A Real Time Accident Detection Framework for Traffic Video Analysis

Hadi Ghahremannezhad, Hang Shi, and Chengjun Liu

New Jersey Institute of Technology
Department of Computer Science
Newark, NJ 07102 USA

Abstract. Traffic accident detection is an important topic in traffic video analysis, and this paper discusses single-vehicle traffic accident detection. Specifically, a novel real-time traffic accident detection framework, which consists of an automated traffic region detection method, a new traffic direction estimation method, and a first-order logic traffic accident detection method, is presented in this paper. First, the traffic region detection method applies the general flow of traffic to detect the location and boundaries of the roads. Second, the traffic direction estimation method estimates the moving direction of the traffic. The rationale for estimating the traffic direction is that the crashed vehicles often make rapid changes of directions. Third, traffic accidents are detected using the first-order logic decision-making system. Experimental results using the real traffic video data show the feasibility of the proposed method. In particular, traffic accidents are detected in real-time in the traffic videos without any false alarms.

Keywords: Accident detection, intelligent transportation, smart cities.

1 Introduction

Vehicle accidents on major roads and highways are one of the main issues in traffic management. It is important to report the accidents immediately when they occur, so that they can be dealt with without much delay. Automatic detection of traffic accidents helps turn traffic back to normal and if needed further medical assistance may be requested in a timely fashion. The term accident on the road may refer to different scenarios, such as rear-end, side-impact, head-on collisions, vehicle rollovers, or single-car accidents. The focus of this study is on single-vehicle accidents when a vehicle strikes a stationary object such as a tree or a barrier on the side of the road. Such incidents are usually caused by the driver losing control of the vehicle and making a sudden turn towards the road-side when there is no turning point.

In order to detect accidents on a highway involving vehicles, the first step is to detect and separate them from the background. Background subtraction methods based on the Gaussian mixture models are statistical techniques that provide a suitable approach to extract the foreground objects with a relatively low time complexity [3]. We apply the Global Foreground Modeling (GFM) method [14] for foreground detection. Note that the GFM method is chosen due to its robustness to noise, efficiency, and its ability to keep the temporarily stopped objects in the foreground model. This is helpful in cases where the vehicles involved in an accident stop on the road after the accident.

After the moving objects are detected, they should be tracked as long as they are present in the scene in order to monitor their behavior and classify specific types of motion patterns. We apply the blob tracking method introduced in [4] for vehicle tracking. Note that this blob tracking method does not always track the vehicle continuously, but it is chosen for real-time vehicle tracking due to its simplicity and low computational complexity. Note also that in the process of accident detection the vehicle only needs to be tracked for a short period of time when it is involved in the accident.

The idea of our proposed real-time single-vehicle traffic accident detection framework analyzes the motion of each vehicle and applies heuristics to decide whether the pattern of movement matches those of single-vehicle accidents. First, the boundaries of the active traffic region are automatically detected using the foreground masks, which is explained in details in section 4.1. Second, the direction and the speed of a vehicle are examined as for a single-vehicle accident to take place, the vehicle should move towards the side of the road with rather high speed. The tracking information is utilized to estimate the direction for each vehicle, which is detailed in section 4.2. The average direction of the vehicles is calculated to estimate the correct moving direction at each part of the active traffic region. Finally, after noticing a vehicle making a sudden turn and moving outside of the traffic region, the variations in speed and neighboring foreground pixels are examined to decide whether a single-vehicle crash has happened. Section 4.3 explains the specific method.

This paper is organized as follows. In section 2, we will outline the previous related work that have approached the problem from various angles of view and mention their differences to our proposed method. In section 3 a statistical foreground modeling method from [14] that has been used to detect the moving and stopped vehicles is reviewed. In section 4 the three parts of the proposed method for traffic accident detection are described in detail. The performance of the proposed method is evaluated on different videos containing a single-vehicle accident in section 5, and we conclude the paper in section 6.

2 Related work

Over the past several decades there have been some studies addressing the issue of vision-based accident detection on roads and highways. Zu et al. [20] use a Gaussian Mixture Model to detect the moving vehicles and the mean shift method for tracking them. In this study three main motion features, namely, velocity, acceleration, and orientation are derived from the trajectories of the tracked vehicles. When all these values exceed the predefined thresholds, an accident is reported. Ren et al. [11] use a modified Gaussian Mixture Model to extract the moving vehicles in aerial videos and after detecting the lanes and dividing each lane into a cluster of cells, some traffic features are extracted for each cell based on the tracking information. Finally, a support vector machine is trained to detect incident points. Traffic parameters include flow rate, average travel speed, and average space occupancy. In [18] a close to real-time approach is proposed that divides each frame into non-overlapping blocks for each of which an average velocity magnitude is calculated and the low-rank matrix approximation is utilized to detect the increase in approximation error. Although this method is more generalizable to different situations it can result in some false alarms. Also, the method can be computationally expensive for higher resolution videos. Maaloul et al. [8] use the Farneback optical flow to extract motion and a statistic heuristic approach to select thresholds and adaptively model traffic flow for accident detection.

