Université de Valenciennes et du Hainaut-Cambrésis

Thèse de doctorat

Pour obtenir le grade de Docteur de l'Université de

VALENCIENNES ET DU HAINAUT-CAMBRESIS

et l'université de MONS

Discipline :
Informatique

Présentée et soutenue par Boutheina , MAALOUL.

Soutenue le 16 juillet 2018 à Valenciennes

Ecole doctorale :
Sciences Pour l'Ingénieur (SPI)

Equipe de recherche, Laboratoire :
Laboratoire d'Automatique, de Mécanique et d'Informatique Industrielles et Humaines (LAMIH)

Video-Based Algorithms for Accident Detections

Des algorithmes de détection d'accidents routiers par vidéo surveillance

JURY

Président du jury
- Ruichek, Yassine. Professeur. Université de Valenciennes, France.

Rapporteurs

- Ruichek, Yassine. Professeur. Université de Valenciennes, France.
- Lagadec, Loic. Professeur. ENSTA Bretagne, France.

Examinateurs

- Taleb-Ahmed, Abdelmalik. Professeur. Université de Valenciennes.
- Braeken, An. Professeur. Université de Bruxelles, Belgique.
- Harb, Naim. Docteur, Ingenieur. Satt Nord, France
- Dualibe, Fortunato. Professeur, Université de Mons, Belgique.

Co-directeur de thèse : Niar, Smail. Professeur. Université de Valenciennes.
Co-directeur de thèse : Valderrama, Carlos. Professeur. Université de Mons.

Content

List of abbreviations

EDM	Emergency and Disaster Management
OF	Optical Flow
ROI	Region of Interest
EU	European Commission
BS	Background Subtraction
GMM	Gaussian Mixture Model
SIFT	Scale Invariant Feature Transform descriptor
SURF	Speeded Up Robust Features
HOG	Histogram of Oriented Histogram
SVM	Support Vector Machine
VF	Velocity Feature
PF	Position Features
SF	Area Feature
DF	Direction Feature
C-HMM	Continuous Hidden Markov Model
HFG	Histogram of Flow gradient
SPH	Smoothed Particle Hydrodynamics
BOF	Bag of Feature
ELM	Extreme Learning Machine classifier
CPR	Correct Positive Results
APR	All Positive Results
EPR	Effective Positive

List of Figures

List of tables

List of codes

Abstract

Automatic video surveillance systems have been developed to detect and analyze abnormal behavior or situation of risk in many fields reducing human monitoring of activities captured by cameras (security surveillance, abnormal behavior detection, etc.). One of the applications of video surveillance is the traffic monitoring. Analyzing the motion in roads aims to detect abnormal traffic behavior and sudden events, especially in case of Emergency and Disaster Management (EDM).

Road accidents can cause serious injuries affecting mostly the head and the brain, leading to lifelong disabilities and even death; each additional rescue minute can mean the difference between life and death as revealed by *the golden Hour* [Lerner et al., 2001]. Therefore, providing a rapid assistance for injuries is mandatory. Moreover, if not addressed promptly, accidents may cause traffic jams, eventually leading to more accidents, and even greater loss of lives and properties.

Many cities in France are equipped with video surveillance cameras installed on different roads and highways. Traffic monitoring is done by human operators to visualize the congestion of a road or to measure the flow of the traffic. The video stream of this existing network of cameras is delivered unprocessed to the traffic management center. Thus, there are no video storage of accident scenes. In addition, there is no associated technology for a rapid emergency management. Therefore, it is important to design a system able to organize an effective emergency response. This response should be based, firstly on an automatic detection by video analysis, then, on a rapid notification allowing the optimization of the emergency intervention itinerary without affecting the traffic state. Our work resolves the first part of the emergency response.

The objectives of this thesis are firstly the identification of accident scenarios and the collection of data related to road accident; next, the design and the development of video processing algorithms for the automatic detection of accidents in highways. The developed solutions will use the existing fixed cameras, so as not to require significant

investments in infrastructure. The core of the proposed approaches will focus on the use of the dense Optical Flow (OF) algorithm [Farnebäck, 2003] and heuristic computations for features extraction and accident recognition. The purpose of the dense OF is to estimate the motion of each pixel in a region of interest (ROI) between two given frames. At the output of the dense OF, a dense features could be extracted which is more performant than features extracted at some points. Defining thresholds for accident detection in various environment is very challenging. Therefore, studying the motion at a global scale in the image, allows defining a dynamic thresholds for accident detection using statistic computations. The proposed solution is sufficient and robust to noise and light changing.

Introduction

Video surveillance has become today an essential system established in the majority of the public area (stores, public transport, and roads, etc.) and extended to private spaces (houses and companies) to improve the security monitoring and the EDM. The images obtained with this system are then processed for different detection purposes (lost or abandoned object detection, abnormal behavior detection in crowded places, robbers, controlling access to some building, etc.). Videos may be archived for future use by police or insurance companies to provide evidence. Usually, cameras are connected to a control center, using optical fibers and other connectivity solutions for video transmission, where human operators check the presence of risk situations. However, the operator cannot simultaneously focus on multiple monitors. Therefore, automatic processing, detection and alerting techniques are mandatory to ensure continuous monitoring with accurate information to alert adverse events in a short time. The progress of computer vision techniques allowed the development of different algorithms for automatic detection of different incidents and dangerous situations. The implementation of these algorithms in the computers of the monitoring stations allows informing the operator of the occurrence of a sudden dangerous events. Under these conditions, the operator can trigger the most appropriate response scenario in the shortest possible time as presented in Figure 1.

Nowadays, with the increasing number of vehicles due to the rising number of population and urban areas, traffic monitoring has become one of the most important issues of video surveillance [Bunch et al., 2011]. We distinguish two categories of traffic monitoring. The aim of the first category is the measurement in real time of traffic parameters to insure a fluid traffic and avoid accidents. The second category is abnormal event detection which concerns the detection of unexpected events that may affect the safety of road users and thus improving the rescue of injured people [Fishbain and Yaroslavsky, 2009].

Traffic parameters extraction includes:

- Vehicle counting
- Vehicles tracking: speed detection, vehicle motion analysis
- Traffic jam detection
- Distinguishing the absence and the presence of vehicles on road to compute the traffic density on road
- License plate recognition
- Pedestrian detection on road
- The control of traffic light (for example duration of the green traffic signal depends on traffic density)

For the second category of traffic monitoring, traffic abnormal events include all signs of violation of roads rules and traffic legislation by road users (drivers and pedestrians). The most considered dangerous abnormal situations are accidents where the life of road users must be saved.

Figure 1: An operator supervising some roads in Gironde traffic management center-France[1]

[1] http://www.sudouest.fr/2015/11/02/le-trafic-en-temps-reel-2172644-4778.php

In our research, we are interested in the detection of accidents on the road because of their significant impact on the death rate on especially in motorways.

Road accidents can cause serious injuries leading to lifelong disabilities and even death; each additional rescue minute can mean the difference between life and death as revealed by The Golden hour [Lerner et al., 2001]. It is the hour representing the maximum time to rescue injured from the beginning of accident until they arrive to the operation room. Therefore, providing rapid assistance to injured is mandatory. Moreover, if not addressed promptly, accidents may cause traffic jams and eventually accidents' escalation, especially on highways, where accidents are most likely to occur. A statistic study in France shows that the number of accidents in highways has increased by 25% in 2015[2], while the number of accidents in urban region remains the same.

European Commission(EU)[3], in 2011, has set the goal of halving the number of fatalities caused by traffic accidents. One of the token step is to implement automatic detection systems, based on sensors, for early notification of accidents to authorities and emergency centers.

Existing techniques of automatic accident detection are based on sensors [Parkany and Chi Xi, 2005]. The most used sensors are:

1. Inductive loop detector

It is a low cost sensor fixed in road and the most commonly used traffic collection data. There is many studies for incident detection using this sensor with the application of artificial intelligence [Rossi et al., 2015].

2. Acoustic sensor

It is composed of dipole array of microphone. It detect the presence of a vehicle by their noise allowing vehicles counting and tracking, and the measurement of the

[2] http://www.securiteroutiere.gouv.fr
[3] http://ec.europa.eu/transport/road_safety/topics/serious_injuries/index_en.htm

traffic flow. A sound of an accident could be recognized as presented in [Kim, 2008]. The limitation of these technique is the presence of interference between the sounds of many vehicles.

3. Smartphones

Some applications were developed for smartphones to guide the drivers towards the shortest roads avoiding traffic jams and places where an accident occurs. Some of these applications are based on drivers' information share, by manual entering the information to the application when witnessing an accident. An example of these applications is Wize[4]. Other researchers combine smartphone with accelerometers and acoustic data, to detect and notify emergency centers after an accident and provide situational awareness through photographs, GPS coordinates and VOIP communication [White and all, 2011].

4. Radar

Radars can help detecting a slight increase of a vehicle speed but alone cannot give enough information when accident occurs. Indeed, multi-tracks radars can count the number of vehicles, thus, the road traffic density, what can indicate a possible accident or traffic jam. These systems are combined with cameras to identify the nature of the problem.

5. Ecall system

The EU aims at introducing the eCall system in all vehicles[5]. The concept of the eCall system is based on sensors detecting the occurrence of the accident (for example when the airbag is triggered) and the activation of a SIM card that automatically calls the authorities and emergency services.

[4] https://www.waze.com
[5] http://ec.europa.eu/digital-agenda/ecall-time-saved-lives-saved

6. Video surveillance

The systems mentioned above do not provide sufficient information like surveillance cameras, covering vast areas. Extra information such as the number of injured persons, the number of damaged cars and the severity of the accident help analyzing the traffic situation and making rapid decisions. Traffic cameras provide detailed information on the accident causes, a direct communication infrastructure, and video storage resources that could be used as evidence or for further analysis.

Some industrial companies like Citilog[6], Ipsotek[7], Macq[8] and Dallmeier[9] have deployed accident detection systems based on cameras with frameworks running in centers of traffic controls or using intelligent cameras. They assume that their systems are able to detect accidents in tunnel, bridges, at intersection roads and in highways. Information can be reported after that to the infrastructure. However there is no enough information about their algorithms and there are no meaningful results declared such as the detection rate and the used datasets for algorithms testing, etc. Consequently, we cannot conclude about the reliability of their methodology. In addition their systems are based on their own made intelligent high resolution cameras and are not yet deployed in many cities.

In our study, we consider video surveillance techniques as the most promising technique providing a big amount of information from images with a low cost. Indeed, the detection can be performed using the video flow coming from standard cameras, already mounted on poles above roads.

[6] http://www.citilog.com
[7] http://www.ipsotek.com
[8] http://www.macq.eu/fr_BE/
[9] http://www.dallmeier.com/en/home.html

1. The thesis objectives

Many cities in France are equipped with video surveillance cameras on different roads and highways. Traffic monitoring is done by human operators to visualize the congestion of a road or to measure the traffic flow. The video stream of this existing network of cameras is delivered raw and unprocessed to the traffic management center and is not saved. Thus, there is no video storage of accident scenes. In addition there is no associated technology for emergency management. It is therefore important to design a system for an automatic accident detection and a rapid notification.

The objectives of this thesis is the study and the development of accurate video processing algorithms for the automatic detection of accidents. Our developed approaches will use the stream of existing fixed cameras on roads, so as not to require significant investments in infrastructure. The algorithms will be based on the dense OF for features extraction and heuristic computations for accident detection. The dense OF allows the estimation of the velocity and the orientation of each pixel in the ROI which provides dense information at a global scale. However, the accident recognition is the most challenging part especially in case of the lack of a common dataset. Therefore, static and adaptive dynamic thresholds for accident recognition will be used. The proposed solution must be accurate and robust to noise and light changes.

2. The thesis organization

We present in this section a brief summary of the content of the different chapters detailing our work and contributions:

- **Chapter 1: Abnormal behavior and accidents detection on the road**

  In this chapter, we present the state of the art of the video based abnormal behaviors detection and accidents detection on roads. First we classified the different scenarios of abnormal behavior on road. Then, we analyzed and classified the existing approach of detection. A theoretical and a mathematical explanation of the algorithms was presented also. Our study shows that the approaches based on optical flow computation and learning techniques are more promising. However, since experimental tests of all approaches in literature were not performed on the same datasets, the comparison between the different results is not possible. Therefore, in the next chapter we will focus on developing algorithms based on pixels motion analysis using OF and compare them to an algorithm based on vehicles motion analysis.

- **Chapter 2: Constructive approaches for video-based road accident detection**

We firstly proposed an accident detection approach based on vehicle motion analysis using the kalman filter for cars tracking and a static predefined threshold for accident detection on highways. Then we proposed two other algorithms based on pixels motion analysis using the dense OF for pixels tracking. For the first algorithm we proposed an histogram of orientation computation and a predefined static threshold for the accident recognition part. For the second algorithm, we proposed a traffic modeling approach and a dynamic threshold computation for accident recognition.

- **Chapter 3: Evaluation results and performance analysis**

In this chapter we experimentally defined the values of the different metrics of our proposed approaches. The values of these metrics were changed, correspondent detection rate and the false alarm rate was computed. Consequently, the best fitted values was fixed. Since there is no common benchmarks in compute vision for these topic, experimentations were done on collected videos from websites. Results show that the dense OF combined with a dynamic threshold computation performs a sufficient accident detection rate.

- **Conclusion and future work**

  This final part presents the general conclusion of the thesis then the perspectives and recommendations for new researches topics.

Chapter 1: Abnormal behavior and accident detection on the road

Contents

In this chapter we describe the general steps of abnormal behavior detection on road by video surveillance and we present the more relevant existent approaches in literature. Then we focus on the state of the art of video based accident detection techniques and their results. Finally a comparison of this techniques is proposed based on different criteria's: methodology and used algorithms, accuracy, used datasets, scenarios of detection and time of detection. The limitation of these solutions is discussed.

1. Introduction

Abnormal event detection in videos is one of the important issues for computer vision community. It covers different public areas like shopping centers, crowded spaces, and roads. A quick reporting of these activities with detailed information may help avoiding risky situation, and in case of EDM, providing quick rescue for affected people. In this chapter, we classify abnormal behaviors on road then we focus on road accident detection by video surveillance.

The chapter is organized as follows: Section 2 presents the different scenarios of abnormalities on roads, their classification and a quick review of existent approaches for their detection. In section 3, the different steps of video based accident detection approaches are described and the more relevant finding approaches in literature are summarized. We conclude in Section 4.

2. Abnormal behavior scenarios on roads

2.1. Definitions and classification of abnormal behaviors on the road

A road user (driver and/or pedestrian) behavior is considered as abnormal when the behavior does not respect the roads legislation leading to serious problems such as traffic jams, incidents and accidents. Abnormal behaviors on road are classified in the Table 1 according to our estimation of the severity of each behavior. In case of an abnormal behavior with "*Week Severity*", unlikely crashes between vehicles occur. An abnormal behavior with "*Medium Severity*" may lead to vehicles crashes or pedestrian injury. We consider an abnormal behavior with "*Important Severity*" all kind of vehicles crashes. Existing approaches for video based abnormal behavior detection on road are based on the vehicles or pedestrian motion analysis. Basically, detecting an abnormal motion in a video sequence starts by extracting the targets that we want to keep an eye on (human or objects, etc.) from the video sequence. These targets are tracked in purpose to compare their activities to the predefined models. These models could be a set of characteristics called features of normal activities and/or abnormal activities. The common basic algorithm used in video processing for motion detection, object recognition, tracking and features extraction in roads will be presented in Section 3.

	Week severity	Average severity	Important severity
Vehicle	-Vehicle intrusion on unauthorized roads and structures -Vehicle stopped on the road or on the emergency line -Driver abandoning his vehicle in the road. - Congestion	-Speeding violation -Illegal turns -Wrong way driving	-Collusion between vehicle/pedestrian, vehicle/vehicle, vehicle/ bicycle/ motor. - Car overturned -Fire and smoke in tunnels and roads
Pedestrian		- Pedestrian crossing road borders/entering road zone - Illegal stay of pedestrian on road - Pedestrian moving long the curb. - Pedestrian fighting	

Table 1: Road abnormal behavior events classification

2.2. State of the art of pedestrian abnormal behavior detection

Pedestrian abnormal behavior on road is classified as abnormal behavior with "*Medium Severity*". Consequently, a rapid detection helps saving lives. For instance, [Qianyin et al. 2015] has established a mathematical model of pedestrian abnormal behavior. Firstly, a Background Subtraction (BS) algorithm and a shadow elimination technique were applied to detect and segment all objects in motion on the road. Each object is modeled by an external rectangle. The ratio of this rectangle (the rectangle's height to weight) helps distinguishing a pedestrian from a vehicle. Thereafter, pedestrian was tracked and his trajectories was extracted and compared to the model of pedestrian abnormal behavior. This model was defined by authors. The detected scenarios are: Pedestrian crossing road borders, pedestrian entering road zone, illegal stay on road, pedestrian crossing the road, pedestrian moving long the curb. [Hou et al. 2013] performed background modeling to extract the motion area in videos. Filtering technique was applied on the foreground image to detect people in motion. Their trajectories, then, were detected and analyzed to distinguish between normal behavior and abnormal behavior. In this work, the categories of abnormal behavior detected are fights and damages.

2.3. State of the art of vehicle abnormal behavior detection

The majority of the state of the art of vehicle abnormal behavior detection are based on vehicles trajectories analysis. The general steps of abnormal behavior detection in road are presented in Figure 2. The first step is motion detection and tracking, then feature extraction and finally abnormal behavior detection. For the first step, vehicle in motion are recognized and tracked, for the second step features are mainly the trajectories and/or velocities of vehicles. Different techniques are used to model the normal trajectories which help the detection of unusual motion.

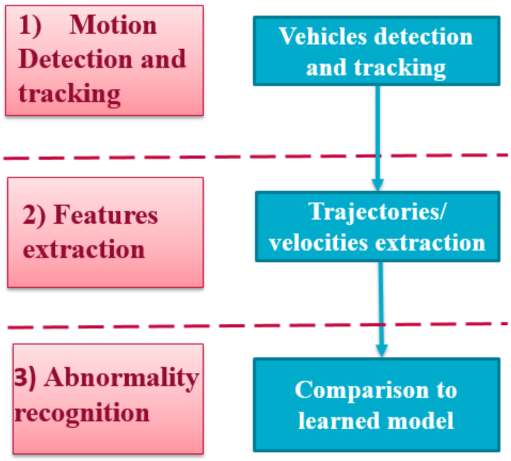


Figure 2: The block diagram of driver abnormal behavior detection on roads

In [Moris and Trivedi, 2008], a study of different approaches of abnormal motion detection technique in different field, including traffic areas, was presented. The authors show that in literature, neural network, iterative optimization, online adaptation, hierarchical method and co-occurrence method are the most used for path learning and normal traffic modeling. They precise that collusion between vehicles at an intersection could be recognized by the detection of the interaction between objects in motion. Figure 3 shows some examples of abnormal trajectories on road presented in this work. Since abnormal detection remains a challenging task, researches continue proposing different approaches. For example, [Cui et al., 2011] used BS and pixel moving velocity computation for motion region classification (pedestrian, vehicle, noise region, etc.). The motions detected are classified using trained local feature' distribution map. A classifier is used to detect abnormal behaviors. [Li et al. 2015] used the local invariant features and the visual codebook approach for image descriptor, where a Gaussian distribution model was used to detect abnormal behavior. [Cai et al. 2015] proposed a new algorithm for trajectory analysis composed by two parts: trajectory learning and online abnormal detection. The Hidden Marcov model was used to define an activity path pattern and abnormality was detected by comparison to normal trajectories. However, [Mehboob et al, 2016] used other features than trajectory extraction for abnormal detection in road. They extract vehicle number in the frame and their mean velocity to detect congestion. Congestion is considered as incident or caused by incident. Fuzzy logic was used to analyze features for the detection.

