
Vehicle Detection and Tracking Using Inter-Frame Coding

Vaishali Kalaria,

Assistance Professor, I.T. Department, RK University, Gujarat, India

Abstract: - The moving vehicles detection and tracking is a necessity for collision-free navigation. Motion-based detection is challenging due to low signal to noise ratio in natural unstructured environments. This paper aims at real-time analysis to detect and track objects ahead for safety, motion detection, and target tracing. This paper describes of variation in distances between the camera and the objects in different parts of the scene (object depth) in surveillance videos. 'Vicinity Factor', is robust to noise and object segmentation. It can applied to estimate spatial thresholds and object size in different parts of the scene. A comprehensive approach to localizing target Objects in video under various conditions. I have investigated videos of day and night on different types of conditions showing that my approach is robust and effective in dealing with changes in environment and illumination and that real-time processing becomes possible.

Keywords: - *Target Tracking, Object depth, spatial threshold.*

I. INTRODUCTION

Sensing vehicles ahead and traffic situations during driving are important aspects in safe driving, accident avoidance, and automatic driving and pursuit. I designed a system that is capable of identifying vehicles ahead, moving in the same direction such as car, by tracking them continuously with an in-car video camera. The fundamental problem here is to identify vehicles in changing environment and illumination. Although there have been numerous publications on general object recognition and tracking, or a combination of them, not many of these techniques could successfully be applied in real time for in-car video. This paper introduces an effort to design and implement such real-time oriented algorithms and systems that are highly adaptive to the road and traffic scenes based on domain-specific knowledge on road, vehicle, and control.

The process of automatic tracking begins with the identification of moving objects. I use an improved background subtraction method in conjunction with a novel yet simple background model to achieve very good segmentation. Once the moving pixels are identified, it is necessary to cluster these pixels into regions, which I refer to as blobs, so that pixels belonging to a single object are grouped together. Single moving objects are often incorrectly separated into two or more sub-regions because of lack of connectivity between pixels, which usually occurs due to occlusion from other objects (e.g., trees). A blob merging module to merge close-by blobs is implemented. Having established individual moving objects, the next task in tracking is to achieve correspondence between the blobs in one frame and those in the next. Since I work with real-time Video, the algorithm is designed to be robust to real-life tracking issues like occlusion, appearance and disappearance of objects, and abrupt change in speed, location, and orientation of objects. The robust tracking system has been satisfactorily tested on various static camera scenes involving both humans and vehicles. I first perform Motion Segmentation to extract moving blobs in the current frame. Some blobs that are very small and are likely to be noise are deleted. Due to partial occlusion, some moving objects get incorrectly segmented into two or more separate blobs. Using color and position information, such blobs are merged. The Object Tracking module tracks these moving objects over successive frames to generate Object Tracks. Significant improvements are brought into the basic tracking system by using the velocity of moving objects as an important factor in the tracking algorithm.

II. MOTION SEGMENTATION

Segmentation using moving average background model

A standard approach for finding moving objects in still-camera video data is to create a model for the background and compare that model with the frame in question to decide which pixels are likely to be the foreground (moving objects) and which the background. There are various methods for developing background models. My approach is to use a simple average model where the average intensity values for each pixel over a window of N (typically 150) frames is regarded as the background model. The entire background modelling and segmentation process is carried out on grayscale versions of the images. Fast-moving objects do not contribute much to the average intensity value. Thus, the model becomes a reasonable estimate of the background.

The average model is used to identify moving regions and the Secondary Background is used to identify the valid moving object within these regions. The oval region around the car is the moving region. I can differentiate between the car region (D) and background regions (C and E) by using the Secondary Background for regions D (from frame 2), C (from frame 1), and E (from frame 2) in Figure 1.

F

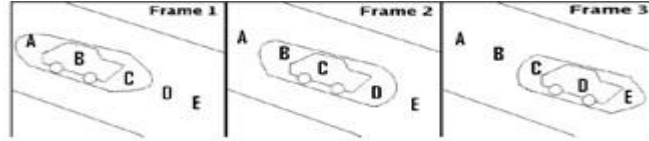


Fig -1 shows illustration of the Secondary Background concept

III. ALGORITHM

Here I mention the algorithm for the vehicle detection.

1. Read the centroid position of each image.
2. Calculate the distance between two centroid images
3. for (present position=initial value: final value) of X resolution
4. for present position=initial value: final value) of Y resolution
5. Calculate change in distance by

$$\text{Distance} = p(X2: X1)^2 + (Y2: Y1)^2$$

Where X1=previous pixel position and X2=present pixel position in width Y1=previous pixel position and Y2=present pixel position in height .Store all the distance values in an Array.