Some other studies use more complex methods to detect abnormality in the traffic flow. In [16] vehicle incident analysis is formulated as an optimization problem. An optimal summarization framework is proposed which relies on the salient features of the moving vehicles. This method achieves comparatively good results. However, it suffers from errors in segmentation techniques. Ahmadi et al. [1] use a group sparse topical coding-based technique to model the normal traffic motion using the Lukas-Kanade's optical flow vectors in a document of words. In this model, each word corresponds to velocities in a specific range of orientations and when the computed words do not match the model it means some abnormal motion has happened. This approach is focused mostly on abnormal movement detection and is not specific to a type of accident. In [9] a three-stage approach is proposed to detect car crash incidents. First cars are detected using the You Only Look Once (YOLO) deep learning model. Then after tracking each detected car the Violent Flow (ViF) descriptor is used alongside an SVM to detect car crashes. This approach is not real-time, and there can be some false alarms. Xu et al. [19] present a model for anomaly detection in road traffic by analyzing vehicle motion patterns in static and dynamic modes. In the static mode, the background is subtracted and fed into a Faster R-CNN model for detecting stopped vehicles. In the dynamic mode, the trajectories of vehicles are tracked to find abnormal trajectory which is aberrant from the dominant motion patterns. This method ranked first place on the NVIDIA AI City Challenge. However, it has some limitations due to the use of a supervised deep learning model.

There have been more studies for vision-based traffic accident detection with the use of deep convolutional networks in recent years. In [2] Batanina et al. use a video game to generate synthetic data due to the lack of real videos of car crashes. After training a 3D deep convolutional neural network on the synthetic rendered videos domain adaptation is used to adapt the model to real videos. In [7] Huang et al. an integrated two-stream convolutional network architecture is proposed to detect and track the vehicles in real-time and also detect near-accidents in videos from overhead cameras. Appearance and motion features from the two networks are incorporated to detect near-accidents. Wang et al. [17] train a Yolo v3 model to detect objects in traffic videos that are related to different vehicle crash accidents, e.g., fallen pedestrians/cyclists, vehicle rollovers, and stopped vehicles. Then a decision model is trained based on a set of features developed from the outputs of the Yolo model to detect vehicle crash incidents.

Most of these studies are generally designed to detect abnormal traffic motion which can include stopped vehicles, head-to-head collisions, unexpected congestion, etc. and they are not specific to the type of anomaly. Some methods cannot be applied in real-time due to computational complexity. Also, many of the existing methods rely on supervised data to train a prediction model before they can be applied. In this paper, we present a novel real-time traffic accident detection framework to detect a specific type of road traffic accidents, namely, single-vehicle traffic accidents.

(a) Original frame (b) MOG foreground (c) GFM foreground

Fig. 1. The foreground masks extracted using the MOG method and the GFM method, respectively. Note that the GFM method extracts a more accurate foreground mask with both the moving vehicles (blue) and the stopped vehicles (red) clearly detected in the binary mask. In comparison, the MOG method fails to detect the stopped vehicles.

3 A Statistical Modeling Method for Detecting Both Foreground Objects and Stopped Moving Objects

Vehicle traffic accidents often involve moving vehicles and stopped moving vehicles, as when a traffic accident occurs, a vehicle is initially moving and then stops. Therefore traffic accident detection requires a method that is capable of detecting both foreground objects and stopped moving objects. We introduce in this section a statistical modeling approach that applies the Global Foreground Modeling (GFM) method [13], [14], the Mixture of Gaussian (MOG) method [15], and the Bayes Classifier to detect the foreground objects.

The GFM method models the foreground objects using a mixture of Gaussian distributions. Taking advantage of the fact that the foreground objects appear at different locations in some continuous frames, the GFM method models all the foreground pixels globally. In addition, the GFM method updates its parameters as the video progresses in order to adapt to different foreground objects. The global foreground model is described as follows:

$$P(\mathbf{x}|M_f) = \sum_{k=1}^{K} W_k N(\mathbf{x}|\omega_k) \tag{1}$$

$$N(\mathbf{x}|\omega_k) = \frac{\exp\left\{-\frac{1}{2}(\mathbf{x}-\mu_k)^t \Sigma_k^{-1}(\mathbf{x}-\mu_k)\right\}}{(2\pi)^{d/2} \mid \Sigma_k \mid^{1/2}} \tag{2}$$

$$\sum_{k=1}^{K} W_k = 1 \tag{3}$$

where $\mathbf{x} \in \mathbb{R}^d$ is the feature vector that describes each pixel, M_f means the foreground class, K is the number of Gaussian distributions in the foreground model, W_k is the weight of the k_{th} Gaussian distribution $N(\mathbf{x}|\omega_k)$. μ_k and Σ_k are the mean vector and the covariance matrix of the k_{th} Gaussian density $N(\mathbf{x}|\omega_k)$. Note that every pixel that is classified as foreground is used to update the foreground model $P(\mathbf{x}|M_f)$. The foreground model is called global because it contains all the information of foreground pixels in the frame.