Figure 3: Examples of vehicle abnormal trajectory [Moris and Trivedi, 2008]

Early sited works are dedicated for multiple scenarios of abnormal motion detection. Some other researchers preferred to focus their effort on developing algorithm for a specific scenario which is vehicle collusion (accident) detection. Collusion detection can be based also on trajectories analysis and/or other features extraction such as velocity, orientation, acceleration, area, etc. The more relevant approaches for accident detection are described in the following section.

3. The general steps of the video based accident detection techniques

The scenario of road accident used in literature concerns the crash between two or many vehicles in different kind of roads including intersection, urban areas and highways. The same as abnormal behavior detection, the different steps of video based road accident detection are: 1) Motion detection, 2) Feature Extraction and then 3) Features Analysis and Accident Recognition. A block diagram of the sequence of steps to detect an accident is shown in Figure 4 with more details. The most frequently used algorithms for motion detection are BS and OF. The features related to the traffic motion and vehicles trajectories can be extracted using different processing techniques such as tracking. Finally, the analysis of these traffic features allows the differentiation of normal motion versus abnormal.

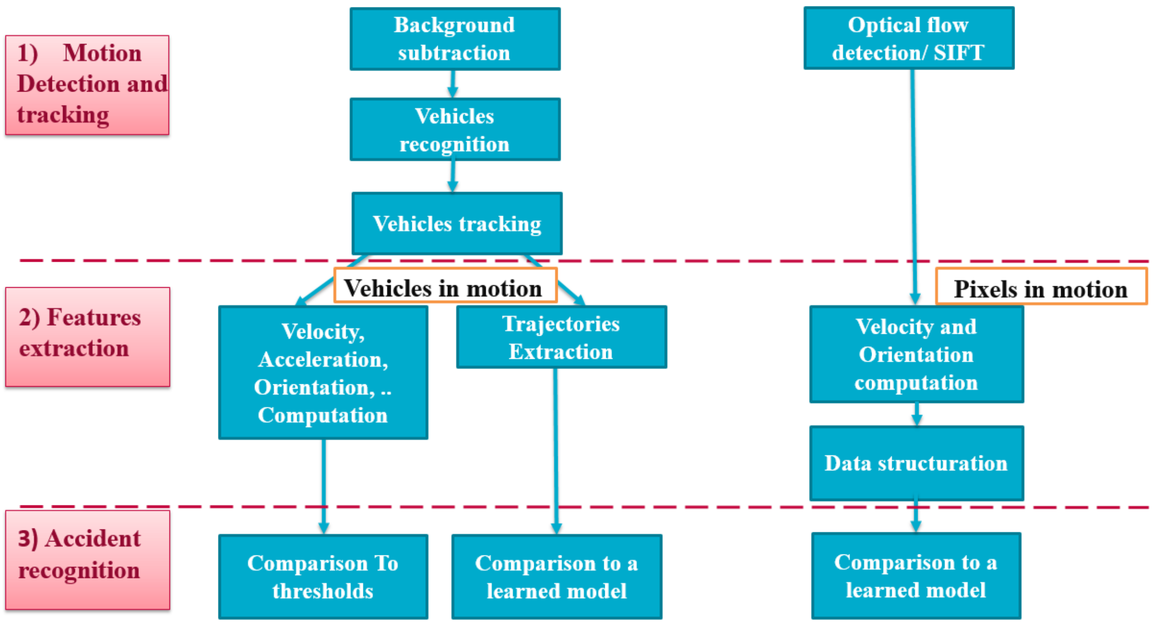


Figure 4: The block diagram of the video based accident detection techniques.

A few researchers dealt with the problem of traffic accident detection by video surveillance. Existing approaches, sited below, follow the sequence of steps listed in Figure 4. However, the video processing techniques used to perform each step are different.

3.1. Motion detection and tracking

The objective of this step is the localization of all moving objects in the image. It can be done using different methods such as BS, OF and SIFT (the Scale Invariant Feature Transform descriptor).

3.1.1. Background subtraction

The objective of the subtraction is to segment the moving objects that are present in the scene. It is used for videos captured by fixed cameras. In order to achieve the motion segmentation, a model of the background scene is necessary. Each image in the video sequence is compared to the background model. The pixels having a significant difference from the model are considered as belonging to moving objects. The output of a BS algorithm is a binary image where the moving objects are presented as a group of white pixels (blobs). The most basic technique for BS is to take an image of the scene with static objects as a background model. The pixels of moving objects are detected by applying a threshold on the absolute difference between the background image and the incoming frame as presented in Figure 5. To update the background image automatically, there is other methods such as the adaptive median approach [McFarlane and Schofield, 1995] which increments or decrement the value of a pixel of the background model depending on the difference between the model and the current image, the temporal median approach [Cucchiara et al., 2001] which computes for each pixel its median value using the last N frames of the video, and the sigma-delta approach based on a simple recursive nonlinear operator with a spatiotemporal regularization algorithm [Manzanera and Richefeu, 2007].

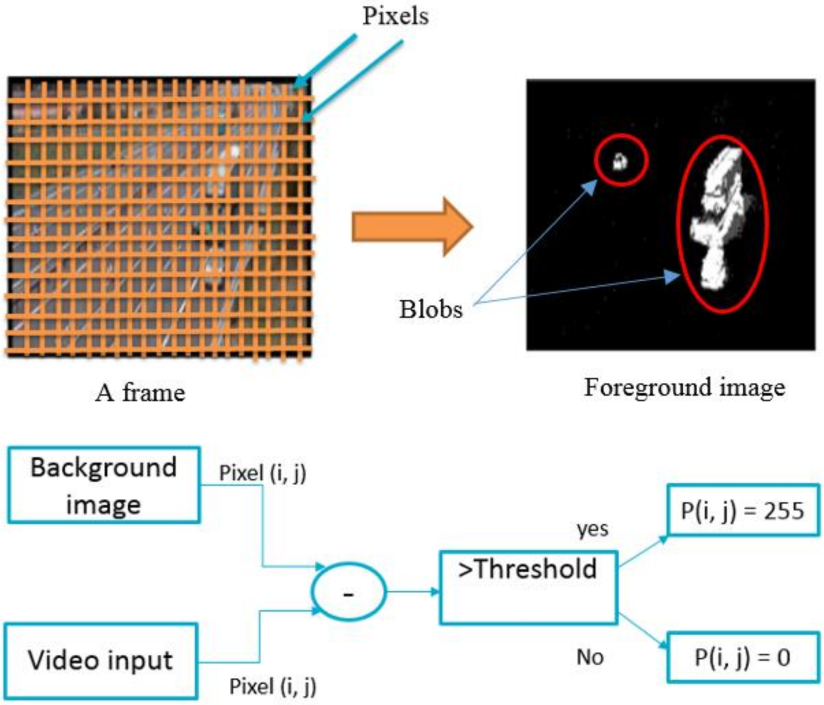


Figure 5: A flowchart of a basic background subtraction

The approaches based on a background images are sensitive to the change of luminosity. As a solution, many approaches based on modeling the background using a statistic distribution of a pixel have been developed. In [Wren et al., 1997], each pixel is modeled by a Gaussian probability density defined by the average color of the pixel and a covariance related to that color. The comparison of the current image to the model is carried out with a log likelihood distance or using the Mahalanobis distance [Benezeth et al., 2008]. The average and variance are updated with each new image. For more accuracy, [Stauffer and Grimson, 1999] used a mixture of Gaussian (GMM) instead of a single Gaussian for background modelling. Other techniques based on Gaussian mixture modeling were developed later such as [KaewTraKulPong and Bowden, 2002], [Zivkovic, 2004] and [Zhao et al, 2012] which are based on Fuzzy GMM and Markov Random Field. However, [Barnich and Van Droogenbroeck, 2011] has developed another method for background modeling named ViBe which consists of modeling each pixel by N samples taken from previous images. A pixel of the current image is considered as belonging to the foreground image when the number of samples of the model inside a circle of radius R centered on the pixel, is greater than a defined number.

Other researchers tried to improve the classic BS algorithm by eliminating noise and shadow. For instance, [Lee, 2012] has developed an advanced shadow elimination technique. An example of shadow elimination is shown in Figure 6.

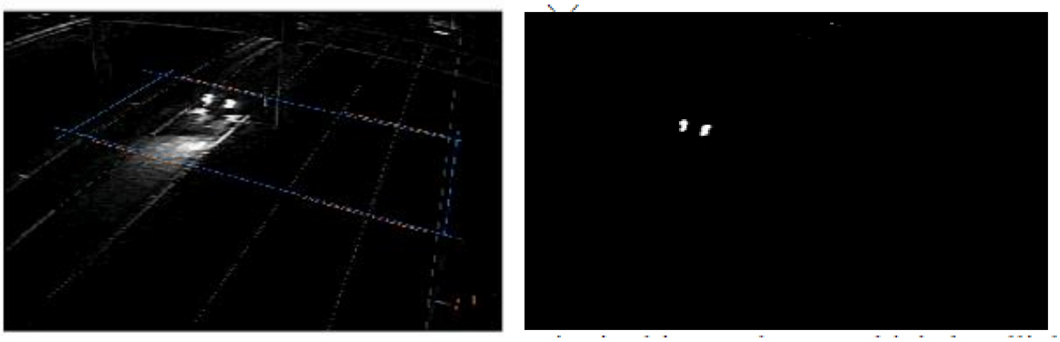


Figure 6: Example of shadow elimination on a traffic scene [Lee, 2012]

3.1.2. The optical flow

The OF is the apparent velocity field observed between two successive frames in a scene. In other words, for each pixel of coordinates (x, y), the OF estimates its position at the next frame. These computations are based on the constant lighting hypothesis. Under this hypothesis, we assume that image brightness in a small area remains the same. That means that the level of gray of an object at a position (x, y) of a frame at the instant (t) is the same at a position $(x + u, y + v)$of the next frame at the instant $(t + 1)$ with a motion following the vector V. This assumption is presented by the equation (1).

$$I(x, y, t) = I(x + u, y + v, t + 1) \tag{1}$$

Where:

- $I(x, y, t)$ is the gray level of the pixel (x, y)at the instant t;
- u and v are the motion along the horizontal and the vertical axes respectively;
- $(x + u, y + v)$ is the coordinates at the instant $t + 1$;

From this hypothesis, the following equation must be satisfied:

$$\frac{dI}{dt}(x, y, t) = 0 \tag{2}$$

$$\leftrightarrow \frac{\delta I}{\delta x}\frac{\delta x}{\delta t} + \frac{\delta I}{\delta y}\frac{\delta y}{\delta t} + \frac{\delta I}{\delta t} = 0 \tag{3}$$

$$\leftrightarrow \quad I.x.u + I.y.v + It = 0 \tag{4}$$

However, this assumption, practically, is very often not respected, for example at the borders objects or on reflective surfaces. Consequently, a second constraint is applied. This constraint assumes that neighboring pixels are likely to belong to the same object. This hypothesis supposes a certain rigidity of the objects locally, however, it is not systematically verified.

Existing algorithms differ in the formulation of these constraints in order to correct these discontinuities and other problems of the OF. Different approaches exist to address these problems such as differential methods, correlation methods and regression methods. Equation (4) presents the constraint to be respected for the motion. However, we have only one equation to determine two unknown u and v. In order to determine these two unknowns, all methods to calculate the optical flow carry out one or more additional assumptions in relation to the nature of the field of motion.

There is different technique of OF computation, the sparse OF applied on some points of interest such as the Lukas kanade OF [Lukas and Kanade, 1981] and the Horn and Schunck OF [Horn and Schunck, 1981]; and the dense OF applied to all points (pixels) in the frame such as the Farnebäck OF [Farnebäck, 2003]

3.1.2.1. Horn and Schunk method

Horn and Schunck [Horn and Schunck, 1981] introduced an overall smoothing constraint, making it possible to estimate the OF throughout the image. They aim to minimize distortions in the OF, they prefer the solutions which present more smoothing. Indeed, the method proposed by Horn and Schunck assumes that the neighboring pixels must have a similar speed of movement, which means that the OF has gradual variation.

3.1.2.2. Lukas Kanade method

Lucas and Kanade [Lukas and Kanade, 1981] has developed a local method for estimating the OF assuming that the latter is constant in a local neighborhood. The OF equation (4) for all the pixels in the neighborhood are considered. The method proposed by Lucas and Kanade is also known for its robustness to noise. It is usually applied on selected points of interest such as corners. However, this approach may present errors in the processing of uniform regions. Figure 7 shows the result of Lukas kanade OF applied on a traffic scene. The computed velocity vectors are presented as blue arrows.

Figure 7: Velocity vectors of some points of interest in a frame computed by the Lukas Kanade OF approach

3.1.2.3. Farnebäck method

An efficient and robust algorithm for the OF computation was designed by Gunnar Farnebäck [Farnebäck, 2003]. This algorithm is a dense computation of the flow of each pixel. The first step of this algorithm is to approximate each neighborhood of each pixel between two frames by a quadratic polynomial.

$$f_1(x) = x^T A x + b^T x + c_1 \tag{5}$$

Where A is a symmetric matrix, b is a vector and c is a scalar.

We consider d as an ideal translation of the pixel. A new signal $f_2(x)$ is then constructed as follow

$$\begin{aligned} f_2(x) = f_1(x-d) &= (x-d)^T A_1 (x-d) + b_1^T (x-d) + c_1 \\ &= x^T A_2 x + b_2^T x + c_2 \end{aligned} \tag{6}$$

We obtain the flowing equations:

$$A_2 = A_1 \tag{7}$$

$$b_2 = b_1 - 2A_1 d \tag{8}$$

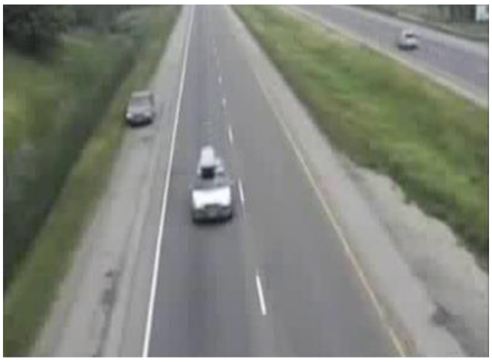
$$c_2 = d^T A_1 d - b_1^T d + c_1 \tag{9}$$

If the matrix A_1 is non-singular, then from the equation (8) we get the translation d as follow:

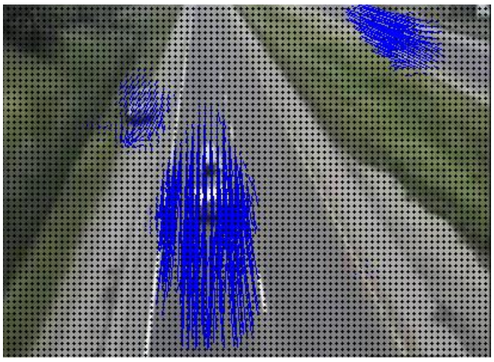
$$d = \frac{-1}{2} A_1^{-1} (b_2 - b_1) \tag{10}$$

From the value of d, the velocity of a pixel could be computed.

Using the dense OF allows having more information about the motion at a global scale compared to the sparse OF which is related to some point of interest. Figure 8 shows the velocity vectors computed by the Farneback OF in a traffic scene.



(a)



(b)

Figure 8: The results of Farnebäck OF applied on a traffic scene. (a): the original frame, (b): the drawing in blue of the velocity vectors computed with the Farnebäck OF.

3.1.3. Scale Invariant Feature Transform

The SIFT is an algorithm widely used in the field of computer vision. Developed by David Lowe in 1999 [Lowe, 1999], it can detect and identify similar objects between different images. So, it allows characterizing the visual content of the image independently of the scale, the brightness and the position of the camera. SIFT is based on 3 steps: first gradient computation for pixels; second the Gaussian computation applied to the values of gradients and finally the histogram computation. Figure 9 shows an example of the use of SIFT descriptor for object recognition as presented in [Morel, 2011]. The descriptor SIFT is used for motion tracking using matching between objects. However after few frames, and due to the video distortion, the number of matches decrease so the number of false alarm increase. Researches, used to combine SIFT descriptor with other tracking technique like kalman in [Mantripragada et al, 2014] and OF in [Chen et al. 2016].

Figure 9: An example of matching between two images using SIFT [Morel, 2011].

3.1.4. Vehicle recognition

After motion detection, the most challenging task is vehicle recognition. Moving targets can be represented by points or geometric areas like rectangles. However to distinguish vehicles from other objects, there is two categories of approaches: The first ones are the representative approaches based on the colors, the edges, the shape of vehicles or their parts like windshield and lights. The second one are more complex based on training techniques like Haar [Viola and Jones, 2001] or descriptors like the Speeded Up Robust Features (SURF) [Bay et al., 2006] used in [Li and Zhang, 2013] for vehicle detection and the Histogram of Oriented Histogram (HOG) [Dalal and Triggs, 2005]. The Haar-like cascade classifier is a classifier trained with hundreds of samples of the object to detect the called positive samples from other arbitrary objects called negatives samples. After training, a classifier can be applied to a zone of interest to perform the detection. However, in case of HOG, the image is divided into cells which are grouped into overlapping blocks. The gradient is computed for each pixel and the mean orientation of the gradient is associated with each cell. The vector of features is then formed, for each block, to form the histograms of the oriented gradients. This vector, normalized later, characterize the shape of the object. Flowed by a classification technique, the nature of object could be recognized. The most used classifier are the Support Vector Machine (SVM) and AdaBoost [Wu and Weng, 2017] [Chen et al, 2017]. Figure 10 shows the bloc diagram of HOG followed by the SVM classifier.

Some researchers improved the HOG algorithm to enhance the detection rate such as [Kim et al, 2015] who invented the PIHOG which add to the histogram the position information.

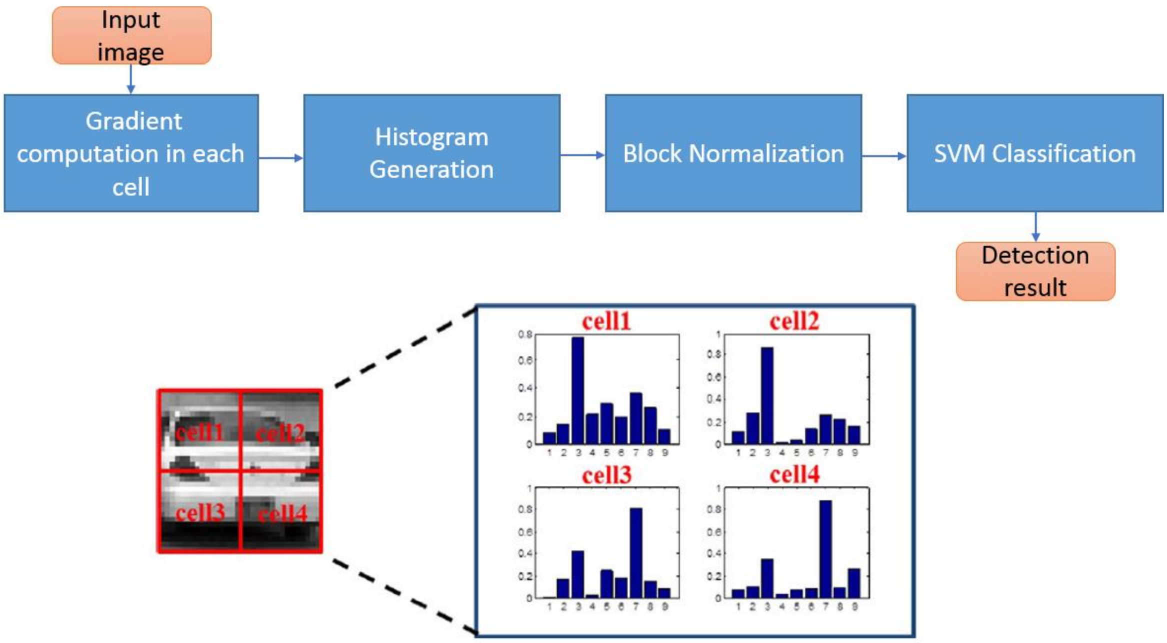


Figure 10: The block diagram of HOG followed by the SVM.

