Motion Vector based Abnormal Moving Vehicle Detection in Nighttime

Cuong Nguyen Khac\textsuperscript{1}, Ju H. Park\textsuperscript{2}, Ho-Youl Jung\textsuperscript{3}
\textsuperscript{1,3}Department of Information and Communication Engineering, Yeungnam University, South Korea
\textsuperscript{2}Department of Electrical Engineering, Yeungnam University, South Korea
\textsuperscript{3}Correspondence: Prof. Ho-Youl Jung (hoyoul@yu.ac.kr)

Abstract – Collision prediction during nighttime driving is helpful to avoid accidents because the driver has time to prepare upcoming situations. This research intends to find a real-time solution for frontal and lateral collision warning in an intelligent vehicle. Motion vectors of scene objects are estimated from time-varying frames. Small motion vectors are eliminated by using empirical threshold values pre-determined based on distribution of motion magnitudes. The remaining motion vectors are segmented to rectangular regions by using the unsupervised clustering K-means algorithm. After ROI setting, all segment candidates are classified into vehicle or non-vehicles by using SVM algorithm. From our experiments on real driving situations, the collision risk from lateral and preceding abnormal moving vehicles can be predicted with 91.96% accuracy. The detected abnormal moving vehicles include on-coming, lane change, abrupt speed change, roadside-parking, and overtaking.

Keywords: Collision prediction, motion information, empirical threshold, K-means clustering, SVM classification.

1 Introduction

Over the past decade, a large number of researches have proposed to support drivers in various driving situations.

For intelligent monitoring system (surveillance), an unsupervised model of activity perception by vehicle trajectories has been proposed for vehicle behavior detection [1]. Similarly, abnormal moving vehicles have been identified by other ideas. For example, a novel kernel density estimation approach to vehicle trajectory learning and motion [2], a graph based approach [3], movement string based approach [4], background subtraction and information chain of tracked vehicles analysis [5], local features based approach [6], short-term continuous velocity and trajectory analysis [7], and so on. In surveillance systems, the aforementioned approaches aim to detect automatically the abnormal behaviors of moving vehicles that can lead to collisions or abnormal traffic events.

In case of dynamic scenes, a camera is usually placed behind the windshield of a vehicle. Common types of accidents are rear-end and lateral collisions that take place on roads and freeways. Therefore, it is crucial to recognize the moving vehicle behaviors for collision avoiding. There have been many existing methods that proposed for vehicle behavior detection in various conditions.

Under daytime condition, future behavior of an ego-vehicle in an inner-city environment is predicted using a sequence of elementary states termed behavior primitives [8]. Perspective Transformation and Template Matching are used to detect vehicle behaviors based on road-markings and stop signs [9]. Grey System Theory and a method of car-following behavior in multilane are introduced to detect vehicle behaviors [10]. Under nighttime condition, lots of factors cause collision such as low illumination, moving vehicles with high speed, drowsy and neglecting of drivers, dangerous glare from on-coming vehicles. In [11], HSV color information and magnitude of vector are used to find the warning threshold of dangerous glare. Tail-lights of preceding vehicles are identified by using multi-level image processing algorithms and clustering processing. Then, related distance between the host vehicle and the preceding vehicle are estimated for collision warning [12]. Head-lights of rear moving vehicles are extracted by using the principle of Blob Analysis, DOF (Depth of Field) theory and identify the distance with perspective technique. The distance information is used to obtain the collision warning of overtaking vehicles on two-lane highway while driving during nighttime [13]. Besides, vehicle detection, tracking and behavior analysis are obtained from monocular vision, stereo vision, and active sensor–vision fusion [14].