After the global foreground modeling, we also need to estimate a background model. We use the traditional MOG method, which estimates a Gaussian mixture density function for every location in a frame as the background model. The probability density function $P(\mathbf{x}|M_b, L)$ is calculated for location L as described in [15].

In order to classify a pixel into a foreground class or a background class, we apply the Bayes classifier for the classification.

$$p\left(\mathbf{x}|M_f, L\right) P(M_f, L) > p\left(\mathbf{x}|M_b, L\right) P\left(M_b, L\right) \tag{4}$$

For each pixel in a frame, if the inequality 4 holds, the pixel is classified as a foreground pixel. Otherwise, it is classified as a background pixel. Note that the conditional probability density functions $p(\mathbf{x}|M_f, L)$ and $p(\mathbf{x}|M_b, L)$ are estimated using the GFM model and the MOG model, respectively. The prior probabilities $P(M_f, L)$ and $P(M_b, L)$ are estimated using the weights of the MOG model [6].

Fig. 1 shows the foreground masks extracted using the MOG method and the GFM method, respectively. Note that the GFM method extracts a more accurate foreground mask with both the moving vehicles and the stopped vehicles clearly detected in the binary mask. In comparison, the MOG method fails to detect the stopped vehicles.

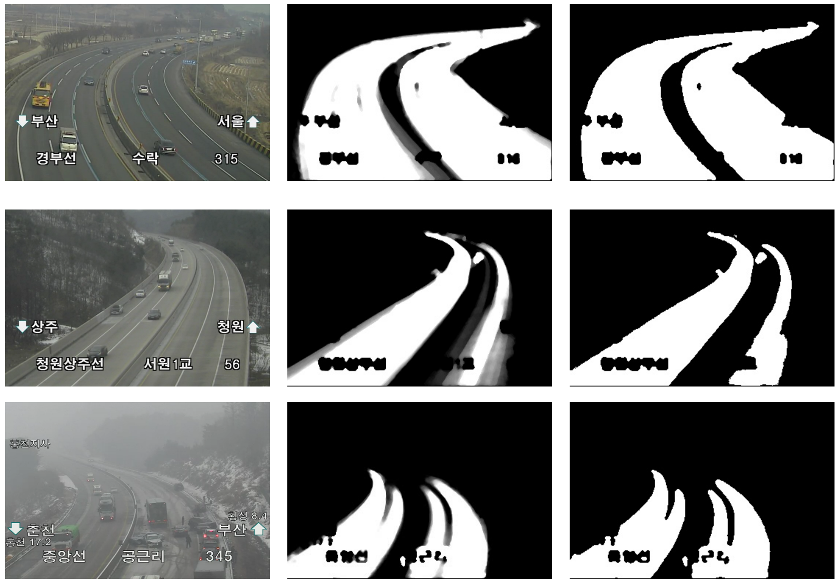
4 A Novel Real Time Traffic Accident Detection Framework

Our proposed real-time single-vehicle traffic accident detection framework consists of three major methods: an automated traffic region detection method, a new traffic direction estimation method, and a traffic accident detection method using the first-order logic. These three methods detect the active traffic region, estimate the traffic direction, and detect the single-vehicle traffic accidents by applying the assumptions about the abnormality of the movement and specific behaviors of a vehicle that lead to crashing to the traffic barrier.

4.1 An Automated Traffic Region Detection Method

We can use the general flow of traffic to obtain some information about the location and boundaries of the roads. When a vehicle is moving along the road it usually follows the right direction within the predefined lanes. The foreground mask from the GFM method contains a white blob for each vehicle. Every time a vehicle passes the road, we add a portion of its mask color to the estimated road. After enough vehicles have passed the road these color values increase to the point that the road section is estimated with a high degree of certainty. By applying the Otsu's threshold [10, 5] we can get rid of noise and the areas outside of the road, that are usually added by the mask of larger vehicles like trucks or buses, and obtain a binary image representing the estimated zone of the traffic flow. The boundaries of the active traffic region becomes closer to the road boundaries as more vehicles pass along the road.

$$r_c = \sum_{f=1}^{T} m_f / F \tag{5}$$



(a) (b) (c)

Fig. 2. Estimation of the active traffic area using the cumulative foreground mask of the GFM method and the Otsu's threshold. (a) The original frame from traffic video. (b) Accumulative foreground mask of the moving vehicles. (c) Estimated traffic region using Otsu's method.

$$r_f = \begin{cases} 1, & \text{if } r_c \geq \tau \\ 0, & \text{otherwise} \end{cases} \tag{6}$$

where r_f is the final traffic region binary map, r_c is the accumulative traffic region map, f is the current frame, τ is the calculated Otsu's threshold, m_f is the foreground mask of frame f, and F is the number of frames.