3.1.5. Vehicle velocity and orientation computation

The tracking methods consist of estimating the motion of a specific points or a regions of the frame. For an object at the position (x_1, y_1)in a frame (i), tracking techniques estimate its next position (x_2, y_2) in the frame $_{(i+1)}$. Using this information, the vehicle velocity vector $\vec{u}$ and the angle of orientation θ could be computed as presented in Figure 11. The acceleration and the vehicle trajectory could be determined. Tracking techniques are important for the applications related to the behavior analysis. The most commonly used techniques are Kalman filtering [kalman, 1990], Bayesian filtering [Haritha and Ramadevi, 2013] and OF.

.

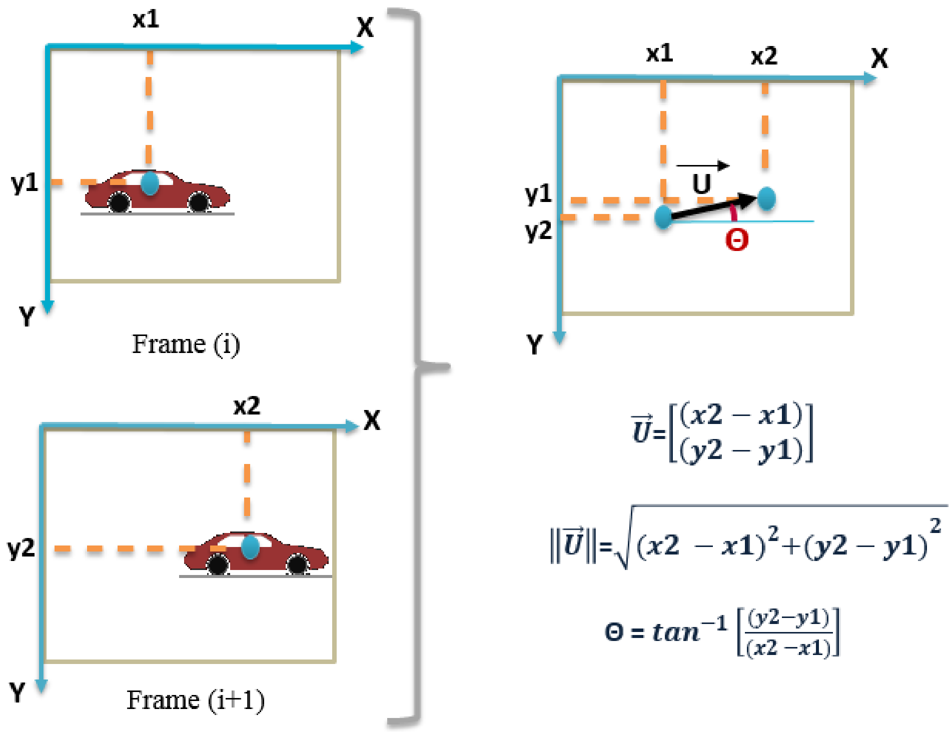


Figure 11: An example of vector velocity computation

3.2. Features extraction

A feature is a characteristic of a given object in motion. It could be velocity, orientation, trajectories, area, position and histograms, etc. Usually combining more than one characteristic gives more details about the motion. Some researchers are interested on extracting features of the vehicles in motion, other researchers preferred extracting features of all particles (pixels) to have dense motion information at a global scale which we assume more accurate.

A comparison between the extracted features and predefined conditions allow the detection of abnormal behavior. Thus, the accuracy of each algorithm of accident detection depends on the quality of the extracted features. To have sufficient results, features must be relevant covering the maximum of the aspect of the motion.

3.3. Accident recognition

In literature, researchers used different approaches to recognize an accident. We classify them in two categories. The first one is based on the comparison of the extracted features to predefined fixed thresholds. The second one is based on the comparison of the extracted features to learned models. These models could be the normal trajectory of a vehicle or a specific representation of its motion such as the histogram of velocities. The models of a normal traffic are obtained from a large dataset of videos of normal traffic situation.

3.3.1. The comparison to predefined fixed thresholds

Ki and Lee [Ki and Lee, 2007] used, for moving object extraction, difference between two frames, binarization, horizontal and vertical projection then, the extraction of pixels that exceed a threshold. In their tracking approach they estimate the area of vehicle in next frames according to the direction of motion. The area of the vehicle is expected to have the same size in the second frame. In the third frame the estimated area can be reduced. To make the process faster, vehicles are estimated in the third frame by using direction and velocity. Next, the algorithm extract features as acceleration, position area and direction of the vehicle and compute the variation rate of vehicle speed (Velocity Feature: VF), the variation rate of position (Position Features: PF), the variation rate of area (Area Feature: SF) and the variation rate of direction (Direction Feature: DF). An accident is detected if these values exceed a threshold. A diagram of the algorithm is presented in Figure 12. [Hui et al. 2014] used the GMM for motion detection then the Mean Shift algorithm for tracking and features extraction (velocity, acceleration and orientation). The accident recognition is also performed by comparison to static predifined thresholds.

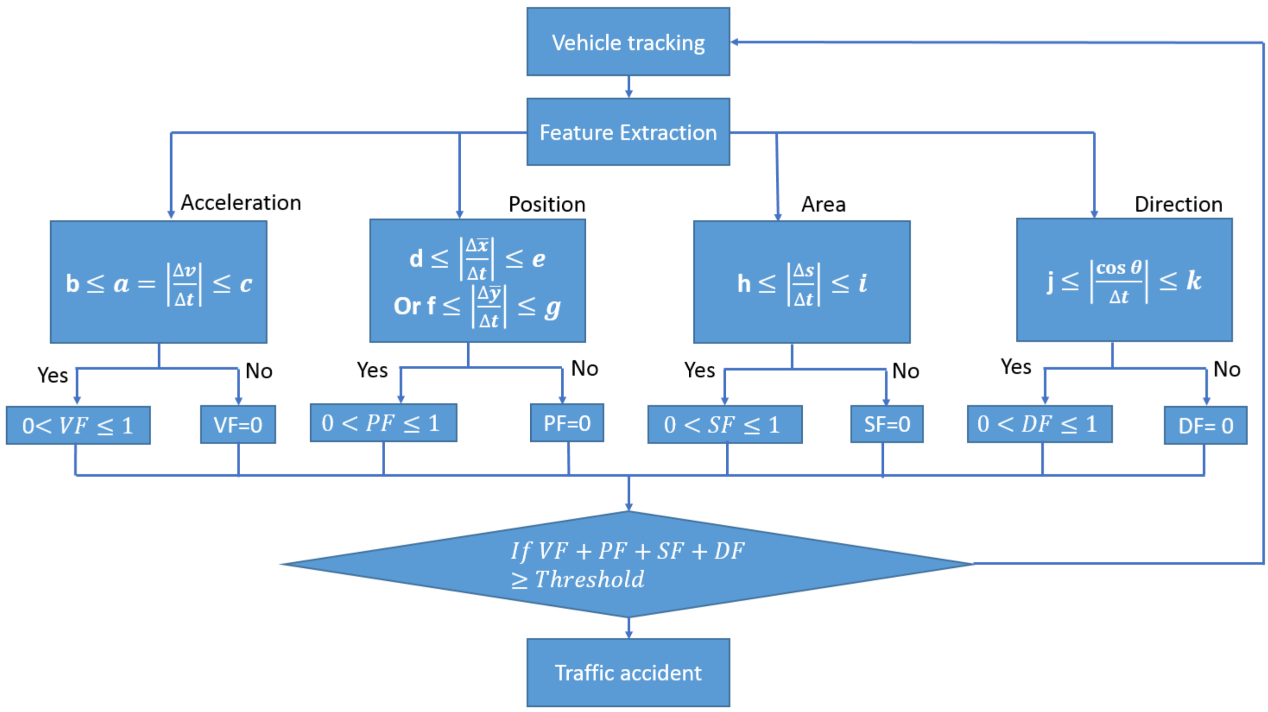


Figure 12: Accident detection algorithm flowchart. *V*, *(x , y)*, *s*, θ are respectively the velocity, the coordinate of the centroid of the vehicle, its surface and the angle formed between two motions vectors. *a*, *b*, *c*, *d*, *e*, *f*, *g*, *h*, *i*, *j*, *k* are thresholds.

The limitation of the approaches based on static thresholds is that we have to manually fix new thresholds values for each traffic scene. A lot of information are needed for that such as the environment of detection, the camera position and calibration and the resolution of the image.

3.3.2. The use of learned models

To avoid the use of fixe threshold, researchers used trajectories analysis. For example in [Lee, 2012], the authors perform road lane analysis based on Calogero-Moser system. Accident is detected when an abruptly change in line is detected. In [Akooz and Karsligil, 2010], detection is performed by extracting moving blobs and removing background noise using smoothing techniques. Next, the position, the velocity, the acceleration and the vehicle trajectory are extracted. Thereafter the Continuous Hidden Markov Model (C-HMM) was used to cluster trajectories and find activities path. A model of normal vehicle behavior is learned so an accident is detected when the system detects an activity that does not match the learned model.
The limitation of these approaches is the impossibility of distinguishing an accident from another abnormal behavior. Any abrupt change is reported, which could be a vehicle turning in a wrong direction for example.

The different works sited above are based on vehicle motion analysis which give less information compared to the detection based on particle (pixels) motion analysis. For that reason, some researchers used OF to track pixels and extract more features to detect accident. For instance, [Sadek et al., 2010] proposed a new framework for accident recognition relatively tolerant to changes of lights based on a new algorithm named HFG (Histogram of Flow gradient) which is similar to HOG running on OF algorithm for motion estimation. Figure 13 shows the different steps of HFG algorithm. After video acquisition, the first step is the sparse OF computation. Then, the angle and the magnitude of the flow velocity are represented by an 8 bins histogram of gradient orientation. Each bin represents a range of orientation in the trigonometric circle as presented in Figure 13. The last step is an automatic classification for accident detection. [Ahmadi et al., 2016] exploit the Topic Model technique creating a model of normal traffic motion based on Lukas Kanade OF vectors indexed in a document of words. Each word represents velocities in a specific range of orientation. Abnormal traffic is detected when the computed words are different from the model

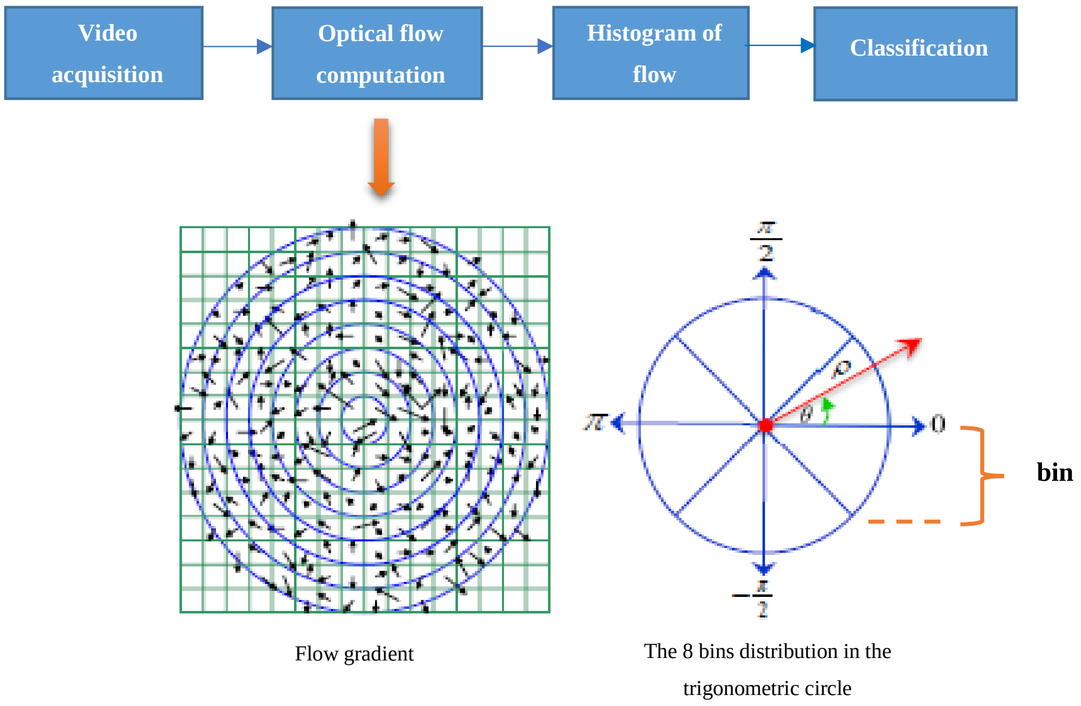


Figure 13: The block diagram of HFG algorithm. ρ and θ are the magnitude and the velocity angle respectively.

In these works, the OF is applied on points of interest, so many information in the image was not considered. Besides, OF creates distortion vectors, if not filtered, it can affect the accuracy of the results. As a solution, researchers used dense OF and filtering techniques. In [Ullah et al., 2015], authors used Farnebäck OF for motion detection then applied Thermal Diffusion to find the coherent motion field. They used the Enthalpy Model to filter particles and model the traffic motion. Then they used Smoothed Particle Hydrodynamics (SPH) for accident recognition. In another work, [Ren et al., 2016] proposed to extract moving objects using background subtraction then extract road segments and lines positions. Then a traffic tracking and time and space diagram was generated. SVM was used for classification and accident recognition. Chen et al [Chen et al., 2016] extracted OF and SIFT features. Then they used Bag of Feature (BOF) for features encoding and finally they used Extreme

Learning Machine classifier (ELM) for accident detection. Figure 14 shows the block diagram of their algorithm.

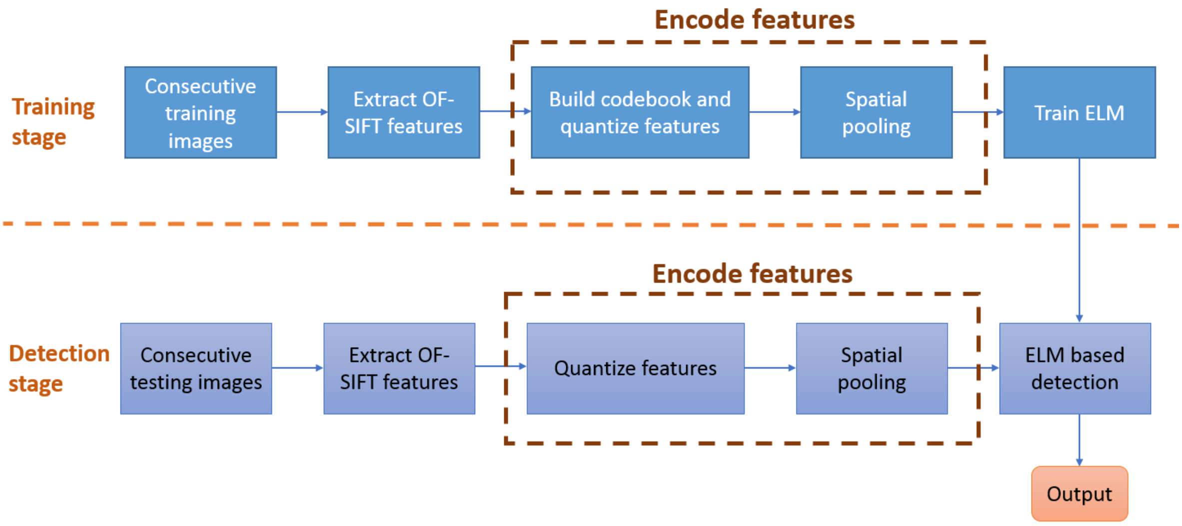


Figure 14: The block diagram of accident detection algorithm based on ELM.

The limitation of these approaches is the need to collect a large amount of datasets for more accurate results. However, challenging database of accident videos does not exist so far, since accident simulation is dangerous, and registering videos of real traffic is not allowed by authorities. Created synthetic scenarios or 3D simulation of accident does not take into consideration luminosity or meteoric change and noises.

Researchers used different videos found on websites, created synthetically or given by the authorities in their country [Ki and Lee, 2007].Table 2 shows a comparison study of existent approaches depending on different criteria: the used algorithms, the type of road, the accuracy, the time for detection, the used datasets, and the platform of implementation. For instance, a comparison in term of accuracy and robustness is not possible for many reasons. First, researchers did not use the same datasets for experimental results. In addition, they did not use the same metrics for accuracy computation, for example [Sadek et al., 2010] computed the detection rate and false alarm to evaluate their approach, while [Ullah, 2015] computed the *F1_score*. Other information are missing in some works such as the type of road and the time of detection.

4. Conclusion

This chapter resumes the common techniques used in computer vision for vehicles recognition and tracking. In addition, an overview of the state of the art of different approach of abnormal behavior detection and accident detection in road was presented. We highlighted the performance of each technique and its limitations. OF based techniques are promising thanks to their robustness. However, the modeling of a traffic flow needs a large dataset of videos which does not exist. In the next chapter we will present new algorithms of accident detection.

Ref	Motion detection and tracking	Features extracted	Accident recognition	Region of detection	Dataset	Results	Time of detection	Simulation/Implementation
[Ki and Lee, 2007]	BS+ Vehicle area tracking	Velocity, acceleration, orientation, area	Static threshold	Intersections Free ways	Videos from cameras installed in Seoul	4 accident detected during the period of test	Real time	?
[Hui, 2014],	GMM + Mean Shift Algorithm	Velocity, acceleration, orientation	Static threshold	Highway	One video,320×240 pixels, total frame is 347	Accident detected on one video.	Real time	Windows 7, memory of 2.00G, Visual studio 2010
[Akooz and Karsligil, 2010]	BS, KLT Tracker	Vehicle trajectory clustered	C-HMM	Intersections	106 synthetic accident scenes creation	84% detection rate according to accident severity	?	?
[Lee, 2012]	BS, shadow filtering, vehicle tracking	Vehicle lane (trajectory)	Calogero-Moser system	Highway	Stored moving images	100% of detection for 10 cases of study	1-1.5 minutes	?
[Sadek et al., 2010]	Lukas kanade Optical flow	HFG	Logistic regression	Multiple scenarios	45 Websites videos with 250 accident scene	Detection rate : 99.6% False alarme: 5.2%	Real time	Intel(R) Core(TM)2 Q9550 2.83GHz 4GB of RAM Visual studio 2008, openCV
[Ahmadi et al., 2016]	Lukas kanade Optical Flow	Histogram of words	Topic Model	Multiple scenarios	Tehran traffic videos	?	?	?

[Ullah, 2015]	Farneback OF, thermal diffusion, enthalpy Model	Traffic model	SPH	Multiple scenarios	20 video sequence of accidents from websites	F1score = 0.73	?	?
[Ren et al., 2016]	Background subtraction, extracting line positions, tacking	Time and space model diagram	SVM	Multiple scenarios	12 lane-hours traffic which includes 6 lane-hours video collected in YanTai 6 lane-hours video collected in Beijing. 640 × 480 resolution at 25 fps	Detection rate= 96.87%	?	VC++ 2008 in Windows XP and ran on a computer with Intel® Core(TM) i3-3220 CPU at 3.30 GHz, 4 GB RAM and AMD Radeon HD6700 display adapter
[Chen et al,. 2016]	Optical flow and SIFT	BOF for feature coding	Extreme learning machine	Highway	Videos from websites	Precision : 83.3%-100% Recall: 100%-50% Accuracy : 82.6%-96.7%	?	?

Table 2: A comparison study of the different video based approaches for accident detection

Chapter 2: Constructive approaches for video-based road accident detection

Contents

In this chapter we present our algorithms for video-based accident detection on road segments. The first one is based on vehicles tracking using the Kalman filter. Extracted features are vehicles velocities and orientations. A static threshold was computed for accident recognition. The other algorithms are based on particle tracking using the Farnebäck OF. Extracted features are histograms of pixel velocities and orientations for each frame. Different thresholds are defined. An improvement of this approach is presented as a last algorithm based on traffic modeling and dynamic threshold computation. The conclusion of the accuracy of each approach for accident detection is presented in chapter 3 based on experimental results.

