal for research in emerging science and technology, 2(4):59-62, April 2015.
- [3] M. C. Narhe and M. S. Nagmode. Vehicle counting using video image processing. International Journal of Computing and Technology, 1(1): 358- 362, Aug. 2015.
- [4] K.Srijongkon, R. Duane Soithong, N. Jindapetch, M. Ikura, and S. Chumpol. SDSoC Based Development of Vehicle Counting System Using Adaptive Background Method. IEEE Regional Symposium on Micro and Nanoelectronics (RSM), 235-238, 2017.
- [5] Fedorov A, Nikolskaia K, Ivanov S, Shepelev V, Minbaleev A. Traffic flow estimation with data from a video surveillance camera. J Big Data. 2019.
- [6] Li C, Dabler G, Feng X, Wang Y. TrackNet: simultaneous object detection and tracking and its application in traffic video analysis. 2019; pp. 1–10. arxiv.org/pdf/1902.01466.pdf.
- [7] Zhang F, Li C, Yang F. Vehicle detection in urban traffic surveillance images based on convolutional neural networks with feature concatenation. Sensors. 2019;19(3):594.
- [8] Akshay A, Temizel A, Cetin AE (2007) Camera tamper detection using wavelet analysis for video surveillance. In: IEEE international conference on advanced video and signal based surveillance, AVVS, pp 558–562
- [9] Albashiti AI, Malkawi M, Khasawneh MA, Murad O (2018) A novel neuro-fuzzy model to detect human emotions using different set of vital factors with performance index measure. J Commun Softw Syst 14(1):121–129
- [10] Altamira, Ahmed A (2014) A framework for automatic semantic video annotation. J Multimed Tools Appl 72:1167–1191
- [11] Alessandretti G, Broggi A, Cerri P (2007) Vehicle and guard rail detection using radar and vision data fusion. IEEE Trans Intell Transp Syst 8(1):95–105.
- [12] Barnich O, Van Droogenbroeck M (2011) Vibe: A universal background subtraction algorithm for video sequences. IEEE Trans Image Process 20(6):1709–1724.
- [13] Bouvi'e C, Scharcanski J, Barcellos P, Escouto FL (2013) Tracking and counting vehicles in traffic video sequences using particle filtering. In: 2013 IEEE international instrumentation and measurement technology conference (I2MTC), pp 812–815.