Some traffic videos are captured by cameras overlooking the highway from one side of the road. In this type of videos, the boundary between two directions may not be accurate since the foreground masks of vehicles in two opposite directions overlap. The foreground mask of vehicles that are closer to the camera can cover the ones of the further side. In such cases, this boundary can be detected by considering the direction of movement of each vehicle to separate the foreground masks. Fig. 2 shows some samples of the traffic area estimation using the cumulative foreground mask of the GFM method and the Otsu's threshold. In the case of videos with shorter lengths, the number of passing vehicles might not be enough to cover the entire surface of the road sections. Also, the stationary captions on the videos block parts of the scene and are always classified as part of the background. Therefore, some gaps can be seen in the map of the traffic region, some of which can be covered as time passes and more vehicles move along the road.

4.2 A New Traffic Direction Estimation Method

The first step after segmenting the foreground and tracking the vehicles is to estimate the traffic direction on the road. Since in many traffic videos the roads are curved we cannot use a single direction for the entire road segment. Therefore, we divide the road into a number of rectangular areas and estimate one traffic direction for each of these areas based on the average direction and magnitude of the moving vehicles for each area of the road. A number of frames (f) are used to estimate the direction of each vehicle. This is done by finding the mean centroid from the first half and the second half of the f frames to estimate a consistent and smooth direction for the movement of each vehicle. The direction and magnitude of each vehicle are estimated as follows:

$$\begin{aligned} v_x &= x_{m_2} - x_{m_1} \\ v_y &= y_{m_2} - y_{m_1} \\ d_i &= arctan(v_y, v_x) \\ m_{v_i} &= \sqrt{v_x^2 + v_y^2} \end{aligned} \tag{7}$$

where v_x and v_y are the components of the velocity vector, x_{m_2} and y_{m_2} are the mean x and y values of the blob centroid in the most recent $f/2$ frames, x_{m_1} and y_{m_1} are the mean x and y values of the blob centroid in the remaining $f/2$ frames, d_i is the estimated direction of the vehicle i, and m_{v_i} is the estimated magnitude of the vehicle i, respectively. Note that we do not consider the slow movements for the direction estimation; since in these cases, the centroids are too close which can lead to faulty results. Furthermore, the vehicles have to be mostly separated and in situations when traffic congestion occurs, average directions are not updated. Therefore, only movements with considerable speed and size are considered for estimating the average direction and the average speed.

After the calculation of the moving direction of the vehicles the average direction and magnitude at each part of the active traffic region can be estimated based on equations 8 and 9 for each frame.

$$avg(c_{k_d}) = (1/F) * \sum_{f=1}^{F} \frac{\sum_{i=1}^{n} d_{i,f}}{N} \tag{8}$$

$$avg(c_{k_m}) = (1/F) * \sum_{f=1}^{F} \frac{\sum_{i=1}^{n} m_{i,f}}{N} \tag{9}$$

where c_{k_d} and c_{k_m} are the direction and magnitude of $cell_k$ respectively, F is the total number of frames, f is the current frame, n is the number of vehicles at $cell_k$, N is the total number of vehicles passed through $cell_k$, and $d_{i,f}$ and $m_{i,f}$ are the direction and magnitude of vehicle i at frame f respectively, which are calculated based on equation 7.

Fig. 3 shows the estimation of the traffic flow direction at each area of the curved road. The size of these areas can be estimated by considering the size of the road section and the average size of the vehicles. To partition the road we used the contour derived

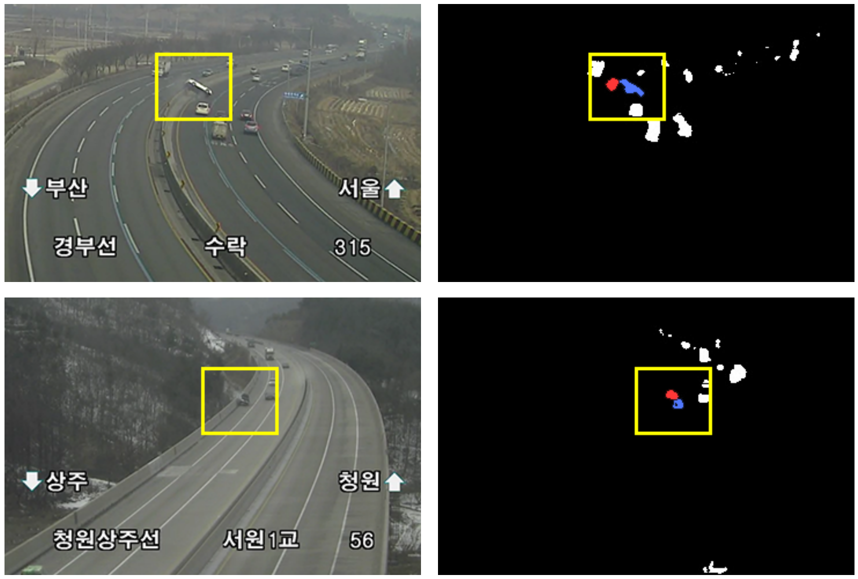
(a) (b) (c)

Fig. 3. Estimation of the traffic flow direction at each area of the curved road. (a) The original frame from a traffic video. (b) The automatically partitioned rectangular areas of the curved road. (c) The estimated average direction and magnitude of the moving vehicles for each part of the road.

from the estimated traffic region map. Multiple squares are drawn over the contour where the area of the contour in each square exceeds a threshold percentage and finally, the squares at the same row are joined together to form a rectangle.