1. Introduction

As presented in the previous chapter, there are two categories of approaches for accident detection. The first one is based on vehicle motion analysis and the second one is based on relevant pixel analysis. With the lack of a common dataset, the comparison between these different approaches in terms of accuracy is still a complicated task. With the purpose of validation of the hypothesis proposed in the previous chapter (the detection based on particle motion analysis is more accurate than detection based on vehicle motion analysis), different algorithms for accident detection on a segment of road were developed. Algorithms are in accordance with the state of the art categories. The first approach is based on vehicle tracking using the kalman filter and then a comparison to a static threshold for accident recognition. In the second approach we are focused on the tracking of all pixels in the image using Farnebäck OF. For this approach, first, different methods for features definition were used, and then the static threshold for accident detection was fixed. Finally, we propose a novel approach based on road traffic modelling and dynamic threshold computation. Experimental results will be discussed in the next chapter.

2. Accident detection based on vehicle motion

Usually an accident is characterized by a sudden change in the velocity (increasing speed or sudden stopping) and the orientation of one vehicle or more. To have this information we first need to detect the presence of vehicles, and then track them to save the evolution of their motion in a video sequence. Figure 15 shows the flow chart of the algorithm. First, after testing different BS algorithm, the GMM was chosen for motion detection since it has better accuracy. Vehicles are recognized by the size of their areas. Then, the centroid of each vehicle is computed and tracked between two different frames using Kalman filtering.

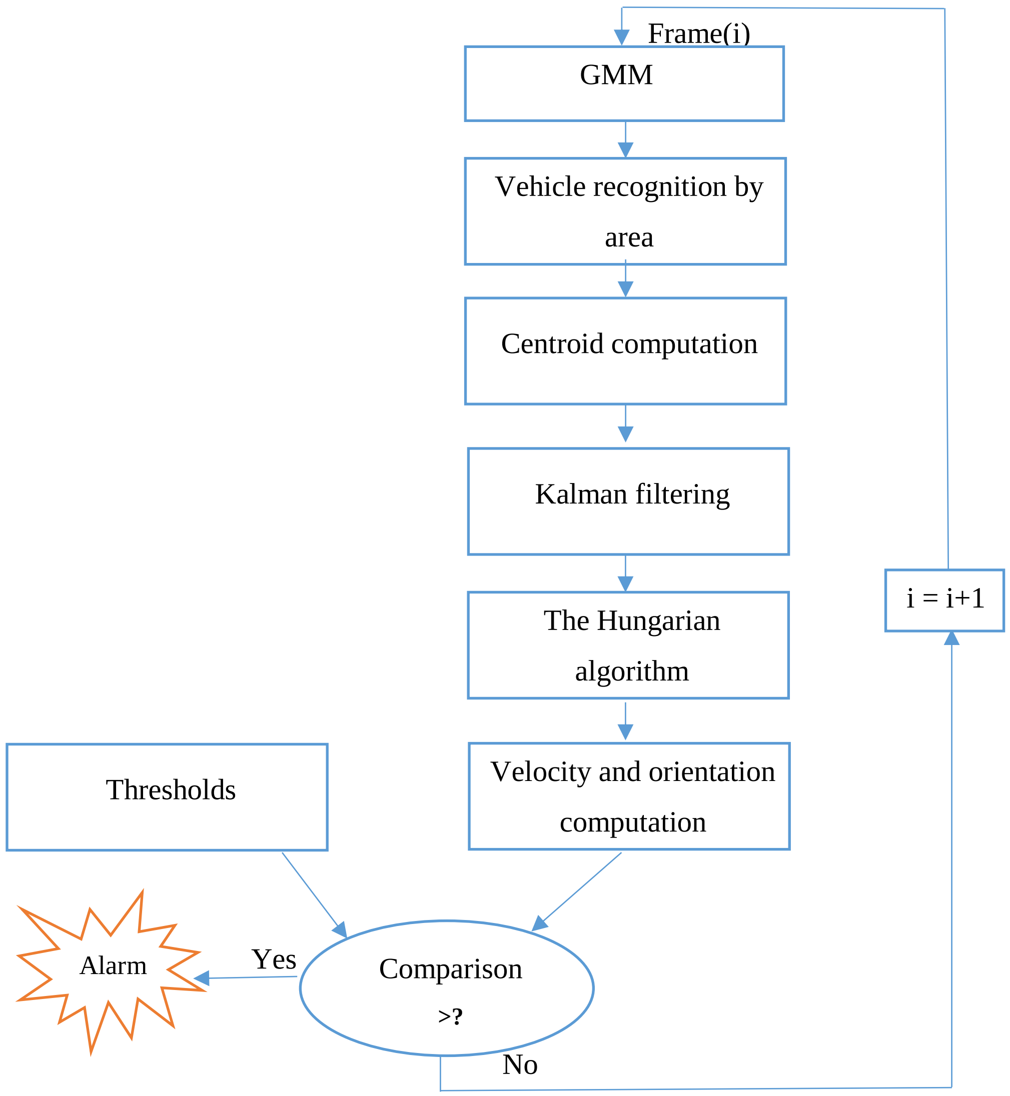


Figure 15: The flow chart of the algorithm of accident detection based on vehicle tracking

A common problem with tracking is to identify the same vehicle from a frame to another. The problem of vehicles assignment is resolved using the Hungarian algorithm. Then velocity and orientation are computed (as explained in the previous chapter) and compared to predefined thresholds. If the velocity and the orientation values exceed the value of the thresholds, an accident is detected.

2.1. Vehicle recognition

After applying the GMM algorithm, a binary image representing the foreground image is extracted. From this image, the contours delimiting the white blobs that represent the moving objects are extracted. Each contour is composed of a set of points. These points are known by their coordinates (pixels coordinates). From these values, areas of moving object are computed. Depending on these areas, we conclude the nature of the object and where it is a vehicle or not. In other words, the area of each moving object is computed and compared to a threshold ‘min’ and ‘max’. If the area is bigger than the value of ‘min’ and smaller than the value of ‘max’ we consider the moving object as a vehicle. However if the area is smaller than the value of ‘min’, the detected object is not considered. In the case that the area is bigger than the value of ‘max’ a problem is noted. This problem may be a big shadow, rain, wind or congestion (vehicles are too close to each other). The values of ‘min’ and ‘max’ depends on the camera position and its calibration. Consequently, these values are experimentally defined. Thereafter, the centroid of each vehicle is computed to be tracked later.

The C++ code below Code 1 shows how to detect and compute the centroid of each vehicle. The function *findContours* was used to extract the vector of coordinates of each point in the contour of each moving object. *ContourArea* is used to compute the area of each moving object. Then centroids are computed using the function *moment.*

```
cv::findContours(image, contours_all,CV_RETR_EXTERNAL,CV_CHAIN_APPROX_NONE);
 for( int i = 0; i < int (contours_all.size()); i++ )
   {
    if (cv::contourArea(contours_all[i])> 200)

          vehicles.push_back(contours_all[i]);
   }
 vector<Moments> mu(vehicles.size() );
 vector<Point2f> mc(vehicles.size() );
 vector<Point2f> mco(vehicles.size());

 if (vehicles.size() >= 1)
   {
    for( int i = 0; i < int (vehicles.size()); i++ )
          {
            mu[i] = moments( contours[i], false );
            mc[i] = Point2f(double (mu[i].m10/mu[i].m00) , double
                    (mu[i].m01/mu[i].m00) );
             points_for_tracking.push_back(mc[i]);
          }
   }
```

Code 1 : The C++ code for vehicle detection and its centroid computation.

2.2. The Kalman filter

The Kalman filter [Kalman, 1990] is an optimal estimator of the state of Gaussian systems. It estimates recursively the state of a system whose state transition are known, as well as the uncertainty associated with it. It receives input in the form of a series of observed measurements as well as the noises of the state evolution. The strength of the Kalman filter is that, beside the computation of a vector of estimated metric of a given system, it computes a covariance matrix of estimated errors. The Kalman filter is the most commonly used algorithm in many fields related to signal processing, radar and image processing especially for objects tracking. Thus, it estimates the next position of an object in motion from a series of observations of its previous positions.

2.3. The Hungarian algorithm

The Hungarian algorithm is a combinatorial optimization algorithm that solves the polynomial time assignment problem. It is therefore an algorithm that allows finding a perfect coupling of maximum weight in a graph whose edges are valued. Alternatively, it is possible to find a perfect minimum weight coupling in such a graph. The input of the algorithm is an *n* by *n* square matrix with nonnegative elements. The execution of the algorithm is performed in four steps: 1) subtracting row minima, 2) subtracting column minima, 3) covering all zeros with a minimum number of lines and 4) creating additional zeros[10]. If *n* lines are required in step 3, an optimal assignment does exist among the zeros and the algorithm must stop. Else, no optimal assignment does exist and step 4 should be executed. The Hungarian algorithm is used to assign every vehicle in motion to its corresponding estimated position computed by the Kalman filter.

2.4. Accident recognition

After tracking, the velocity and orientation of each vehicle are computed. The history of position, orientation and velocity of each vehicle is saved for X frames. X is fixed with experimentations on videos. Figure 16 presents a screen shot of the results of tracking using the Kalman filter. Each colored line draws a line linking the X saved positions (coordinates) of the same vehicle (X here is equal to 10 frames). These colored lines present the trajectories of tracked vehicles.

[10] http://www.hungarianalgorithm.com/hungarianalgorithm.php

The saved histories allows for the detection of two abnormal scenarios described by the equations below:

$$\text{Stopped vehicle : if } \sum_{t}^{t+th_1} V_{(i,t)} \approx 0 \tag{11}$$

$$\text{Accident : } (\theta_{(i)} > th_2)\ \&\&\ (V_{(i,t)} \approx 0)\ \&\&\ (V_{(i,t-1)} \approx 0) \tag{12}$$

Where:

$V_{(i,t)}$ is the velocity of vehicle (i) at time t (which represents a velocity of a vehicle (i) at a frame(t)).

th_1 is the number of frames needed to be sure that the vehicle has stopped.

th_2 represents the maximum allowed orientation of a vehicle.

th_1 and th_2 are experimentally fixed. The unit used for velocity computation is pixels/frame because of the lack of information about camera calibration. For the orientation, the unit used is degrees.

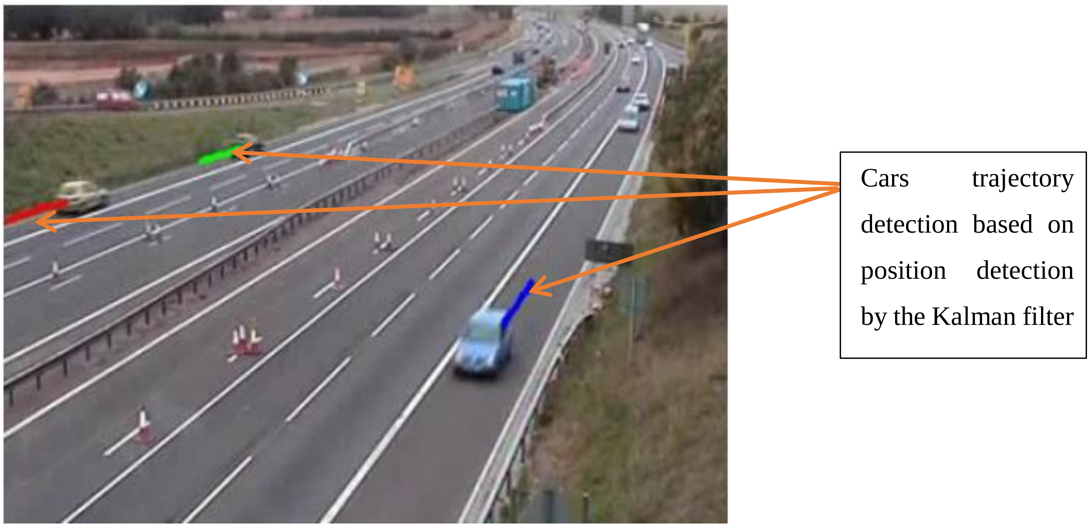


Figure 16: A screen shot of tracking results using the Kalman filter. Colored lines are vehicle trajectories.

A stopped vehicle in the middle of the road or at the emergency line (the hard shoulder) of highways is considered as an emergency event that should be notified as an accident. Using tracking or just applying a BS technique on predefined zones (such as the emergency line), allows this kind of abnormal behavior detection.

3. Accident detection based on particles motion

3.1. First approach: OF, histogram computation and fixed thresholds

The basic idea of this approach is to detect accidents at a global scale by computing the velocity and orientation of each particle (pixel) in a frame. We use for that the *Farnebäck OF*, which detects the motion field in a ROI. The ROI, which is a window of *n*n* pixels, can be restricted to the road. OF allows estimating for each pixel the velocity vector per frame. These vectors will contribute by their magnitude on a histogram of 8-bin orientations in the range of [0°, 360°]. However the number of obtained velocity vectors in the ROI is big and some of them are noisy or distorted. As a solution, we propose three different methods to reduce the number of velocity vector while keeping enough information about the motion and reducing noise. These methods are described below:

1) *Method 1*: Dividing the ROI into *k***k* blocks and computing the average of the velocity vectors of each block

2) *Method 2*: Dividing the ROI into *k***k* blocks and computing the median of the velocity vectors of each block

3) *Method 3*: Computing the average of the magnitude of all velocity vectors in the ROI that have the same range of orientation.

For each method, each computed new vector contributes to the histogram by its magnitude to the bin that represents its range of orientation.

The purpose of developing these different methods is to compare them to find the most suitable features. Obtained histogram is compared to a predefined threshold. The method that allows us to have a histogram that is more sensitive to sudden changes in vehicle motion is the more accurate. Figure 17 shows the flowchart of the algorithm.

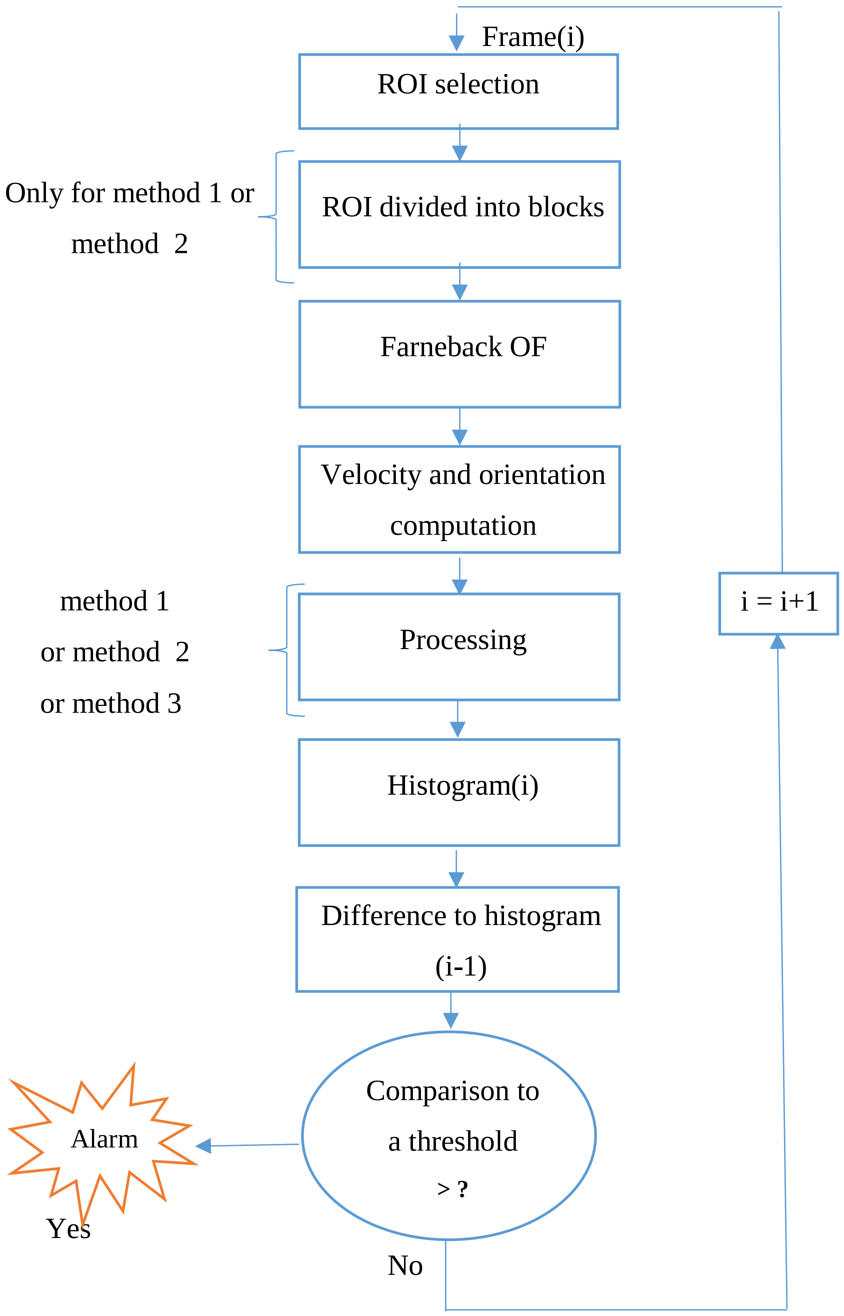


Figure 17: A flowchart of an accident detection algorithm based on OF and static threshold

3.1.1. Method 1

This approach is based on dividing the ROI into blocks of k*k pixels. One vector will represent the pixels motion in each block by computing the average of pixels velocity vectors. The average vector is given by equation (13) and the value of each bin in the histogram is computed as shown in equation (14). k is experimentally defined in chapter 3. Figure 18 shows a schematic explanation of the histogram computation based on this method.

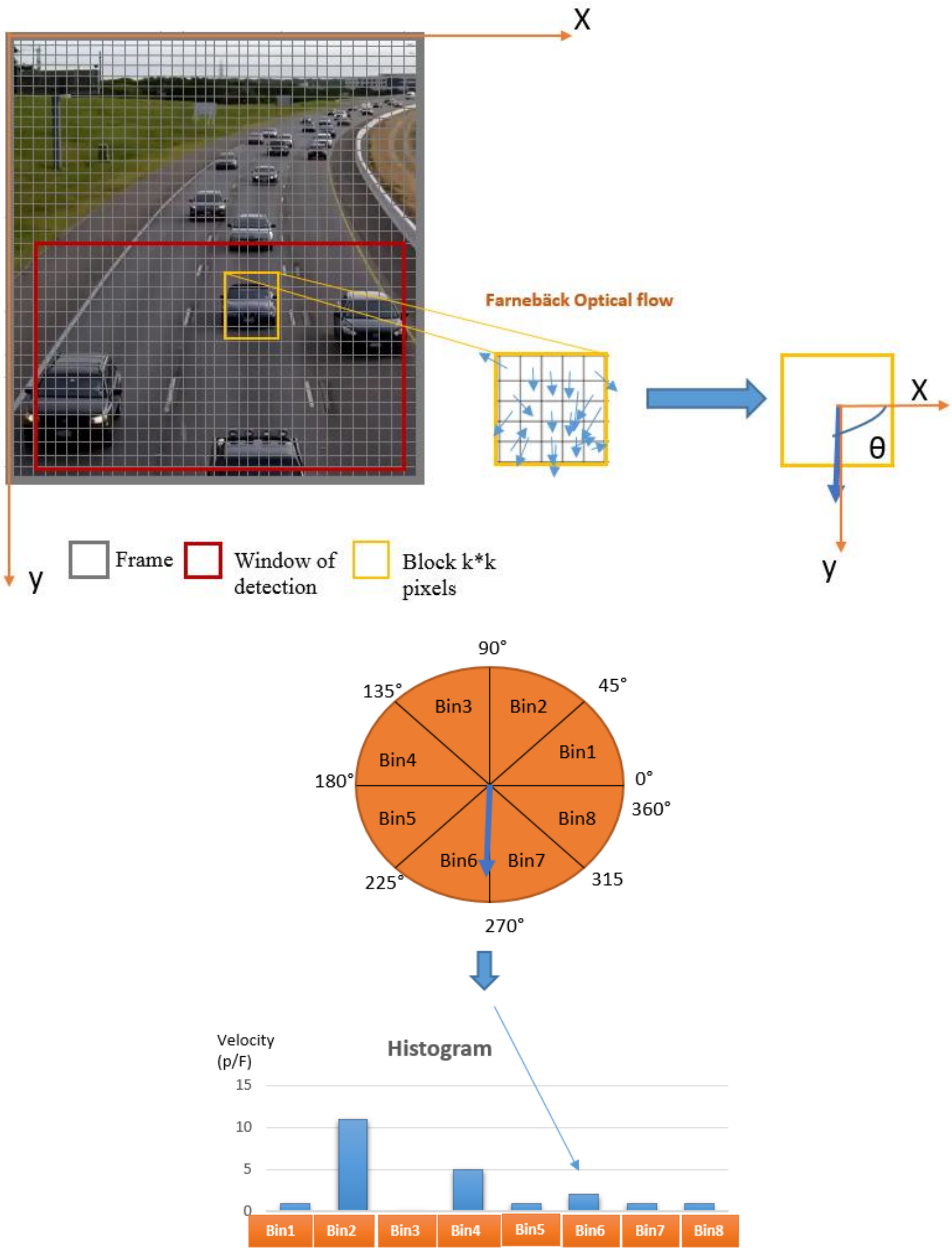


Figure 18: A schematic explanation of the histogram computation

$$\overrightarrow{AverageVector} = \begin{vmatrix} (\sum_{1}^{k} x(i))/\, k^2 \\ (\sum_{1}^{k} y(i))/k^2 \end{vmatrix} \tag{13}$$

$$\text{Velocity}_{(binx)} = \sum_{1}^{m} \left\| \overrightarrow{AverageVector(\iota, \theta)} \right\| \tag{14}$$

Where:

m is the total number of the Average vector within the orientation (θ) in the interval [bin_x]

bin_x is the interval of orientation as presented in the trigonometric circle in figure 17.

x € [0, 8]. 8 is the total number of bins.