The distance d_{ab} between two normalized color histograms H_a and H_b with N bins are calculated by using the L1 metric as follows:

$$D_{ab} = \sum_i^n |H_a(i) - H_b(i)|$$

$$D_{total} = D_{upperhistogram} + D_{lowerhistogram}$$

$$DS[i] = DS[i] * \frac{N}{C}, \forall i \in [1, \dots, C]$$

$$DS[i] = \frac{DS[i]}{\sum_i^n DS[i]}$$

Where, $DS[i]$ is normalized to have integral unit area.

6. `dataregionprops(L2,BW,{'Centroid','BoundingBox','ConvexHull','Area'});`
7. `rectangle('Position',cardata(3).BoundingBox,'EdgeColor','g','LineWidth',2)`

Tracking is essentially a correspondence task, the goal of which is to find a match for each blob or object in the previous frame to one of the blobs or objects in the current frame. The tracking algorithm is based on the match matrix values and the track assignments were made by looking for the blob which was nearest (in Euclidean distance) to the track.

Since the video shows high amount of noise, occlusion, and sudden appearance and disappearance of objects, a probabilistic track assignment algorithm can lead to better inference about the object behaviour in higher-level intelligence applications. For example, if two tracks have converged into a single blob in a particular frame, a matrix of probabilities of track-to-blob associations would be able to offer a better numerical representation of the situation making it easier to infer the 'merge' event. Similarly, if an object is occluded or leaves the scene, a probabilistic approach can give the probability that the track is 'lost'. The distance-based approach that I described earlier cannot be extended to find the probability that a track has been 'lost'.

$$|I\eta(x) - B\eta(x)| > T\eta(x)$$

Where $T\eta(x)$ an adaptive threshold value is estimated using the image sequence I_0 through $I_{\eta-1}(x)$. The Equation is used to generate the foreground pixel map which represents the foreground regions as a binary array where a 1 corresponds to a foreground pixel and a 0 stands for a backgroundpixel.

Bayesian inference method is used to find matching blobs in consecutive frames to facilitate track correspondence. I also calculate the probability that a track is 'lost' in the current frame. The basic idea behind the approach is that given a track in the previous frame, there is some probability that a blob of similar color and position will be observed in the current frame. Conversely, using the Bayes formula, given the current

blob's color and position observations, I can find the probability of the current blob being associated with a track from the previous frame.

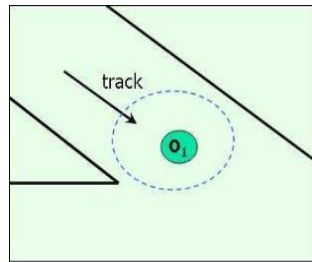


Fig -2 if a blob is seen in current frame, Bayes' formula may be used to associate it with a track from previous frame.

IV. EXPERIMENTAL RESULTS

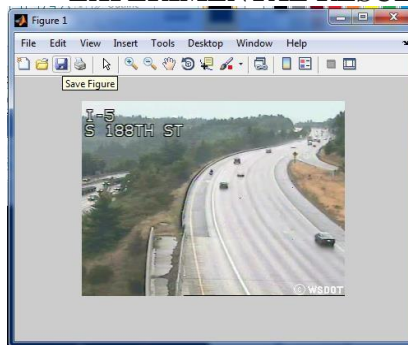


Fig -3 Real-Time Video's Frame

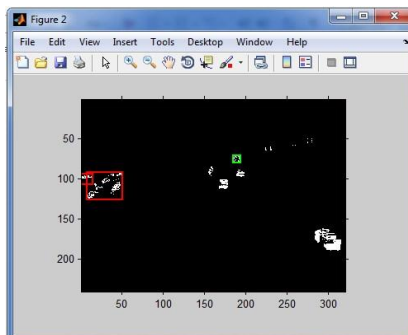
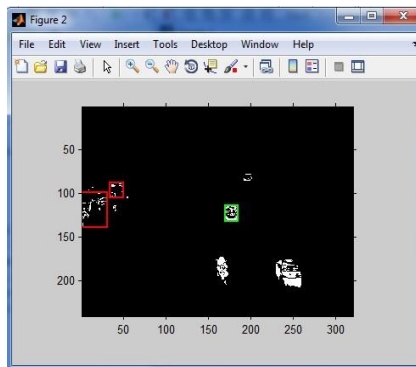


Fig -4 Vehicle Detection and Tracking Demo

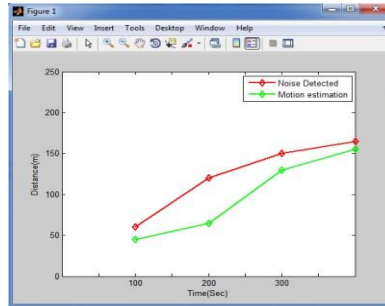


Fig -5 Vehicle tracking using Motion Estimation

Vector Median Factor has been implemented, the tables of section are presented again with the addition of a row for results after the use of the Vicinity Factor. Table shows errors that result in the segmentation when blob merging is not used along with errors that result from the use of blob merging with various thresholds and blob merging with Vicinity Factor.