When a vehicle hits the traffic barrier it usually starts with an abrupt movement which is mostly caused by the driver losing control of the vehicle. This rapid movement can be detected by comparing the direction of the moving vehicle and the estimated direction for the area of the road where the vehicle is currently on. If the two degrees differ more than a notable value (d) and the magnitude of the movement is also large, it means that the vehicle is making a sudden unexpected move that often can be dangerous. This kind of hasty movement alone does not necessarily result in the vehicle colliding the traffic barrier or another vehicle.

We should also consider the location of the vehicle after it has made a hasty move. If the vehicle goes out of the estimated boundary of the road without slowing down its speed, there can be two possibilities. Either the vehicle is making a turn to another road that is not detected in road estimation (because there have not been enough cars making a similar turn), or the driver is making a rapid side move which can be due to losing control. For the first case, the vehicle will not crash and will continue its movement and that road will be added to the estimated road map. However, if the vehicle actually hits an obstacle it will most probably have a considerable change in its speed, direction, and acceleration. In some scenarios, this type of accident may also lead to vehicle rollovers. After the traffic accident, the vehicle itself and some of the surrounding vehicles usually stop and traffic congestion occurs. All these cues can help detect a single-vehicle traffic accident. Furthermore, another cue of a single-vehicle collision can be the foreground segmentation mask showing a splash (an unexpected blob detected in the middle of the road) caused by the vehicle hitting the traffic barrier (see Fig.4).



(a) (b)

Fig. 4. Unexpected blob detected in the middle of the foreground mask caused by the vehicle hitting the traffic barrier. The vehicle and the unexpected emerged blob are indicated by blue and red colors respectively. (a) The original frame from traffic video. (b) The foreground mask and the unexpected blob caused by the vehicle crashing the traffic barrier.

4.3 A Traffic Accident Detection Method Using The First-Order Logic

By considering the occurrence of a sequence of steps we are able to detect single-vehicle collisions. To keep track of the target vehicle we can use the stopped vehicle strategy as a factor that makes the assumption more certain. To detect whether the vehicle is stopped, we use the foreground mask from the GFM method which keeps the corresponding foreground information for temporarily stationary objects. Due to the fact that in most cases the vehicle stops after having an accident and there might be some level of congestion and slow traffic flow. In other words, the probability of accident having taken place is high if the same vehicle stops after the abnormal movement and if the nearby vehicles also stop or move at a slow speed. We can make an assumption about an accident occurring after having all these incidents happened in close proximity to each other. Here we consider all these factors in order to decide on the possibility of a single-vehicle traffic accident.

The first step of the proposed method is to estimate the location and boundaries of the two directions of the road by thresholding their accumulative foreground masks. As the number of vehicles passing through different parts of the road grows, the probability of that region belonging to the road increases. This step is useful for having an estimation of the correct traffic zone and the boundaries of the road.

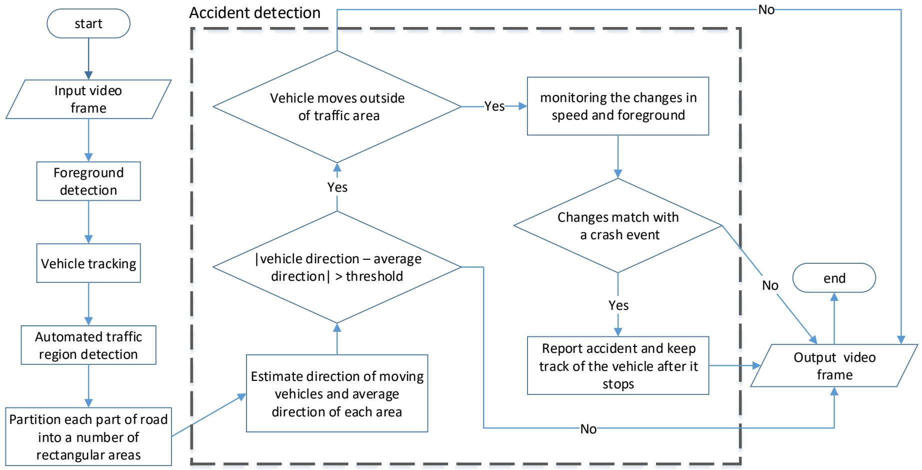


Fig. 5. Flowchart of the proposed real time single-vehicle traffic accident detection framework.

The second step is to partition each part of the road into a number of rectangular areas, each of which has an average direction, average speed, and average blob size. The purpose of dividing the road into different areas is to estimate the direction of the traffic flow at each area of the road. Note that while on straight roads the direction of the traffic flow does not change much, but on curved roads, the direction changes rapidly. Therefore, partitioning the traffic region into smaller areas and assigning a unique average direction and speed to each of them can help improve traffic accident detection accuracy. The rectangular areas on each side of the road are calculated automatically based on the contour of the active traffic region map for that side. Each rectangular area covers the width of the road at the corresponding location and the height of each rectangle is set to be small enough to be reliable even at the curvy parts of the road.