3.1.2. Method 2

As with the method 1, this method is also based on dividing the ROI into blocs of $k*k$ pixels. Then one vector will represent the motion of each block by computing the median of pixels velocity vectors. An example of how this vector is computed for a block of 3*3 pixels is shown in Figure 19.

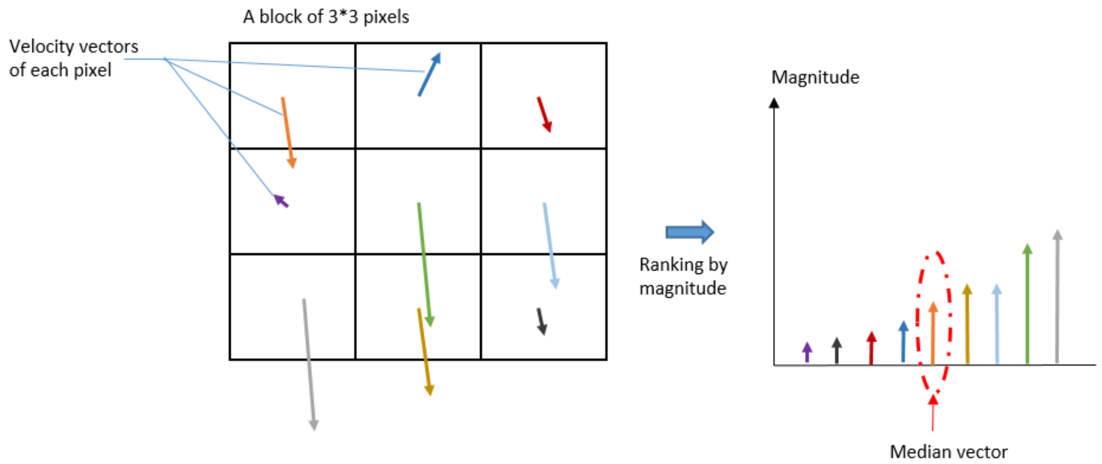


Figure 19: An example of median computing for a block of 3*3 pixels

3.1.3. Method 3

For this method, the average of the magnitude of all velocity vectors of the pixels in the ROI having the same range of orientation is computed and added to the corresponding bin in the histogram. Figure 20 shows a schematic presentation of different vectors with orientations in different ranges of the trigonometric circle before and after average computation. The final values are then indexed on the histogram.

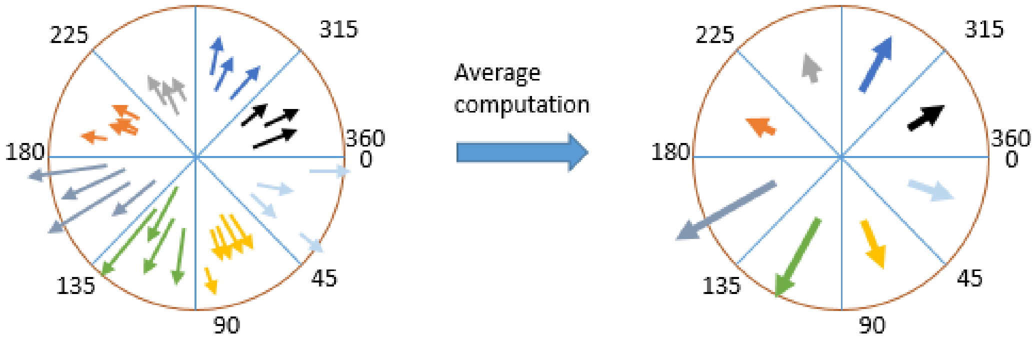


Figure 20: A schematic representation of average vector computation

3.1.4. Accident recognition

An accident is detected when a dramatic change of bins on the histogram between two successive frames occurs. Figure 21 shows an example of the changes on the histogram before and after an accident. This dramatic change is expressed by exceeding a predefined threshold. To fix the threshold we first compute the histogram of each frame in a normal traffic scene. Then, we compute the difference between every two successive histograms *H(t)* and *H(t+1)* of every two successive frames *F(t)* and *F(t+1)* by the equation (15):

$$H(t+1) - H(t) = \sum_{1}^{x} abs\big(b(x,t+1)\big) - abs\;(b(x,t)) \tag{15}$$

Where :

bin(x) is a bin from the histogram, *x* € [0,8]

F(t) is a frame at instant *t*, *t* € [0, *l*]

l is the total number of frames

We consider a maximum change between two histogram for *l* frames in a normal traffic scenes a threshold *TH* given by the equation (16):

$$TH = \; max_1^l \; (H(t+1) - H(t)) \tag{16}$$

Figure 21: Example of histogram variation before and after a traffic accident.

This threshold represents the value of the maximum change between successive frames for normal traffic scene. *TH* depends on many factors like the traffic flow (congestion or not), changes in weather condition, the position of the camera and its calibration and the video quality and resolution. The value of *TH* is adjusted by experimental results. The experimentation was performed on scenarios taken from websites, as samples of different quality, orientation, and environmental conditions. It helps establishing parameters needed to compute histogram and to show in which scenario our algorithm succeed. We will explain in details these steps in Chapter 3.

3.2. Second approach: Traffic modeling and dynamic threshold computation

At first, we consider a traffic motion scene in a highway as normal motion flow when the velocity vector of each moving vehicle has a constant or gradual magnitude. Without exceeding the traffic's low limitation, moving vehicles have an orientation suitable to the shape of the road and the traffic direction. Consequently, an accident including one or more vehicles can be detected when a sudden and sharp change in their velocity vectors occurs. This may concerns either their magnitude and/or orientation. It is important to note that velocity and orientation parameters depend on the position and the calibration of the camera. Hence, detecting an accident first needs the extraction of normal motion flow parameters from different traffic scenes captured by the same camera to create a normal traffic model. This model is composed of a number of thresholds that we update with every new frame when the traffic motion parameters do not exceed the initial thresholds. The initial thresholds are calculated based on the first N frames of the video. The N frames value is fixed with experimentations.

The *normal traffic model* is created by analyzing the traffic motion field. This is done using the Farnebäck optical flow applied to the ROI. The ROI is limited to the highway road segment. Following on, we first filter the noise and then we create the model before finally comparing it to *the normal traffic model* to detect an accident.

3.2.1. Velocities classification and noise filtering

After observation, we distinguish two categories of velocity vectors: vectors generated by video distortion (dust, wind, etc.) and vectors related to the vehicles motion. Since OF is sensitive to changes in luminosity and noise, distorted vectors can appear. These vectors typically have small magnitudes so that a threshold ε has been set for filtering. Every vector with a magnitude smaller than ε is filtered. The threshold ε is calculated by performing the OF on a fixed number of frames in which no traffic is detected. We fixed the number of frames to calculate ε to 10. Remaining velocity vectors belong to the following groups: vectors with orientation corresponding to *the normal motion flow* and distracting vectors with other orientations. These distracting vectors can be caused by the motion of the wheels, the presence on the road of shadows, variation in the light reflected on the car's body, etc. In Figure 22, an example of vectors classification is presented.

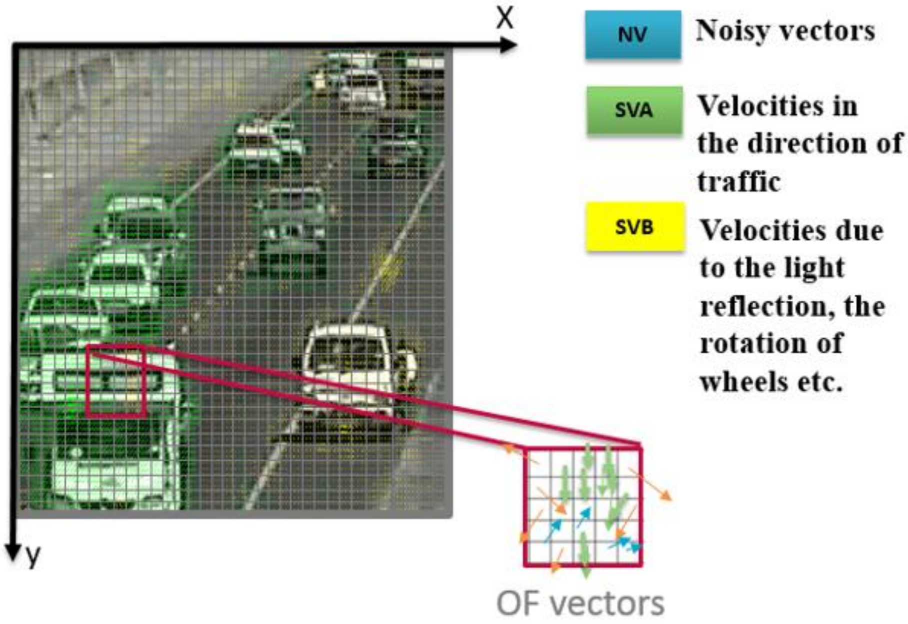


Figure 22: Different velocity vectors classification

3.2.2. Normal motion flow modeling

The different steps of *normal motion flow* modeling are presented in Figure 23. They are identified for a one-way road with a normal traffic behavior. After motion detection by applying Farnebäck OF and then noise filtering, feature are extracted: For each frame (t) we compute the sum of the vectors *SVA* with an orientation range corresponding to the direction of traffic [θ_1, θ_2] (which are presented in green color in Figure 22 and Figure 23) and the sum of the remaining distracting vectors (presented in yellow color in Figure 22 and Figure 23) *SVB*.

SVA(t) and *SVB*(t) present a *sub-model* of traffic motion of the frame(t) .

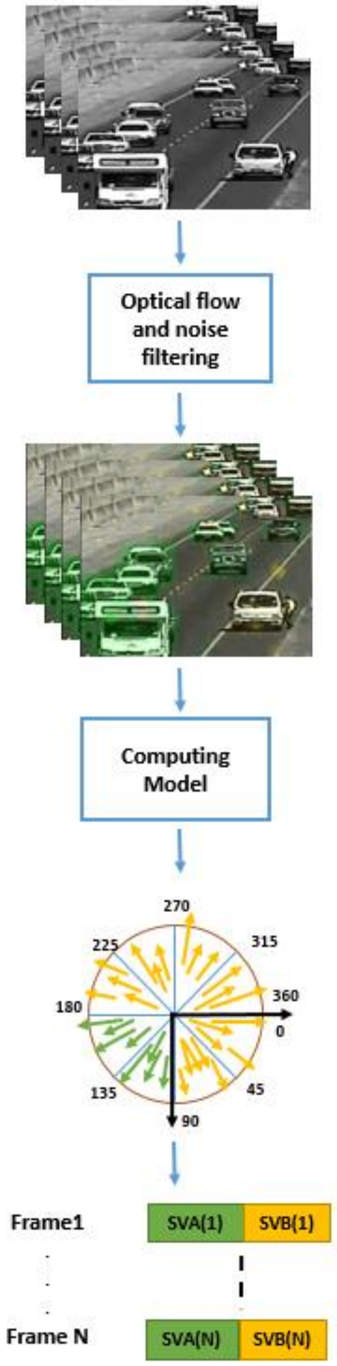


Figure 23: The different steps for normal traffic modeling

The range of orientation [θ_1, θ_2] corresponding to the *normal motion flow* is already known as the position of the camera is known. Figure 24 shows the curves of *SVA* and *SVB* variation as a function of frames for a normal traffic scene.

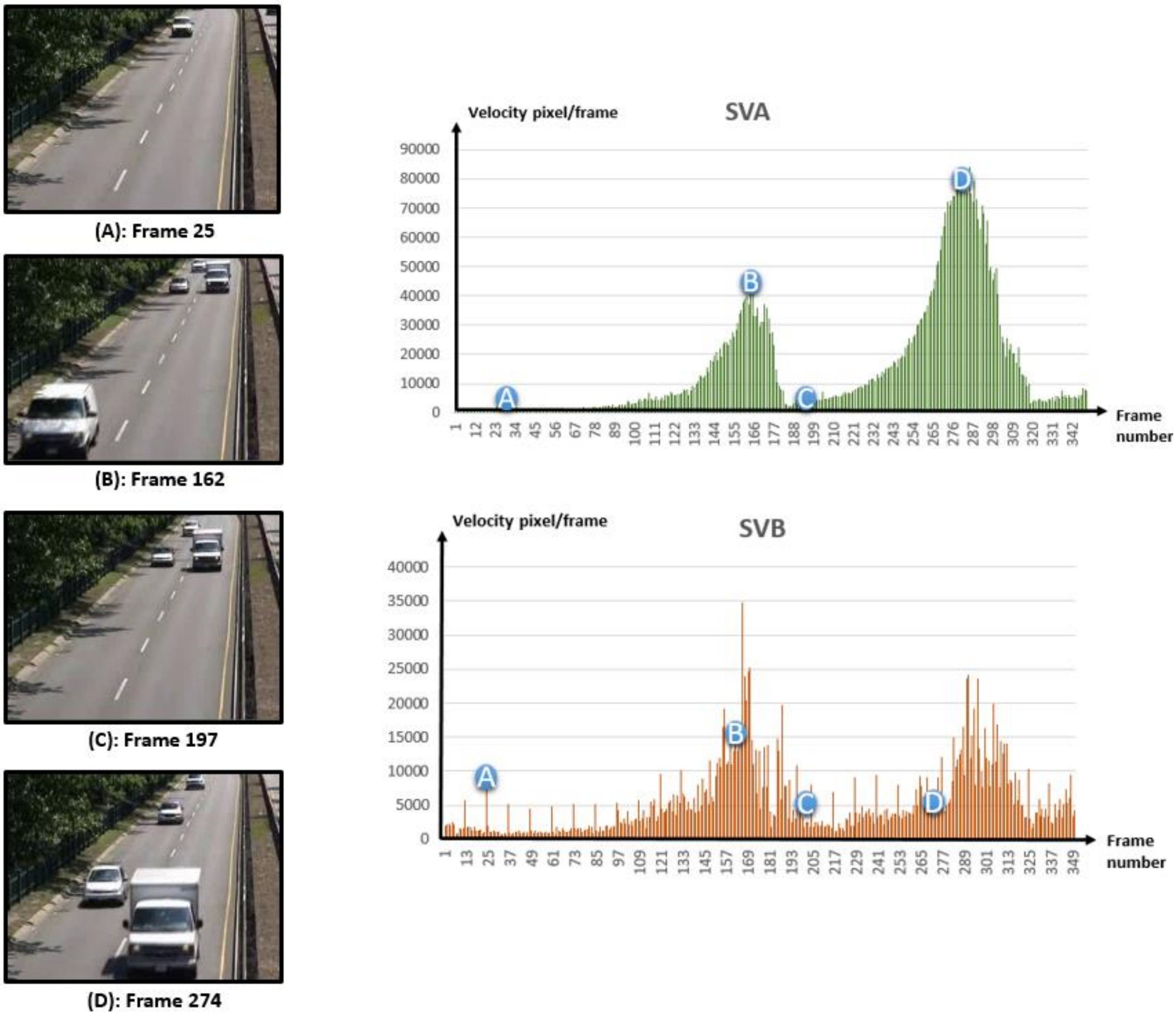


Figure 24: SVA and SVB variation as a function of frames for a normal traffic scene

We notice that, the closer the vehicles are to the camera, the more important the velocities magnitudes are. We assume that the variation of *SVA* and *SVB* follows the binomial parameter. So, we compute the average of the different *SVA* and *SVB* values and their respective standard deviation σ_1 and σ_2 for N frames. These N samples are saved as a list $L(N)$ of N *sub-models*. The thresholds *THA* and *THB* are *the normal traffic model* and are computed as formulated in equations (5) and (6).

$$THA = \frac{1}{N}\sum_{1}^{N} SVA(i) + \lambda\,\sigma_1 \tag{17}$$

$$THB = \frac{1}{N}\sum_{1}^{N} SVB(i) + \lambda\,\sigma_2 \tag{18}$$

λ is a tunable constant fixed to acquire higher precision for the model. N is also a parameter that will be experimentally determined.

3.2.3. Accident recognition: Adaptive threshold computation and accident detection

After N processed frames, we start the detection step by comparing every new *sub-model* extracted from the following frame to *the normal traffic model* thresholds. In case of exceeding values, an accident is reported. On the other hand, if the *sub-model* presents a normal traffic flow, the threshold will be updated using the late *sub-model* as a new sample in $L(N)$ and thus deleting the first saved *sub-model* sample. In this case, car queues change, traffic density, and luminosity change from day-to-night and night-to-day are not detected as an accident thanks to our adaptive threshold. This, on the other hand does not affect the precision of our system since these changes are progressive while an accident causes sudden changes. The flow chart of the algorithm with different steps is presented in Figure 25.