Especially, driving in nighttime on freeway with non-lane discipline barriers is a high risky task. Our previous work was abnormal driving vehicle detection using motion information and a fixed value of threshold [20]. This paper contributes an improvement of the previous work. The previous practical threshold is extended to low and high threshold values. These thresholds eliminate more effectively the interferences. Moreover, we apply a new idea of ROI (Region of Interest) setting. This ROI is useful to reduce significantly the number of segment candidates. Thus, the number of training samples in training dataset is reduced, also. This idea helps to optimize the training time faster. This paper also performs the evaluation of proposal approach that did not do it in the previous work. The proposal method is
useful for collision warning and increase the concentration of drivers while driving in nighttime.

The rest of the paper is organized as follows. In the next section, we describe the related theories that are Lucas-Kanade sparse optical flow and K-means clustering algorithm. In Section III, we explain the proposal method. In Section IV, we describe the experimental results and evaluation of proposal method. Finally, the paper is concluded in Section V.

2 Related theories

In this paper, the motion vectors of moving objects are estimated by using Pyramidal Lucas-Kanade optical flow algorithm. K-means clustering algorithm is used to group the similar motion vectors together. This section aims to summary briefly the theories of aforementioned algorithms. From that, we can see how these techniques are able to segment moving objects from frame by frame.

2.1 Lucas-Kanade optical flow algorithm

Optical flow is the motion of image points over successive frames. Popular gradient based optical flow techniques were introduced by Horn-Schunck [15] and Lucas-Kanade [16]. The performance evaluation of these optical flow techniques can be found in [17]. The basic idea of Lucas-Kanade is described here. Let \( I(x,y,t) \) is an image or a frame of video, \( m = [x, y]^T \) is a pixel coordinate.

The first assumption of Lucas-Kanade algorithm is that the intensity of pixel \( m \) will not be changed during temporal domain \( dt \). This assumption is represented by below equation

\[
I(x+v_x dt, y+v_y dt, t+dt) = I(x,y,t)
\]  

(1)

The second assumption of Lucas-Kanade algorithm said that an object does not move very far from frame to frame. In another words, object’s motion is change small follow time. The representing of this assumption is described as below

\[
I(x,y,t) + \frac{\partial I}{\partial x} v_x dt + \frac{\partial I}{\partial y} v_y dt + \frac{\partial I}{\partial t} dt + O(dt)^2 = I(x,y,t)
\]  

(2)

Because the higher order terms \( O(dt)^2 \) is very small, then the equation (3) can be simplified as follow

\[
\frac{\partial I}{\partial x} v_x + \frac{\partial I}{\partial y} v_y + \frac{\partial I}{\partial t} = 0 \quad \text{or equivalent form } \nabla I \cdot v + \frac{\partial I}{\partial t} = 0
\]  

(3)

where \( \frac{\partial I}{\partial x}, \frac{\partial I}{\partial y}, \frac{\partial I}{\partial t} \) are difference between successive follow x direction, y direction and temporal derivative of image \( I \), respectively.

And \( \nabla I = \left[ \frac{\partial I}{\partial x}, \frac{\partial I}{\partial y} \right]^T \) is image gradient at pixel \( m \).

The above derivative operations are applied to corner points only. This is also the reason why Lucas-Kanade algorithm is called the sparse optical flow estimation.

The equation (3) is called Lucas-Kanade optical flow constraint. If equation (3) can be solved, then the motion vector of each corner point will be obtained as follow

\[
v_m = \left[ v_x, v_y \right]^T = \left[ \frac{dx}{dt}, \frac{dy}{dt} \right]^T
\]  

(4)

However, equation (3) is single equation with two unknowns. We need more equations to find the motion vector \( v_m \). The third assumption of Lucas-Kanade is used in this case. It said that neighbor points belong to the same object should have similar motions. Thus, this assumption helps to obtain more equations by writing a Lucas-Kanade optical flow constraint for each of point within a small window \( \Omega \) surrounding the considering corner point. After writing more equations, the system becomes over-determined. To solve this system, Least-Squares minimization is used to find the best vector \( v_m \) which is closest to the real solution of the system. Let \( W(m) \) is a window function where \( m \in \Omega \). The idea of Least Squares minimization is solved by finding the min of residual function as follow

\[
\min_{v} E = \sum_{m \in \Omega} W(m) \left( \Delta I \cdot v + \frac{\partial I}{\partial t} \right)^2
\]  