Percentage Errors (against number of car instances) in segmentation and merging				
[Frames] (Number of instances of cars)	Sequenc e 1	Sequenc e 2	Sequenc e 3	Total 1 + 2+3
No Blob merging	20:7	9:1	24:0	17:9
Blob merging (threshold = 10)	19.8	9:1	19:0	16:0
Blob merging (threshold = 15)	23:3	3.6	15.7	14:2
Blob merging (threshold = 20)	29:3	3.6	16:5	16. 5
Blob merging (Vicinity Factor)	19.8	3.6	16:5	13. 3

Table -1shows errors as a percentage of number of car instances in the sub sequences.

V. CONCLUSION

This paper has focused on an important task to detect and track vehicles ahead with video camera. My approach is mainly based on motion information. Several general features that characterize the vehicles ahead are robustly extracted in the video. Automatic tracking is a very interesting research area that can lead to numerous intelligent applications. My work focused on setting up a system that may be used for intelligent applications in video surveillance of outdoor traffic scenarios. Outdoor videos are more challenging than indoor videos because of illumination changes, non-static backgrounds, and occlusion. The high amount of noise and uncertainty observed in outdoor sequences makes tracking in these sequences a difficult problem to solve. This a tracking system that is able to detect and track moving objects in outdoor video.

REFERENCES

- [1] Luigi Di Stefano, Enrico Viarani,"Vehicle Detection and Tracking Using the Block Matching Algorithm",in IEEE- 2013
- [2] M. Betke and H. Nguyen, "Highway scene analysis from a moving vehicle under deduced visibility conditions," in Proc. IEEE Intell. Vehicle, 2012, pp. 131–136. [3] J. Chu, L. Ji, L. Guo, B. Li, and R. Wang, "Study on method of detecting preceding vehicle based on monocular camera," in Proc. IEEE Intellectual. Vehicle, 2012, pp. 750–755. [4] D. Alonso, L. Salgado, and M. Nieto, "Robust vehicle detection through multidimensional classification for on board video based systems," in Proc. IEEE ICIP, Sep. 2011, vol. 4, pp. 321–324.
- [3] P. Parodi and G. Piccioli, "A feature-based recognition scheme for traffic scenes," in Proc. IEEE Intellectual. Vehicle, 2010, pp. 229–234.
- [4] L. Gao, C. Li, T. Fang, and Z. Xiong, "Vehicle detection based on color and edge information," in Proc. Image Anal. Recognition. vol. 5112, Lect. Notes Computer Science, 2010, pp. 142– 150.

- [5] C. R. Wang and J.-J. Lien, "Automatic vehicle detection using local features—A statistical approach," *IEEE Trans. Intell. Transp. Syst.*, vol. 9, no. 1, pp. 83–96, Mar. 2010.
- [6] W. Zhang, X. Z. Fang, and X. K. Yang, "Moving vehicles segmentation based on Bayesian framework for Gaussian motion model," *Pattern Recognit. Lett.*, vol. 27, no. 9, pp. 956–967, Jul. 2009.
- [7] C. Démonceaux, A. Potelle, and D. Kachi-Akkouche, "Obstacle detection in a road scene based on motion analysis," *IEEE Transport. Vehicle. Technol.*, vol. 53, no. 6, pp. 1649–1656, Nov. 2008.
- [8] S. Sivaraman and M. Trivedi, "A general active-learning framework for on-road vehicle recognition and tracking," *IEEE Trans. Intellectual. Transport. System.*, vol. 11, no. 2, pp. 267–276, Jun. 2008.
- [9] N. Ghosh and B. Bhanu, "Incremental unsupervised three-dimensional vehicle model learning from video," *IEEE Trans. Intellectual. Transp. Syst.*, vol. 11, no. 2, pp. 423–440, Jun. 2006.

BIOGRAPHIES

Vaishali Kalaria received M.E degree in Computer Engineering, Assistant Professor with Department of Computer Engineering, RK University. My research interests includes with Image Processing, pattern reorganization, computer Graphics.