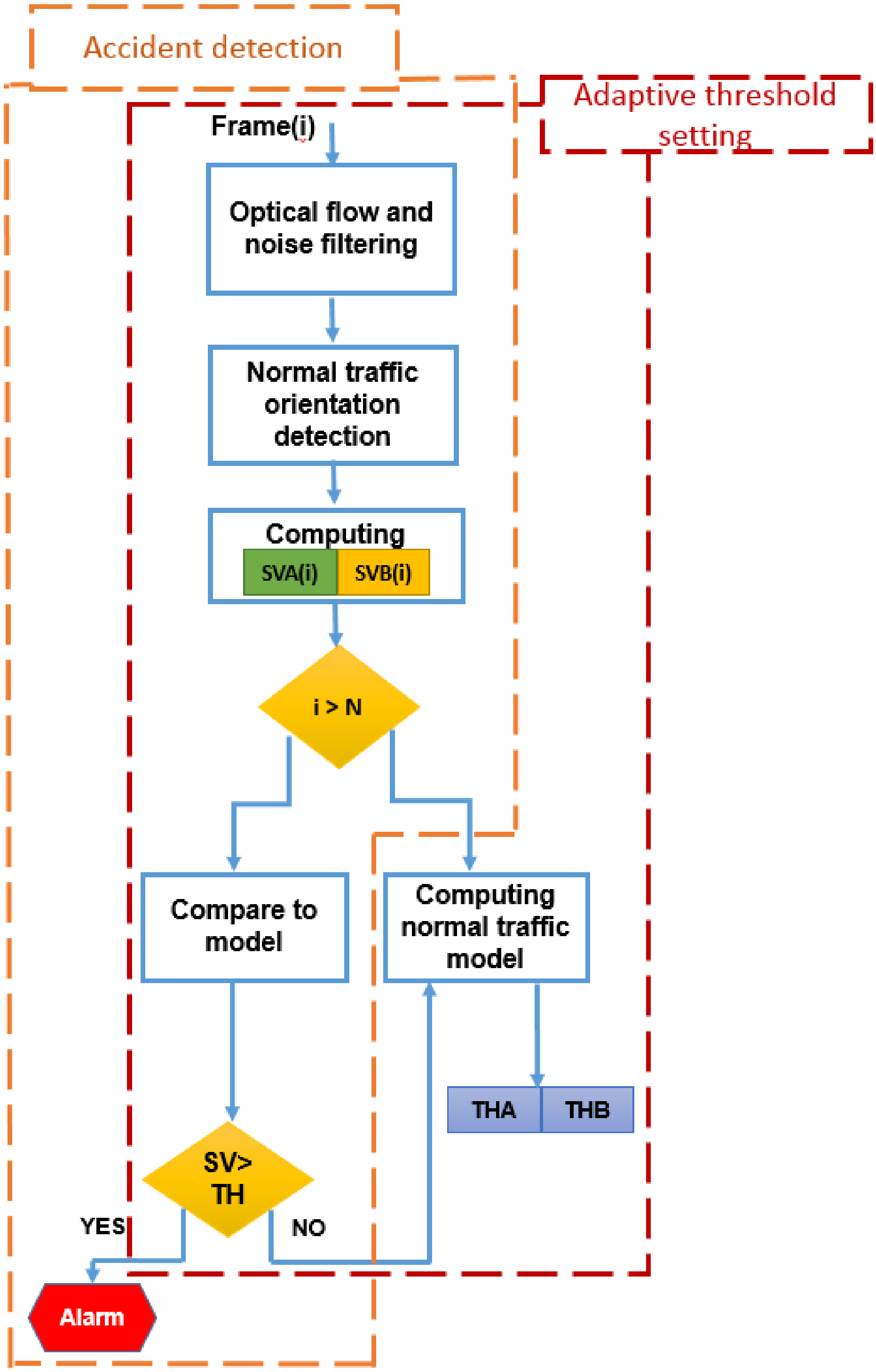


Figure 25: Adaptive threshold computation and accident detection flow chart.

3.2.4. Automatic traffic orientation detection

For instance, the orientation of traffic flow is known if the position of the camera is already known. While using existent videos, this orientation is known manually by observing the first frame and added to the algorithm as input. This step could be done automatically by developing an algorithm for the automatic detection of the direction of the traffic [θ_1, θ_2]. As explained before, after applying the Farnebäck optical flow, the majority of resultant vectors are in the direction of the traffic [θ_1, θ_2]. Finding the largest number of vectors in the same range of orientation allows the detection of the normal direction of the traffic flow.

3.2.5. Accident localization

An approximate accident localization could be made. For this step, instead of applying the algorithm to the whole ROI (which is a one way road, we segment the road into *i* zones (usually three depending on the range of sight of the camera) where we apply the algorithm to each part in parallel. In Figure 26 a schematic presentation of accident localization is shown.

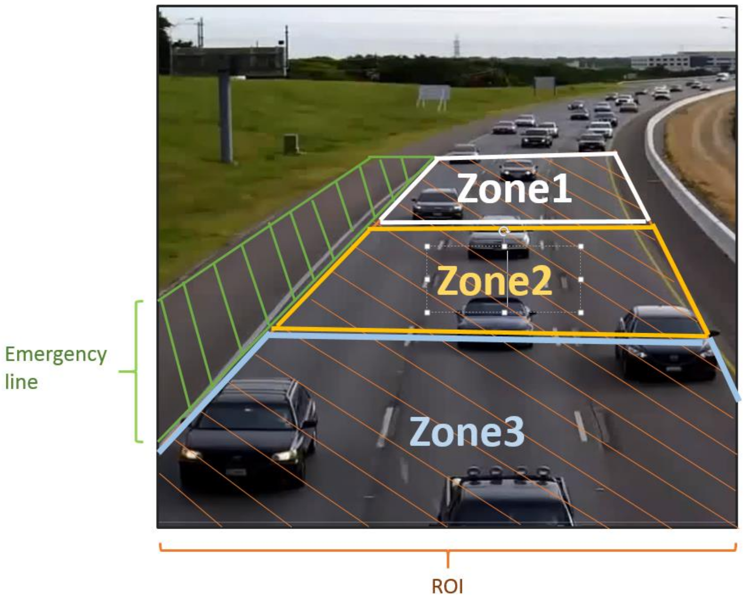


Figure 26: An example of road segmentation

4. Conclusion

In this chapter, different categories of algorithms of video based accident detection are described. The first one is based on vehicle detection and tracking, then a comparison of extracted features with a static threshold. The second ones are based on pixels in ROI tracking using Farnebäck OF. Different kinds of features are described. The basic idea of defining different approaches is to find the most accurate technique that offers the best features for accident recognition, while a comparison with the state of the art is difficult. The different metrics of these approaches and experimental results will be discussed in the next chapter.

Chapter 3: Evaluation results and performance analysis

Contents

In this chapter we present and discuss the results of the experimental test of our approaches. The algorithms are developed on Visual C++ 2010 express, running on a windows 7 machine with an Intel i7 processor. Experimentation was performed on different videos collected from websites as samples of different quality, orientation and environmental conditions, with normal traffic and accidents on highways. It helps in establishing parameters needed to compute histograms and threshold building for a calibration method and to show in which scenarios our algorithm succeeds.

1. Introduction

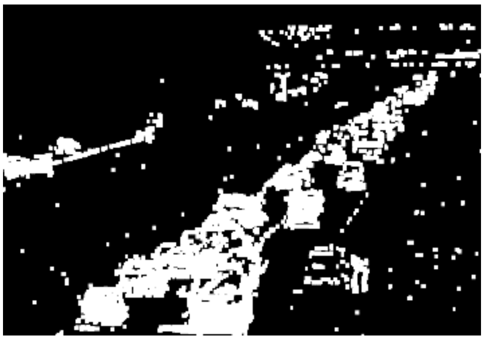
In the previous chapter, three different techniques for accident detection were presented; the first one is based on vehicle tracking and the other algorithms are based on particles tracking. The metrics of these different approaches such as thresholds are experimentally defined. First for each algorithm, the criteria of choice of each parameter and their values will be presented. Then after, results of accident detection will be presented and discussed in term of accuracy and performance.

2. Accident detection based on vehicle tracking

2.1. BS and object extraction experimental results

The BS algorithm is used in our approach to detect moving objects in the image. After applying the BS algorithm we noticed that in some cases blobs in the foreground image are close together and close objects could be seen as one. A case study is shown in Figure 27 using the GMM algorithm. With the purpose of finding out if an accurate BS exists, we compared different background subtraction algorithms..

a) A frame from a video sequence



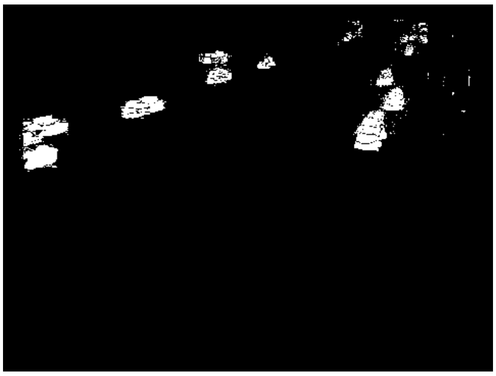
b) Foreground detection result

Figure 27: The result of the background subtraction on a frame, close together vehicles are seen as one object

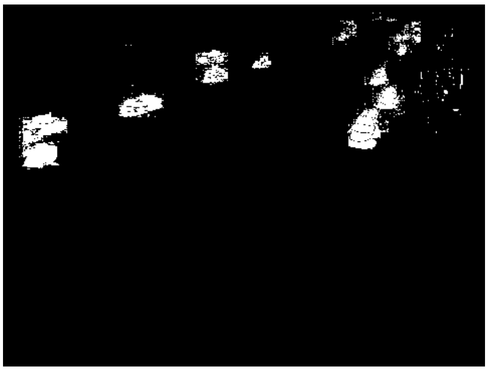
Table3 shows the results of the test of five different BS algorithms explained in the Chapter 1. As we see, the simple Gaussian BS gives the noisier foreground and the adaptive median BS algorithm gives a foreground with less information. However the Zicovic BS and Sigma delta BS has approximate results. The GMM results are more appreciated because the blobs that represent vehicles are clearer. GMM is widely used in video processing for it robustness, combined with morphological techniques, all moving object can be detected. In our algorithm we choose the GMM for the BS step.

a) Input frame

b) Adaptive median BS result

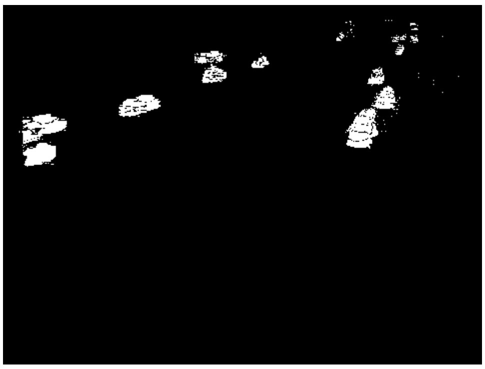


c) GMM Grimson BS result



d) Simple Gaussian BS result

e) Zicovic BS result

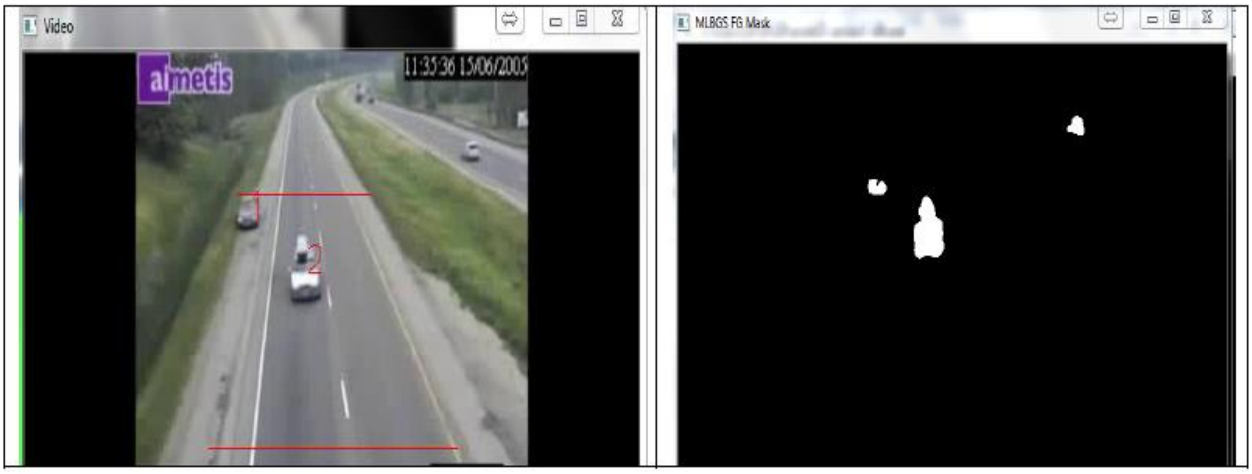


f) Sigma Delta BS result

Table 3: Different Background subtraction algorithm tested on an accident scene.

2.2. Vehicle tracking

Our algorithm is based on Kalman tracking in a region of interest. The Hungarian algorithm was used for the assignment of each blob to the corresponding vehicle. An example is shown in Figure 28. The red lanes are used to limit the ROI. For each vehicle a number is assigned. We assume that saving the parameters of each vehicle (velocity and orientation) for 10 consecutive frames is sufficient to detect a sudden change. Consequently, each vehicle has a motion history for 10 frames.



a) A frame from a video sequence b) Foreground detection result

Figure 28: An example of vehicle tracking. The red lines are the limitation of the ROI. The numbers (1) and (2) are the assignment of each vehicle.

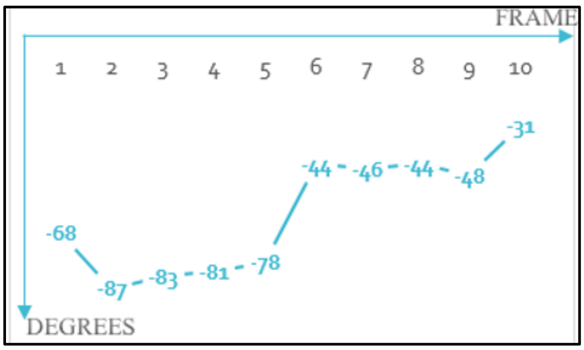
2.3. Threshold computation and accident detection

When a sudden event occurs, the orientation of the vehicle and its velocity change. To detect these changes we compare the parameters of each vehicle in each frame to its motion history.

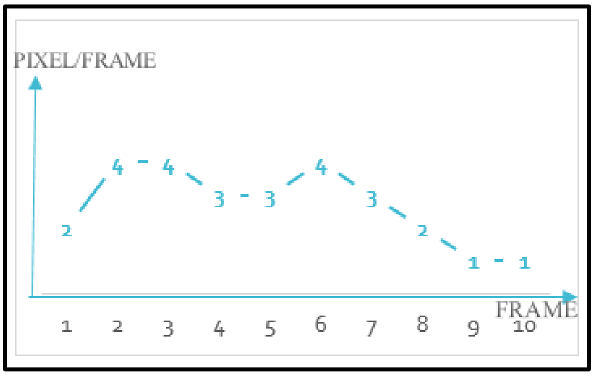
The curves in Figure 29 present the variation history of the orientation and the velocity of two vehicles (Car (4) and Car (5)) for 10 frames. The unit of orientation is degrees and the unit of velocity is pixels per frame ($\frac{\Delta p}{\Delta t}$). Car (5) has normal motion.

Car (4) has abnormal behavior. We notice that for Car (5) the orientation is increasing slightly, which reflects normal motion of the vehicle compared to the position of the camera, while the orientation of Car (4) is continually changing, reflecting abnormal behavior. The velocity of Car (5) is not really changing while the velocity of Car (4) is decreasing which means that the vehicle is going to stop. From these observations, we conclude that analyzing motion features of 10 consecutive frames is enough to detect abnormal behavior of a vehicle.

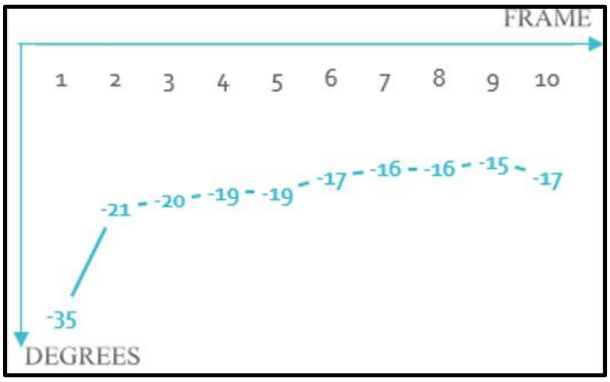
a) A traffic scene



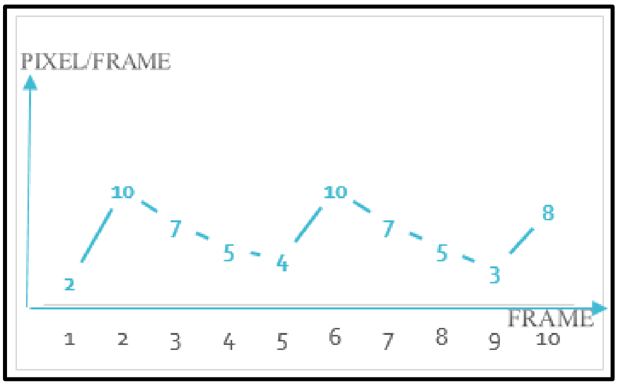
b) The variation of the orientation of the car 4



c) The variation of the velocity of the car 4



d) The variation of the orientation of the car 5



e) The variation of the velocity of the car 5

Figure 29: Velocity and orientation variation of two different vehicles in a traffic scene: Car(4) with abnormal motion and car(5) with normal motion

The performance of this approach is shown in table 4. This approach helps to detect different scenarios besides accidents, which are: stopped vehicle detection and traffic jam detection. The accuracy of the algorithm is computed using the equation (19). The time execution of the most consuming parts of the algorithm is presented in the table too.

$$Accuracy = Detection\ rate = \frac{Detected\ accident}{Total\ number\ of\ accidents} \quad (19)$$

Vehicle abnormal behavior detection	Traffic jam detection	Stopped vehicle detection	Accident detection	Sensibility to noise and change of illumination	Accuracy	Time execution per frame with 480*360 resolution	
						BS	Kalman & Hangarian algorithms
Yes	Yes	Yes	Yes	Yes	75%	0.007s	0.002s

Table 4: Experimental results of the first approach

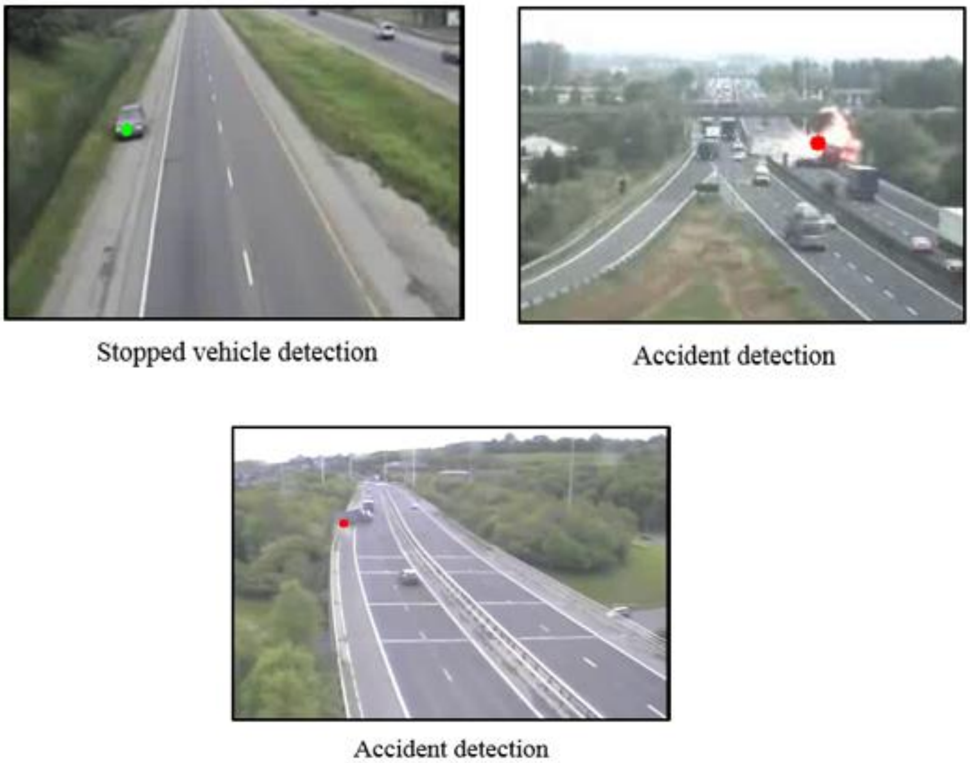


Figure 30: Some annotated detected accidents

Figure BS and tracking are very sensitive to changes in luminosity and the presence of occlusion: in some cases two close together vehicles are seen as one which affects the quality of tracking. This explains the accuracy level of 75%. In this case techniques of segmentation or shadow elimination must be used. Figure 30 shows some annotated accidents using this algorithm.

3. Accident detection based on particle tracking

3.1. First approach: defining histogram and fixed thresholds computation

We notice, after experimentations, that the method 1 is more suitable for accident detection. The method 2 and the method 3 explained in the previous chapter, do not perform a histogram sensitive to the changes of the traffic situation. The value of the obtained histograms were difficult to analyze. Consequently, it is not possible to detect an accident using these two approaches.