(5)

Writing out the derivatives of residual function follow x and y direction, we have

\[
\frac{\partial E}{\partial v_x} = \sum_{m \in \Omega} W(m) \left( \frac{\partial I}{\partial x} v_x + \frac{\partial I}{\partial y} v_y + \frac{\partial I}{\partial t} \right) = 0
\]  

(6)

\[
\frac{\partial E}{\partial v_y} = \sum_{m \in \Omega} W(m) \left( \frac{\partial I}{\partial x} v_x + \frac{\partial I}{\partial y} v_y + \frac{\partial I}{\partial t} \right) = 0
\]  

(7)

Finally, the equation system need to solve will be

\[
A^T W^2 A v = A^T W^2 b
\]  

(8)

Using algebra, the motion vector for point \( m \) can be obtained as

\[
v = (A^T W^2 A)^{-1} A^T W^2 b
\]  

(9)

where

\[
A = \begin{bmatrix}
\frac{\partial I_1}{\partial x_1} & \frac{\partial I_1}{\partial y_1} \\
\frac{\partial I_2}{\partial x_1} & \frac{\partial I_2}{\partial y_1} \\
\vdots & \vdots \\
\frac{\partial I_N}{\partial x_N} & \frac{\partial I_N}{\partial y_N}
\end{bmatrix}_{N \times 2}
\]

A disadvantage of Lucas-Kanade method is that the flow is described within a small size local window, so the fast moving points cannot be detected. However, this weak point is fixed by using Lucas-Kanade optical flow algorithm and pyramidal technique [18].
2.2 K-means clustering

K-means clustering is an algorithm to group the $N$ objects based on features into $K$ groups ($K < N$). The grouping operation is done by minimizing the sum of squares of distances between data and the corresponding cluster centroid. Mathematically, K-means algorithm clusters $N$ data points into $K$ disjoint subsets $S_j$ containing data points so as to minimize the sum of squares criterion

$$J = \sum_{j=1}^{K} \sum_{n \in S_j} \|x_n - \mu_j\|^2$$  \hspace{1cm} (10)$$

where $x_n$ is a vector representing the $n^{th}$ data point and $\mu_j$ is the geometric centroid of the data points in $S_j$. K-means is the unsupervised clustering algorithm. The data vectors are automatically assigned to clusters without prior knowledge of data. K-means clustering is simple and fast algorithm. Details of K-means clustering algorithm and evaluations of K-means clustering with respect to others clustering algorithms can be found in [19].

3 Proposal method

The proposal flowchart is shown in Fig. 1.

Various videos of real driving situations are recorded for training purpose. Each RGB video frame is converted to grayscale. The gray-scale frame is passed through Shi-Tomasi corner detector to obtain a set of key points per each frame. Each of corner point in frame $f_i$ is considered to determine its new position in which it moves to within frame $f_{i+1}$. If two corner points in frames $f_i$ and $f_{i+1}$ satisfy the Lucas-Kanade optical flow constraint, then the motion vector will be obtained. A sample of raw motion vectors is shown in Fig. 2.

The vehicle regions in one thousand frames are selected manually to build a database of vehicle ground truths. The lengths of motion vectors belong to vehicle regions are considered to find the practical threshold values. All of motion vectors in one thousand frames whose positions locate inside the ground truths are used to compute the average $\mu$ and standard deviation $\sigma$ of length distribution. Then, low and high threshold are obtained as follow

$$\text{low threshold} = \mu - \sigma$$
$$\text{high threshold} = \mu + \sigma$$  \hspace{1cm} (11)$$

Any motion vector has the length shorter than the low threshold or longer than the high threshold is considered as interference. In practice, most of motion vectors locate outside the vehicle regions are eliminated by applying the above proposal threshold. The result is shown in Fig. 3. However, the practical threshold is not strong enough to eliminate all of non-vehicle motion vectors. Therefore, the vehicle regions need to learn by using machine learning.