The third step of the proposed method is to detect in real-time single-vehicle traffic accidents. Since crashing the barrier usually starts with an abrupt side-move, the direction of each tracked vehicle (not considering slow vehicles) is compared with the average direction of the corresponding area (part of the road where the vehicle is currently on). If a vehicle makes a rapid side-move, we keep track of that vehicle to see whether it moves out of the road boundaries or the abrupt movement ends earlier. In case the vehicle moves out of the traffic region the changes in speed and neighboring foreground mask are monitored. If the speed decreases suddenly and an unexpected foreground blob appears in the proximity of the vehicle, it indicates that a crash has happened. Fig. 5 shows the flowchart of the proposed real-time single-vehicle traffic accident detection framework.

The idea of our proposed real-time single-vehicle traffic accident detection framework may be expressed using the first-order logic knowledge representation language [12]. In particular, the following statements 10, 11 and 12 represent the idea for traffic accident detection.

$$\begin{aligned}&\forall v Vehicle(v) \wedge Fast(v) \wedge Swerve(v)\\&\wedge \neg ShortDistance(v) \Rightarrow Rapid(v)\end{aligned} \tag{10}$$

where v represents a vehicle, $Vehicle(v)$ means that v is an actual tracked vehicle that is in the current frame, $Fast(v)$ means that the estimated magnitude for v is around the average magnitude of the cell containing its centroid or higher, $Swerve(v)$ indicates that the calculated direction of movement for vehicle v is different from the average direction of the cell containing its centroid by a value more than $45°$. $ShortDistance(v)$ stipulates that the size of movement should not be too small in order to avoid false positives caused by the inaccuracies in the blob detection process. $Rapid(v)$ means that vehicle v has made an abrupt side-move in an unexpected location (see Fig. 7(b)). These types of movements do not always result in the vehicle crashing an obstacle on the side of the road. Therefore, in order to draw a conclusion that an accident has happened more information needs to be considered.

$$\begin{aligned}&\forall v Vehicle(v) \wedge OutOfBoundary(v)\\&\wedge Splash(v) \Rightarrow Crash(v)\end{aligned} \tag{11}$$

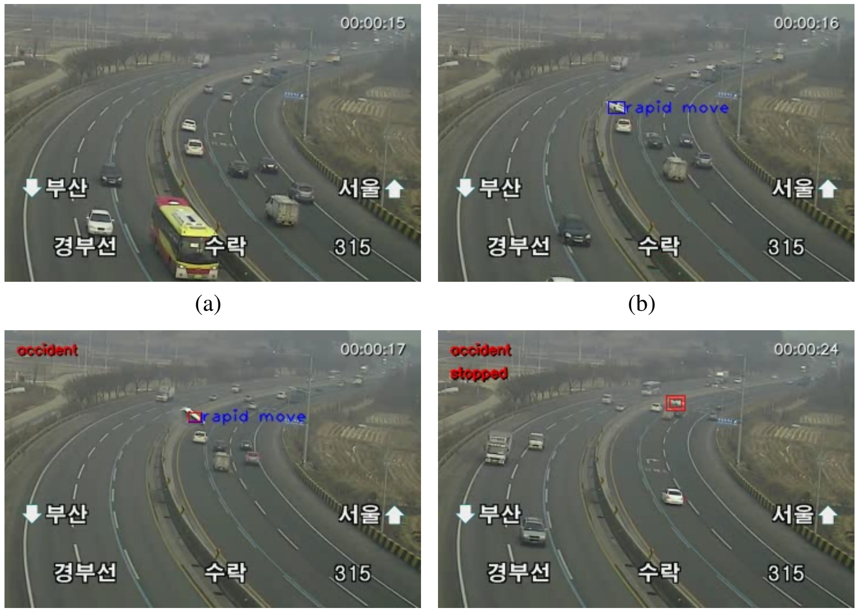
where $OutOfBoundary(v)$ is a predicate which indicates that vehicle v has moved outside of the estimated traffic region, $Splash(v)$ means that there is an unexpected blob in the foreground mask in the surrounding block of vehicle v, and $crash(v)$ means that vehicle v has probably collided with some obstacle on the side of the road.

$$\begin{aligned}&\forall v Vehicle(v) \wedge Rapid(v) \wedge Crash(v)\\&\wedge TimeOf(Rapid(v)) < TimeOf(Crash(v))\\&\Rightarrow Accident(v)\end{aligned} \tag{12}$$

where $TimeOf()$ is a function which returns the time when its input term has occurred, and $Accident(v)$ indicates that vehicle v has had a single-vehicle accident. Therefore, this statement means a single-vehicle crash happens when a vehicle hits the barrier after moving to that direction with a high speed without slowing down during this abrupt movement.