To enhance the accuracy of the method 1, we tried to find out the best *k* width of the blocks. Tests were done on different accident videos using 3*3, 5*5, 7*7, 9*9 blocks as presented in Table 5. We computed the time execution and in which frame the accident was detected. We noticed that the optimal value of *k* is 9 since the time of execution is shorter and, for some cases, the accident is detected earlier (example video 1).

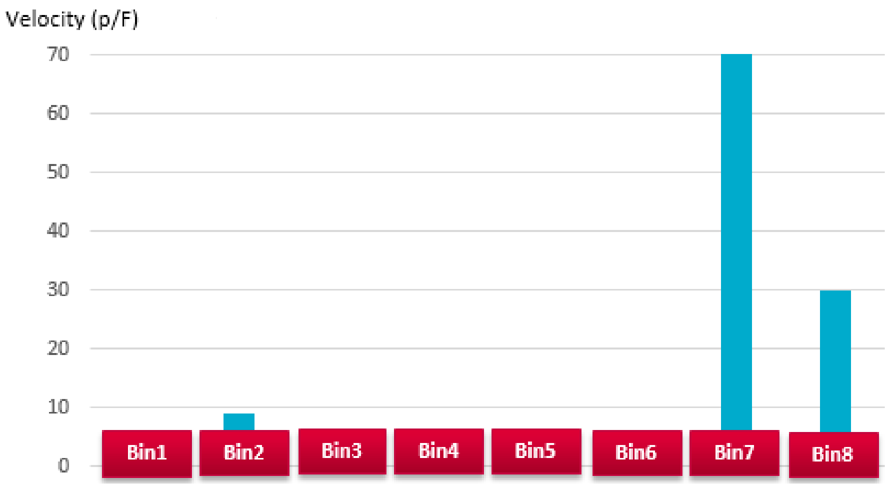
		k = 3	k = 5	k = 7	k = 9
Video 1 -Total number of frames: 500 -Resolution: 480*360 - Frequency: 30f/s	Computation time (seconds)	97,13	95,18	96,3	94,45
	Detection performed at frame number:	394	406	406	393
Video 2 - Total number of frames: 380 -Resolution: 1280*720 -Frequency: 30f/s	Computation time (seconds)	363,33	292,48	286,12	285,74
	Detection performed at frame number:	194	194	194	194
Video 3 -Total number of frames: 200	Computation time (seconds)	31,11	31,11	31,09	31,07

-Resolution: 480*360 -Frequency: 30f/s	Detection performed at frame number:	119	119	119	119

Table 5: A comparative study of the detection rates for different videos using different block sizes.

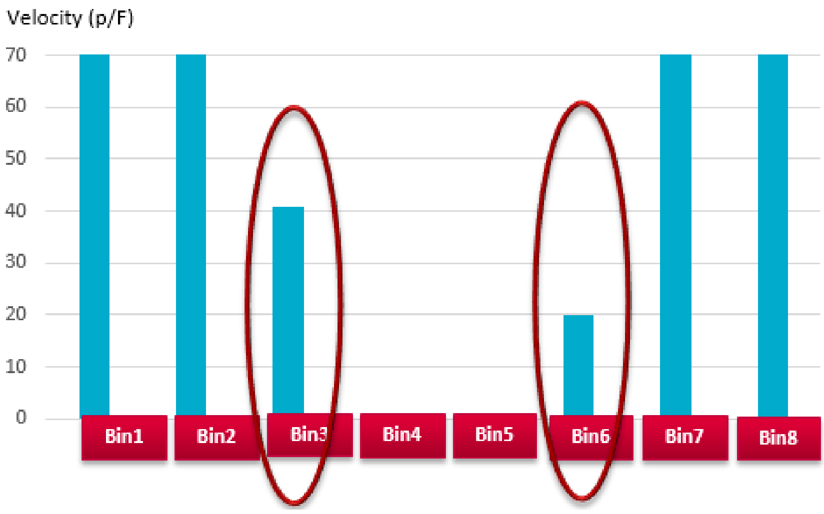
Figure 31 shows the histograms of two frames from the same traffic video before and after an accident. We notice that a slight change occurs in some bins in the case of an accident.

a) Frame 200



b) Zoom on the histogram of the frame 200

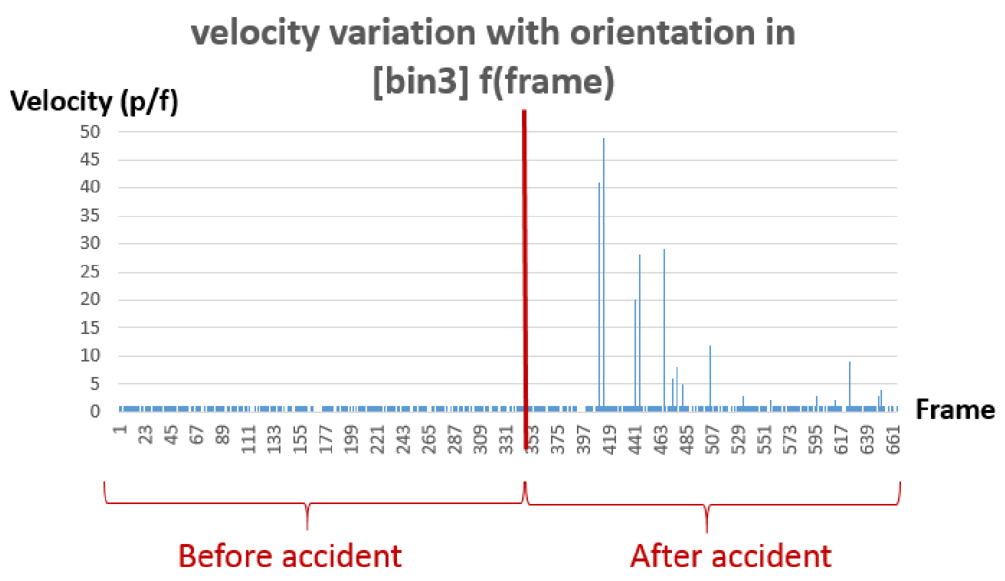
c) Frame 410



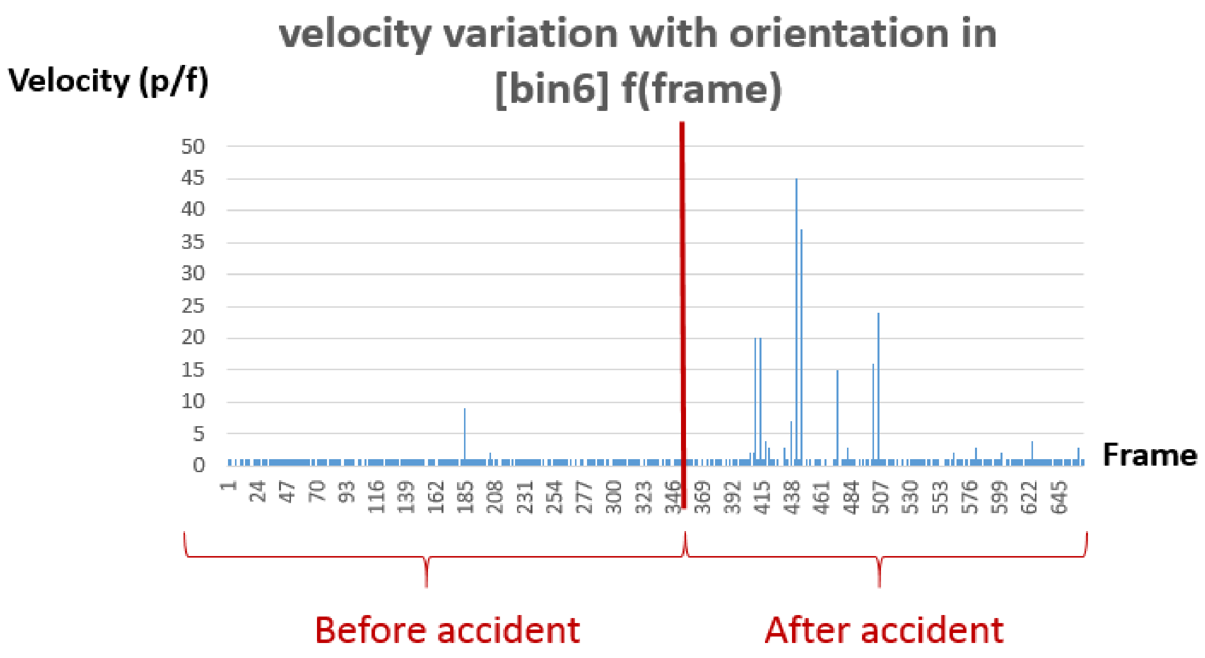
d) The histogram of the frame 410

Figure 31: An example of histogram results before and after accident. For this example, bin3 and bin6 change slightly when an accident occurs

Figure 32 shows the curve of the variation of the value of bin3 and bin 6 per frame. We notice a slight changes after an accident. In case of a normal traffic scene, in the selected video, the vehicles' motion does not follow the range of the orientation of bin3 and bin6. However, after an accident, motion following these orientations appears suddenly, which reflects the presence of velocity vectors with these orientations so an abnormal behavior on the road. These sudden changes appear between two successive frames which confirm our approach presented in chapter 2 paragraph 3.



(a)



(b)

Figure 32: Velocity flowing the range of orientation of bin3 (a) and bin6 (b) variation in function of frames.

The performance of this approach is shown in table 6. This approach helps to detect accidents and traffic jams. Since the analysis is carried out on all the particles of a ROI, information about only one vehicle is difficult to extract. The accuracy of the algorithm is demonstrated by the detection rate presented by equation (19). The time execution of the most consuming part of the algorithm is presented in the table 6 too. For the OF, the time execution is more than one second which means that the approach does not respect the real time detection and need acceleration.

Vehicle abnormal behavior detection	Traffic jam detection	Stopped vehicle detection	Accident detection	Sensibility to noise and changes in illumination	Accuracy	Time execution per frame with 480*360 resolution	
						OF	Histogram computation
No	Yes	No	Yes	no	75%	0.145	0.001s

Table 6: Experimental results of the algorithm based on histogram and fixed threshold computation

3.2. Second approach: Traffic modeling and dynamic threshold computation results

First the threshold ε for noise filtering was defined. This threshold is calculated by performing OF on a fixed number of frames in which no traffic is detected. In our case, we assume that 10 frames with no traffic are sufficient to calculate ε. This value differs from one video to another depending on the calibration of the camera, the quality of the video, etc. The next step is to compute the traffic model. This model is based on building a calibration method to compute *THA* and *THB*. *N* (the number of frames for modeling) and λ are the constants that define these thresholds. To obtain adequate parameters giving the most accurate results, the algorithm was tested on videos using different combinations of values of *N* and λ. The accuracy of the obtained results is measured using the F_{1_score} is formulated in equations (20), (21) and (22). The value *p* represents the precision given by the result of the division of the number of Correct Positive Results (*CPR*) by the number of All Positive Results (*APR*). The value of *r* represents the recall given by the result of the division of the *CPR* by the Effective Positive Results (*EPR*) that should have been returned.

$$F_{1\_score} = 2.\frac{p.r}{p+r} \quad (20)$$

$$p = \frac{CPR}{APR} \quad (21)$$

$$r = \frac{CPR}{EPR} \quad (22)$$

We present the results of the tests of the different combinations of *N* and λ as curves in Figures 33 and 34. In Figure 33, the horizontal axis represents the number of frames. The first vertical axis represents the values of *p* and *r*. We notice that for *N* = 240 and λ equal to one we achieve better precision and recall values.

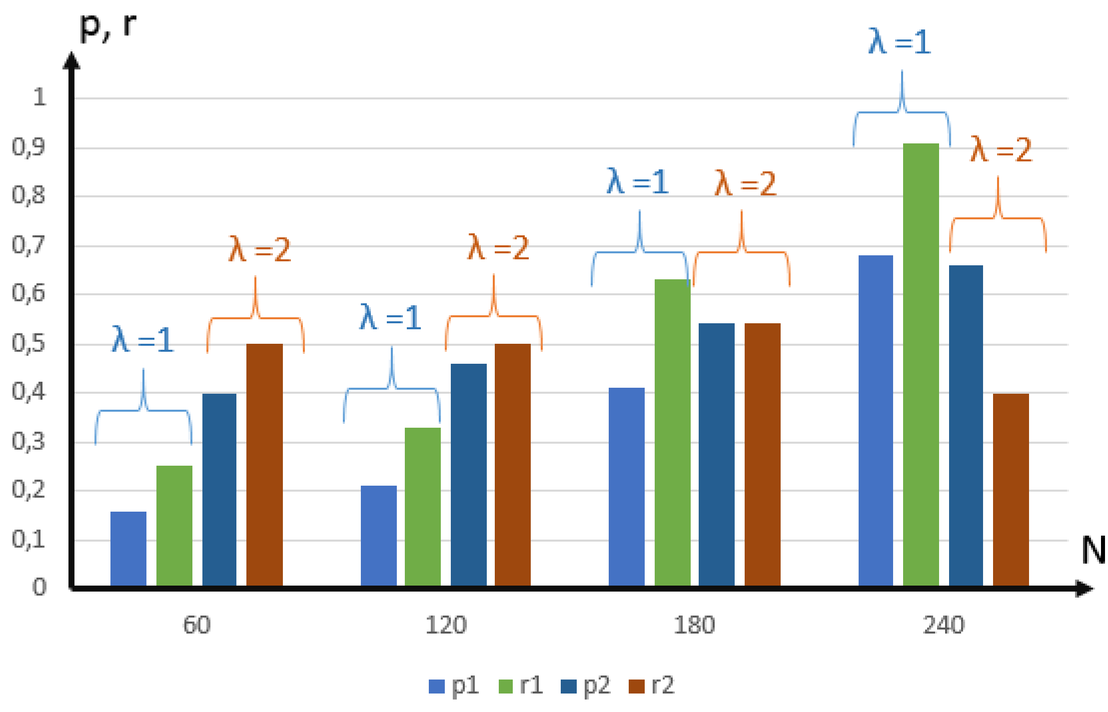


Figure 33: Variation in p and r for different values of N and λ

In the Figure 34 the horizontal axis represents the number of frames and the vertical axis represents the values of the F_{1_score}. The blue curve presents the values of the F_{1_score} as a function of *N* for lambda equal to one and the orange one presents the variation in the values of the F_{1_score} as a function of *N* for lambda equal to two. The results show that for N equal to 240 and λ equal to one, we achieve the highest value of the F_{1_score} , equal to 0.77. While for N equal to 60 and λ equal to one we achieve the lowest value of the F_{1_score} , equal to 0.19.

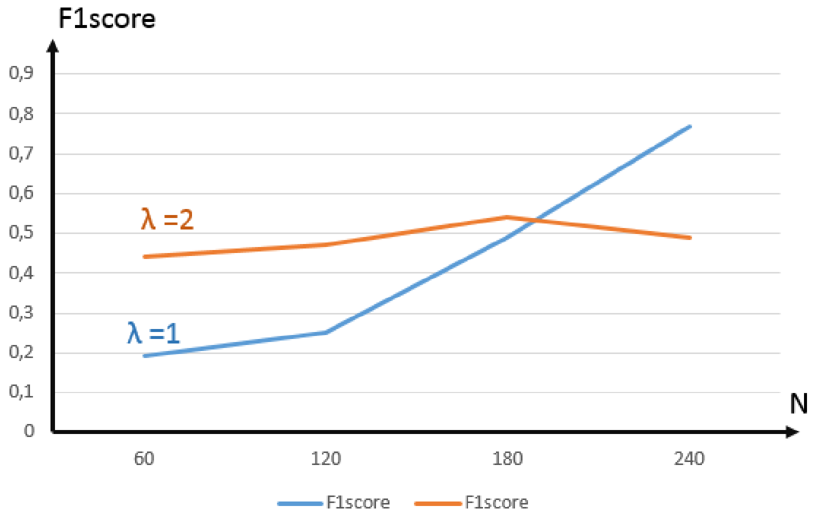


Figure 34: Variation in the F_{1_score} for different values of λ and N

The performance of this approach is shown in table 7. This algorithm helps to detect accidents, traffic jams and abnormal behavior of vehicles. The accuracy of the algorithm is computed using the equation (19). It is less sensitive to the change of luminosity and its accuracy is sufficient. However the time of execution needs acceleration to meet the real time constraints.

Vehicle abnormal behavior detection	Traffic jam detection	Stopped vehicle detection	Accident detection	Sensibility to noise and changes in illumination	Accuracy	Time execution per frame with 480*360 resolution
Yes	Yes	No	Yes	no	90%	0,4 s

Table 7: Experimental results of the algorithm based on adaptive threshold computation

Figure 35 shows some annotated accidents detected by our approach.

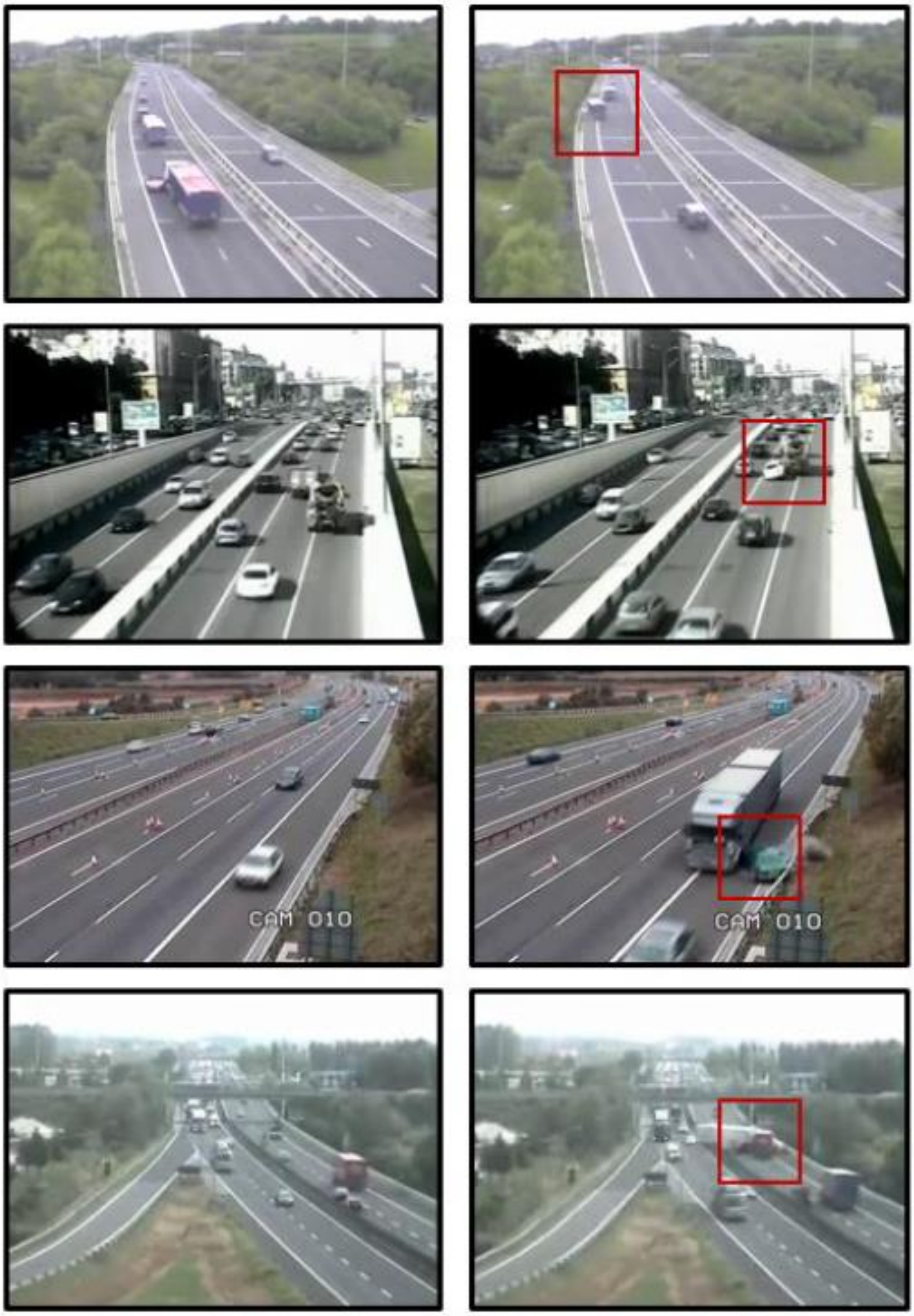


Figure 35: Examples of detected accident scenes with the proposed approach. Accidents are annotated by rectangles

4. Conclusion

In this chapter we presented the experimental results of the different approaches developed and explained in chapter 2. The computation of thresholds and the different metrics of each approach are presented. These metrics depend on the calibration and position of cameras. In our case study, this information is not known since we used videos from the internet.