After applying threshold, the remaining motion vectors are clustered by using unsupervised clustering algorithm K-means. The similar (position and direction) motion vectors are grouped into the same group. Using spatial coordinates of the top-most, bottom-most, left-most, right-most corner point in current frame $f_{i+1}$, each group is represented by a bounding box as shown in Fig. 3. Each bounding region is considered as a segment candidate. The candidates are collected manually into positive samples (vehicle bounding boxes) and negative samples (non-vehicle bounding boxes). The number of samples can be reduced by using a practical Region-of-interest (ROI) setting. All samples are normalization by resizing in the same size. Each sample is represented as a feature vector by using its original RGB color information. The aforementioned dataset is trained by using Support Vector Machine (SVM) to obtain the classifier. This classifier will be used to detect moving vehicles in new videos.

4 Experimental results and Evaluations

The motion vectors of all moving objects are extracted by using Lucas-Kanade optical flow algorithm. The sample result is as follow
After applying the proposal practical threshold, the remaining motion vectors are clustered to obtain the segment candidates. The sample result is as follow.

The ROI setting is considered carefully to reject the obvious interferences from street lamp regions and the preceding part of the host vehicle. The remaining candidates that locate inside the ROI are resized to size of 50×50. These normalization candidates are collected manually from one thousand video frames to build the dataset for SVM training stage. The SVM classifier is used to detect the abnormal moving vehicles in the new video that are shown as follow.

From Table 1, the processing time for each of two successive frames is 0.0127 ms. The rate of the test video is 30 fps. Thus, time to view one frame is 60 ms / 30 fps = 2 ms. Total time for viewing and processing per one frame is 2 ms + 0.0127 ms = 2.0127 ms. Finally, the rate of result video including viewing and processing is 60 ms / 2.0127 ms = 29.8 fps. It is fast for real-time system. Besides, the detection rate is 91.96% (Table 2). The False Positive Rate is 8.04%. In this case, the small motion vehicles also contribute to this 8.04%.

5 Conclusions

This paper proposes a new approach for real-time detection of abnormal moving vehicles in dynamic scene and nighttime driving. The main idea is using motion information of moving objects, practical thresholds and machine learning to classify the abnormal moving vehicles. If a vehicle has strong motion then it can be dangerous with respect to the host vehicle. While other vehicle has no motion or small motion then it is not necessary to detect. The proposal approach in this paper has just detected the strong motion vehicles in ROI regions, only. Based on the experimental results, we can see that most of dangerous moving vehicles that can cause collision to the host vehicles are detected. Whereas the vehicles that have the same motion with the host vehicle are not detected, because they are safe with respect to the host vehicle. The proposal system is useful for lateral and frontal collision warning with respect to the abnormal moving vehicles. Especially, it is more important for drivers who are driving in nighttime and on non-lane discipline barriers freeways.

6 Future works

In Fig. 8, Table 2 and Table 3, we can see the wrong detection results. The main reason is that we used a simple feature extraction method for learning vehicle and non-vehicle regions. In this paper, we used original RGB color information for feature extraction. In the next works, we need to investigate what feature extraction methods are good to fix the error. For example, local feature or gradient based feature and so on. We also need to enhance the evaluation method, because most of non-detected vehicles are normal vehicles. Non-detected vehicles have no motion or small motion. They are not abnormal moving vehicles. They should be eliminated out of the set of ground truths for evaluation.
7 Acknowledgment

This research was supported by Basic Science Research Program through the National Research Foundation of Korea (NRF) funded by the Ministry of Education, Science and Technology (NRF-2011-0011096)

8 References