To prove the rules are complete for FOL, we can use the forward chaining method which is complete for a Horn knowledge base. The knowledge base is a set of information representing facts about a particular subject. As for these facts in case of a single-vehicle road-side accident we have assumed if a vehicle makes a sudden turn to the side with high enough speed and long enough moving distance, it has made a dangerous move which we call a rapid move. Also, we assume if a vehicle moves outside of the common traffic region boundaries and at the same time a blob of pixels appear in the foreground mask around the vehicle a collision with an obstacle might have happened which we call crash.

For a single-vehicle road-side accident to occur, the rapid move should happen before the crash. If we consider V_1 to be a vehicle experiencing both incidents in chronological order the occurrence of a single-vehicle accident can be concluded. To use the



(a) (b) (c) (d)

Fig. 6. Real time single-vehicle traffic accident detection results using a real traffic video. (a) Vehicles move in the correct traffic direction. (b) A vehicle makes a sudden side move. (c) The vehicle hits the road barrier. (d) The vehicle stops after the accident.

forward chaining method we can use the known facts to keep proving new information and eventually prove the final clause. Assuming a vehicle V_1 has met all the preconditions of a single-vehicle accident we can use the known facts to prove the accident has occurred.

According to the statement 10 which is in the form of a Horn clause the rapid movement of the vehicle V_1 can be proved by considering four facts from knowledge-base to be true for this vehicle. These facts being that V_1 is a vehicle and it has made a large movement with a high speed. In line with statement 11 which is also in the form of a Horn clause the predefined crash incident can be concluded by considering two more facts from knowledge-base to be true about vehicle V_1 that are moving outside of the traffic region boundaries and occurrence of an unexpected foreground blob around the vehicle. As stated in 12, the resulting clause which is $Accident(V_1)$ can then be concluded by conjunction of the previous Horn clauses with another fact from knowledge based that indicated the right time order.

5 Experiments

We apply real traffic videos from the department of transportation to evaluate our proposed method. The spatial resolution of the traffic videos used in our experiments is

(a) (b) (c) (d)

Fig. 7. Real time single-vehicle traffic accident detection results using a real traffic video. (a) Vehicles moves in the right traffic direction. (b) A vehicle makes a sudden side move. (c) The vehicle hits the road barrier. (d) The vehicle stops after the accident.

720×480 with a frame-rate of 30 frames per second. Specifically, first, the motion information from the videos is used to estimate the road boundaries. Second, the tracking and the foreground segmentation results are applied to detect the abnormal motion. And finally, the first-order logic decision-making system is utilized to detect single-vehicle accidents. The traffic accidents are detected in real-time in the traffic videos without any false alarms. The experiments are implemented using a DELL XPS 8900 PC with a 3.4 GHz processor and 16 GB RAM.

Fig. 6 and Fig. 7 show the experimental results of real-time single-vehicle traffic accident detection using two real traffic videos from the department of transportation. Considering the limitation of video data for the specific type of single-car traffic accidents we only apply our method on two video sequences. In particular, Fig. 6 (a) shows that the vehicles are moving in the right traffic direction in a frame from one traffic video. Fig. 6 (b) shows that a vehicle makes a sudden side move, which is detected automatically by our proposed method. Fig. 6 (c) shows that the vehicle hits the road barrier, and our proposed method automatically detects such a single-vehicle traffic accident in real-time. Fig. 6 (d) shows that the vehicle stops after the accident, and our proposed method automatically detects both the traffic accident and the stopped vehicle in real-time. Fig. 7 shows the real-time traffic accident results using another real traffic video from the department of transportation. Our proposed method successfully detects the vehicle's sudden move to the side of the road, the traffic accident when the vehicle

Table 1. Running time of the proposed method

	video 1	video 2
Length of video (s)	56	56
Running time (s)	44.70	42.26
Number of frames	1680	1680
Running time for each frame (ms)	26	25

hits the road barrier, as well as the stopped vehicles in real-time as shown in Fig. 7 (b), Fig. 7 (c), and Fig. 7 (d), respectively.

Table 1 shows the length (in seconds) of videos, the running time (in seconds) of our proposed method, the number of frames in each of the videos, and the running time (in milliseconds) for each frame. From the table, we can see that our proposed method runs in real-time.

6 Conclusion

We have presented in this paper a novel real-time single-vehicle accident detection method for traffic video analysis. First, we use a statistical foreground modeling method to detect the foreground objects. In order to detect both the moving foreground objects and the temporarily stopped objects, the Global Foreground Modeling (GFM) method is used together with the Mixture of Gaussian (MOG) method. In addition, the Bayes classifier is applied for foreground and background classification. Second, we propose our novel traffic accident detection method. The contributions of our proposed method are three-fold: (i) a new traffic region detection method, (ii) a traffic direction estimation method, and (iii) a single-car run-off-road accident detection method using the first-order logic. The traffic region detection method is used to find out the boundaries of the road. By detecting the road boundaries, we are able to detect the vehicles that hit or go outside the boundaries. The traffic direction estimation method is able to estimate the correct direction of the moving traffic. A vehicle with an abnormal moving direction may lead to a traffic accident. These two methods can provide some clues for detecting a traffic accident. Finally, we use the first-order logic to make a final decision based on these clues. We implement our proposed method and evaluate it using real traffic video data and achieve good performance in real-time traffic accident detection.