We notice that the approach based on tracking vehicles is good enough for abnormal behavior detection and accident detection in the case of using a high quality camera and a robust BS algorithm. However, the algorithms based on particle tracking help detect abnormal behavior on a global scale including traffic jams and even with a poor quality video. To avoid fixing thresholds experimentally, we proposed the approach based on adaptive threshold computation which is more robust to noise and provides sufficient results. Nevertheless, this approach seams computationally expensive since tracking was performed on every pixel of the ROI. These limitations can be improved by hardware implementation to achieve real time detection.

Conclusion

For the past few decades, automatic detection of road accidents by video surveillance has become a very important issue for many reasons cited below:

- ✓ Making the surveillance cameras installed increasingly on roads more useful, by reducing human monitoring.
- ✓ Enhancing traffic management after an accident, reducing traffic jams and time loss and avoiding accident escalation.
- ✓ Providing rapid assistance for injuries to save more lives by respecting the Golden Hour constraint.
- ✓ Automatically extracting more details about accidents such as the number of vehicles involved and the number of injuries.
- ✓ Enhancing existent accident detection systems based on sensors (for example, Ecall) by providing complementary information.
- ✓ Cases of Emergency and Disaster Management.
- ✓ Providing storage of accident scenes for evidence and further analysis.

In literature, there are two categories of approaches for accident detection. The first one is based on vehicle motion analysis and the second one is based on relevant pixel analysis. The comparison between the state of the art different approaches in terms of accuracy is still a complicated task since the used datasets are not common. We noticed that existent approaches based on pixel motion analysis are more robust than the approaches based on vehicle motion analysis.

With the purpose of confirming this hypothesis, we proposed different algorithms for accident detection on a segment of road. One algorithm is based on vehicle tracking using the Kalman filter. Accident recognition is performed by comparison to experimentally predefined thresholds. Then we proposed algorithms based on the tracking of all pixels in the image using Farnebäck dense optical flow. For these approaches, we studied different methods for feature extraction and threshold computation.

The contribution of our work is summarized below:

Contribution 1:

We developed an accident detection algorithm based on vehicle tracking using the Kalman filter, velocity and orientation extraction, and comparison to a predefined threshold. Experimental results show that the proposed algorithm is good enough for abnormal behavior detection and accident detection in the case of using a high quality camera and a robust BS algorithm.

Contribution 2:

We developed different algorithms based on particle tracking. Farnebäck dense optical flow was used. A histogram of orientation was computed as a feature. A static predefined threshold was experimentally defined to detect an accident. This approach helps detect abnormal behavior on a global scale including traffic jams, even with a poor quality video. However, defining a different threshold for each video is complicated and makes the approach more sensitive to changes in luminosity. This algorithm is good enough in cases when the camera calibration is already known.

Contribution 3:

To avoid using a static threshold, while taking advantage of using pixel tracking (Farnebäck OF), we proposed an automatic adaptive threshold computed automatically taking into account the changes in luminosity. This approach is more robust to noise and provides sufficient results, achieving an F1 score of 0.77. Nevertheless, this approach seams computationally expensive since tracking was performed on every pixel of the ROI. These limitations can be improved by hardware implementation to achieve real time detection.

Perspectives:

The proposed algorithms have to be improved on several levels:

1. The accuracy of detection:

Deep learning approaches are used more and more nowadays in the computer vision field. Combining these techniques with our approach could enhance the detection rate. So, finding a solution to create a big dataset of road accidents helps create a robust algorithm based on deep learning techniques.

2. Detected scenarios:

The proposed approaches are applied only for a specific traffic scenario: highways, during the daytime and on a sunny day. The impact of changing the zone of detection (intersection, tunnel, bridges, etc.) or time of day or the weather conditions has not been studied yet. These metrics could affect the accuracy of our approaches which will need more improvement.

3. Real time constraints:

For instance, the proposed approaches are computationally expensive using the OF. Experimentations show that dense OF performs a computation time of 1f/s for a video of 1280x720 resolution running on an ARM cortex A9. Therefore, algorithm optimization and a specific hardware design implementation, based on FPGA (Field-Programmable Gate Array) for example, is mandatory to accelerate the algorithm, ensuring real time detection. In literature there are different approaches for OF acceleration on FPGA such as [Seyid et al, 2016], [Monson et, al. 2013] and [Rustam et al, 2012]. Xilinx[11] proposes also, a hardware acceleration for the dense OF on a Zync FPGA using the OpenCV libraries. This researchers open up a new study trail for an effective real time accident detection by video surveillance.

[11] https://forums.xilinx.com/t5/Xcell-Daily-Blog/Dense-Optical-Flow-hardware-acceleration-on-Zynq-SoC-made-easier/ba-p/745152

References

[Lerner et al., 2001] Lerner, EB. and Moscati: The Golden Hour: Scientific Fact or Medical Urban Leg-end. Academic Emergency Medicine 8 (7):758–760. doi:10.1111/j.1553-2712.2001.tb00201. (2001).

[Farnebäck, 2003] Farnebäck, G. Two-Frame Motion Estimation Based on Polynomial Expansion. Heidelberg: Springer Berlin Heidelberg. pp. 363-370. Berlin. (2003).

[Bunch et al., 2011] Buch, N., Velastin, S.A. and Orwell, J.: A Review of Computer Vision Techniques for the Analysis of Urban Traffic. IEEE Transaction on Intelligent Transportation Systems. Vol. 12, no. 3. (2011).

[Fishbain and Yaroslavsky, 2009] Fishbain, B., Yaroslavsky, L.P.: Real-time vision-based traffic flow measurements and incident detection. Proceedings of the International Society for Optical Engineering (SPIE). (2009).

[Parkany and Chi Xi, 2005] Parkany, E. and Chi Xi, P.: A Complete Review of Incident Detection Algorithms and Their Deployment: What Works and What Doesn’t. The New England Transportation Consortium. (2005).

[Rossi et al. 2015] Rossi, R., Gastaldi, M., Gecchele, G. and Barbaro, V.: Fuzzy logic-based incident detection system using loop detectors data. 18th Euro Working Group on Transportation (EWGT). (2015).

[Kim, 2008] Kim, J-Y.: An Analysis on Acoustic Characteristic of the Intersection Noise for Performance Improvement in the Accident Sound Detection System. The third International Conference on Convergence and Hybrid Information Technology. (2008).

[White and all, 2011] White, J., Thompson, C., Turner, H., Dougherty, B. and Schmidt, D.C. WreckWatch: Automatic Traffic Accident Detection and Notification with Smartphones. Mobile Network and Application. Sringer. (2011).

[Qianyin et al. 2015] Qianyin, J., Guoming, L., Jinwei, Y. and Xiying, L. A model based method of pedestrian abnormal behavior detection in traffic scene. *IEEE First International Smart Cities Conference (ISC2)*. pp. 1-6. Guadalajara. (2015).

[Hou et al. 2013] Hou, A. L., Guo, J. L., Wang, C. J., Wu, L. and Li, F. Abnormal Behavior Recognition Based on Trajectory Feature and Regional Optical Flow. *Seventh International Conference on Image and Graphics*. pp. 643-649. Qingdao. (2013).

[Moris and Trivedi, 2008] Morris, B. T. Trivedi, and M. M. A Survey of Vision-Based Trajectory Learning and Analysis for Surveillance. IEEE Transactions on Circuits and Systems for Video Technology, vol. 18, no. 8, pp. 1114-1127. (2008).

[Cui et al., 2011] Cui, L., Li, K., Chen, J. and Li, Z. Abnormal Event Detection in Traffic Video Surveillance Based on Local Features. 4th International Congress on Image and Signal Processing. (2011).

[Li t al. 2015] Li, Y., Liu, W. and Huang, Q. Traffic anomaly detection based on image descriptor in videos. Springer Science and Business. (2015).

[Cai et al. 2015] Cai, Y., Wang, H., Chen, X. and H. Jiang. Trajectory-based anomalous behavior detection for intelligent traffic surveillance. *IET Intelligent Transport Systems*, vol. 9, no. 8, pp. 810-816,(2015).

[Mehboob et al, 2016] Mehboob, F., Abbas, M. and Jiang, R.Traffic event detection from road surveillance videos based on fuzzy logic. SAI Computing Conference, pp.188-194. London. (2016).

[McFarlane and Schofield, 1995] McFarlane, N. and Schofield, C. Segmentation and tracking of piglets in images. Machine Vision and Applications, pp. 187–193. (1995).

[Calderara et al., 2006] Calderara, S., Melli, R., Prati, A., and Cucchiara, R. Reliable background suppression for complex scenes. Proceedings of the 4th ACM International Workshop on Video Surveillance and Sensor Networks, VSSN ’06, pp. 211–214, New York. (2006).

[Wren et al., 1997] Wren, C., Azarbayejani, A., Darrell, T., and Pentland, A. Real-time tracking of the human body. IEEE Transactions on Pattern Analysis and Machine Intelligence, pp. 780–785. (1997)

[Benezeth et al., 2008] Benezeth, Y., Jodoin, P.-M., Emile, B., Laurent, H., and Rosenberger, C. Review and evaluation of commonly-implemented background subtraction algorithms. 19th International Conference on Pattern Recognition ICPR. pp. 1–4 (2008).

[Stauffer and Grimson, 1999] Stauffer, C. and Grimson, W. Adaptive background mixture models for real-time tracking. IEEE Computer Society Conference on in Computer Vision and Pattern Recognition, pp –252 Vol. 2. (1999).

[KaewTraKulPong and Bowden, 2002] KaewTraKulPong P., Bowden R. An Improved Adaptive Background Mixture Model for Real-time Tracking with Shadow Detection. Video-Based Surveillance Systems. Springer, Boston. (2002).

[Zivkovic, 2004] Zivkovic, Z. Improved adaptive Gaussian mixture model for background subtraction. Proceedings of the 17th International Conference on Pattern Recognition (ICPR), pp. 28-31 Vol.2. (2004).

[Zhao et al, 2012] Zhao Z., Bouwmans T., Zhang X. and Fang Y. A Fuzzy Background Modeling Approach for Motion Detection in Dynamic Backgrounds. Multimedia and Signal Processing. Communications in Computer and Information Science, vol 346. Springer, Berlin, Heidelberg. (2012)

[Barnich and Van Droogenbroeck, 2011] Barnich, O. and Van Droogenbroeck, M. Vibe: A universal background subtraction algorithm for video sequences. IEEE Transactions on Image Processing, pp. 1709–1724. (2011).

[Lee, 2012] Lee, I-J. and Oh, S.S: An Accident Detection System on Highway through CCTV with Calogero-Moser System. Information Engineering Vol. 2. (2013)

[Lukas and Kanade, 1981] Lucas, B.D. and Kanade, T. An iterative image registration technique with an application to stereo vision. Proceedings of the International Joint Conference on Artificial Intelligence, pp. 674–679. (1981).

[Horn and Schunck, 1981] Horn, B.K.P. and Schunck, B.G. Determining optical flow. Artificial Intelligence, pp.17:185–203. (1981).

[Farnebäck, 2003] Farneback, G. Two-Frame Motion Estimation Based on Polynomial Expansion. Springer, pp. 363-370. Berlin. (2003).

[Lowe, 1999] Lowe, D. G. Object recognition from local scale-invariant features. Proceedings of the International Conference on Computer Vision (ICCV), pp. 1150–11 570. (1999).

[Morel, 2011] Morel, J.M. Is sift scale invariant ?. Inverse Problems and Imaging, vol. 5(1), 2011. [Online]. Available : http://www.aimSciences.org xv, 171

[Mantripragada et al, 2014] Mantripragada, K., Trigo, F. C. and Martins, F. P. R. Vehicle Tracking Using Features Matching and Kalman Filtering. ABCM Symposium Series in Mechatronics - Vol. 6. (2014).

[Chen et al., 2016] Chen, Y., Yu, Y. and Li, T. A vision based traffic accident detection method using extreme learning machine. The international Conference on Advanced Robotics and Mechatronics (ICARM), pp. 567–572. (2016).

[Bay et al., 2006] Bay, H., Tuytelaars, T., and Van Gool, L. SURF: Speeded up robust features. Computer vision–ECCV. Springer, pp. 404–417. (2006).

[Li and Zhang, 2013] Li, J. and Zhang, Y. Learning surf cascade for fast and accurate object detection. IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 3468–3475. (2013).

[Dalal and Triggs, 2005] Dalal, N. and Triggs, B.: Histograms of oriented gradients for human detection. IEEE Conference on Computer Vision and Pattern Recognition (CVPR). (2005).

[Viola and Jones, 2001] Viola, P. and Jones, J.M. Rapid Object Detection using a Boosted Cascade of Simple Features. IEEE Conference on Computer Vision and Pattern Recognition CVPR. (2001).

[Wu and Weng, 2017] Wu, C. C. and Weng, K. W. The detecting and tracking system for vehicles. 10th International Conference on Ubi-media Computing and Workshops (Ubi-Media), pp. 1-5. Pattaya, (2017).

[Chen et al, 2017] Chen, X., Gong, R. X., Xie, L. L., Xiang, S., Liu, C. L. and Pan, C. H. Building Regional Covariance Descriptors for Vehicle Detection. IEEE Geoscience and Remote Sensing Letters, vol. 14, no. 4, pp. 524-528. (2017).

[Kim et al, 2015] Kim, J., Baek J. and Kim, E. A Novel On-Road Vehicle Detection Method Using πHOG. IEEE Transactions on Intelligent Transportation Systems, vol. 16, no. 6, pp. 3414-3429. (2015).

[Haritha and Ramadevi, 2013] Haritha and Ramadevi, Ch. Vehicle detection and tracking by using dynamic bayesian networks. National Conference on Research Issues and Recent Trends in Computer Science & Information Technology (NCRRCSIT). (2013).

[ki and Lee, 2007] Ki, Y. K. and Lee, D. Y. A traffic accident recording and reporting model at intersections. IEEE Transactions on Intelligent Transportation Systems, vol. 8, no. 2, pp. 188–194. (2007).

[Hui et al, 2014] Hui, Z., Yaohua, X., Lu, M., and Jiansheng, F. Vision-based real-time traffic accident detection. 11th World Congress on Intelligent Control and Automation (WCICA), pp. 1035–1038. (2014).

[Akooz and Karsligil, 2010] Akooz, O. and Karsligil, M. E. Severity detection of traffic accidents at intersections based on vehicle motion analysis and multiphase linear regression.

13th International IEEE Conference on in Intelligent Transportation Systems (ITSC), pp. 474–479. (2010).

[Sadek et al, 2010] Sadek, S., Al-Hamadi, A., Michaelisy, B. and Sayed U. Real-time automatic traffic accident recognition using HFG. 20th International Conference on Pattern Recognition (ICPR),pp. 3348–3351. (2010).

[Ahmadi et al., 2016] Ahmadi, P., Tabandeh, M. and Gholampour. I. Abnormal event detection and localization in traffic videos based on group sparse topical coding. IET Image Processing, vol. 10, no. 3, pp. 235–246. (2016).

[Ullah et al., 2015] Ullah, H., Ullah, M., Afridi, H., Conci, N., and Natale, F. G. B. D. Traffic accident detection through a hydrodynamic lens. IEEE International Conference on Image Processing (ICIP), pp. 2470–2474. (2015).

[Ren et al., 2016] Ren, J., Chen, Y., Xin, L., Shi, J., Li, B, and Liu, Y. Detecting and positioning of traffic incidents via video-based analysis of traffic states in a road segment. IET Intelligent Transport Systems. vol. 10, no. 6, pp. 428–437. (2016).

[Chen et al, 2016] Chen, Y., Yu, Y., and Li, T. A vision based traffic accident detection method using extreme learning machine. International Conference on Advanced Robotics and Mechatronics (ICARM), pp. 567–572. (2016).

[Seyid et al, 2016] K. Seyid, A. Richaud, R. Capoccia, and Y. Leblebici. Fpga based hardware implementation of real-time optical flow calculation. IEEE Transactions on Circuits and Systems for Video Technology, vol. PP,no. 99, pp. 1–1, (2016).

[Monson et, al. 2013] J. Monson, M. Wirthlin, and B. L. Hutchings. Implementing highperformance, low-power fpga-based optical flow accelerators in c. IEEE 24th International Conference on Application-Specific Systems, Architectures and Processors, pp. 363–369.(2013)

[Rustam et al, 2012] R. Rustam, N. H. Hamid, and F. A. Hussin, Fpga-based hardware implementation of optical flow constraint equation of horn and schunck. The 4th International Conference on Intelligent and Advanced Systems, pp. 790–794. (2012).

List of publications

- Maaloul Boutheina, Taleb Ahmed Abdelmalik, Niar Smail, Harb Naim, Valderrama Carlos, "Adaptive Video-Based Algorithm for Accident detection on Highways" in "International Symposium on Industrial Embedded Systems" , Toulouse, France (2017)

- Maaloul Boutheina, Taleb Ahmed Abed Al Malek, Niar Smail, Valderrama Carlos, Derraz Foued, Harb Naim, "Vision-based Traffic Accident Detection Techniques Opportunities and Challenges" in The 2015 International Conference on Advanced Communication Systems and Signal Processing (2015)

Résumé

Les systèmes automatiques de vidéo surveillance ont été développés pour détecter et analyser les comportements anormaux et les situations de risque dans de nombreux domaines. L'une des applications de la vidéosurveillance est la surveillance du trafic. L'analyse du mouvement dans les routes vise à détecter les comportements de circulation anormaux et les événements soudains. Les accidents de la route peuvent causer des blessures graves touchant principalement la tête et le cerveau, entraînant des handicapes et même la mort. Chaque minute supplémentaire pour assister les blessés fait la différence entre la vie et la mort, ce qui est révélée par l'Heure d'or (Golden Hour). Par conséquent, fournir une assistance rapide pour les blessés est obligatoire. De plus, les accidents peuvent causer des embouteillages entraînant d'éventuels autres accidents s'ils ne sont pas notifiés rapidement. Par conséquent, il est important de concevoir un système capable d'organiser une réponse d'urgence efficace. Cette réponse devrait être basée, d'une part sur une détection automatique par analyse vidéo, puis sur une notification rapide.

De nombreuses villes en France sont équipées de caméras de surveillance installées sur différentes routes. La surveillance du trafic est effectuée par des opérateurs humains pour visualiser l'état de circulation des routes. Le flux vidéo de ce réseau de caméras existant est livré non traité au centre de gestion du trafic, ainsi, il n'y a pas de stockage des scènes d'accident. De plus, il n'y a pas de technologie associée pour une gestion rapide en cas d'urgence.

Les objectifs de cette thèse sont d'abord l'identification de scénarios d'accidents et la collecte de données liées à un accident de la route; ensuite, la conception et le développement d'algorithmes de traitement vidéo pour la détection automatique des accidents sur les autoroutes. Les solutions développées utiliseront les caméras fixes existantes, afin de ne pas nécessiter d'investissements importants dans l'infrastructure. Les approches proposées sont basées sur l'utilisation de l'algorithme de flux optique et des calculs statistiques pour l'extraction de caractéristiques et la reconnaissance d'accidents. La solution proposée est suffisante et robuste au bruit et à la lumière.

Mots-clés : Vidéo surveillance, traitement d’images, traitement video, détection de comportement anormale, détections d’accidents routiers, flot optique.