Acknowledgments: This work is partially supported by the NSF grant 1647170.

References

1. Ahmadi, P.: Abnormal event detection and localisation in traffic videos based on group sparse topical coding. IET Image Processing **10**, 235–246(11) (March 2016)
2. Batanina, E., Eddine, I., Bekkouch, I.B.I., Khan, A., Khattak, A.M., Bortnikov, M.: Domain adaptation for car accident detection in videos. In: 2019 Ninth International Conference on Image Processing Theory, Tools and Applications (IPTA). pp. 1–6. IEEE (2019)

3. Bouwmans, T., Garcia-Garcia, B.: Background subtraction in real applications: Challenges, current models and future directions. arXiv preprint arXiv:1901.03577 (2019)
4. Chang, F., Chen, C.J., Lu, C.J.: A linear-time component-labeling algorithm using contour tracing technique. Computer Vision and Image Understanding **93**(2), 206–220 (2004)
5. Goh, T.Y., Basah, S.N., Yazid, H., Safar, M.J.A., Saad, F.S.A.: Performance analysis of image thresholding: Otsu technique. Measurement **114**, 298–307 (2018)
6. Hayman, E., Eklundh, J.: Statistical background subtraction for a mobile observer. In: 9th IEEE International Conference on Computer Vision (ICCV 2003), 14-17 October 2003, Nice, France. pp. 67–74. IEEE Computer Society (2003)
7. Huang, X., He, P., Rangarajan, A., Ranka, S.: Intelligent intersection: Two-stream convolutional networks for real-time near-accident detection in traffic video. ACM Transactions on Spatial Algorithms and Systems (TSAS) **6**(2), 1–28 (2020)
8. Maaloul, B., Taleb-Ahmed, A., Niar, S., Harb, N., Valderrama, C.: Adaptive video-based algorithm for accident detection on highways. In: 2017 12th IEEE International Symposium on Industrial Embedded Systems (SIES). pp. 1–6 (June 2017). https://doi.org/10.1109/SIES.2017.7993382
9. Machaca Arceda, V.E., Laura Riveros, E.: Fast car crash detection in video. In: 2018 XLIV Latin American Computer Conference (CLEI). pp. 632–637 (Oct 2018). https://doi.org/10.1109/CLEI.2018.00081
10. Otsu, N.: A threshold selection method from gray-level histograms. IEEE transactions on systems, man, and cybernetics **9**(1), 62–66 (1979)
11. Ren, J.: Detecting and positioning of traffic incidents via video-based analysis of traffic states in a road segment. IET Intelligent Transport Systems **10**, 428–437(9) (August 2016), https://digital-library.theiet.org/content/journals/10.1049/iet-its.2015.0022
12. Russell, S.J., Norvig, P.: Artificial intelligence: a modern approach. Malaysia; Pearson Education Limited, (2016)
13. Shi, H., Liu, C.: A new foreground segmentation method for video analysis in different color spaces. In: 2018 24th International Conference on Pattern Recognition (ICPR). pp. 2899–2904. IEEE (2018)
14. Shi, H., Liu, C.: A new global foreground modeling and local background modeling method for video analysis. In: International Conference on Machine Learning and Data Mining in Pattern Recognition. pp. 49–63. Springer (2018)
15. Stauffer, C., Grimson, W.E.L.: Adaptive background mixture models for real-time tracking. In: The IEEE Conference on Computer Vision and Pattern Recognition (CVPR). p. 2246. IEEE (1999)
16. Thomas, S.S., Gupta, S., Subramanian, V.K.: Event detection on roads using perceptual video summarization. IEEE Transactions on Intelligent Transportation Systems **19**(9), 2944–2954 (Sep 2018). https://doi.org/10.1109/TITS.2017.2769719
17. Wang, C., Dai, Y., Zhou, W., Geng, Y.: A vision-based video crash detection framework for mixed traffic flow environment considering low-visibility condition. Journal of advanced transportation **2020** (2020)
18. Xia, S., Xiong, J., Liu, Y., Li, G.: Vision-based traffic accident detection using matrix approximation. In: 2015 10th Asian Control Conference (ASCC). pp. 1–5 (May 2015). https://doi.org/10.1109/ASCC.2015.7244586
19. Xu, Y., Ouyang, X., Cheng, Y., Yu, S., Xiong, L., Ng, C.C., Pranata, S., Shen, S., Xing, J.: Dual-mode vehicle motion pattern learning for high performance road traffic anomaly detection. In: The IEEE Conference on Computer Vision and Pattern Recognition (CVPR) Workshops (June 2018)
20. Zu, H., Xie, Y., Ma, L., Fu, J.: Vision-based real-time traffic accident detection. In: Proceeding of the 11th World Congress on Intelligent Control and Automation. pp. 1035–1038 (June 2014). https://doi.org/10.1109/WCICA.2014.7052859