

Article

Automatic Detection of Traffic Accidents from Video Using Deep Learning Techniques

Sergio Robles-Serrano ¹, German Sanchez-Torres ^{2,*}  and John Branch-Bedoya ¹ 

¹ Facultad de Minas, Universidad Nacional de Colombia, Sede Medellín 050041, Colombia; srobles@unal.edu.co (S.R.-S.); jwbranch@unal.edu.co (J.B.-B.)

² Facultad de Ingeniería, Universidad del Magdalena, Santa Marta 470001, Colombia

* Correspondence: gsanchez@unimagdalena.edu.co

Abstract: According to worldwide statistics, traffic accidents are the cause of a high percentage of violent deaths. The time taken to send the medical response to the accident site is largely affected by the human factor and correlates with survival probability. Due to this and the wide use of video surveillance and intelligent traffic systems, an automated traffic accident detection approach becomes desirable for computer vision researchers. Nowadays, Deep Learning (DL)-based approaches have shown high performance in computer vision tasks that involve a complex features relationship. Therefore, this work develops an automated DL-based method capable of detecting traffic accidents on video. The proposed method assumes that traffic accident events are described by visual features occurring through a temporal way. Therefore, a visual features extraction phase, followed by a temporary pattern identification, compose the model architecture. The visual and temporal features are learned in the training phase through convolution and recurrent layers using built-from-scratch and public datasets. An accuracy of 98% is achieved in the detection of accidents in public traffic accident datasets, showing a high capacity in detection independent of the road structure.

Keywords: urban traffic accident; deep learning; accident detection; recurrent neural networks; convolutional neural networks



Citation: Robles-Serrano, S.; Sanchez-Torres, G.; Branch-Bedoya, J. Automatic Detection of Traffic Accidents from Video Using Deep Learning Techniques. *Computers* **2021**, *10*, 148. <https://doi.org/10.3390/computers10110148>

Academic Editors: Paulo Quaresma, Vítor Nogueira and José Saias

Received: 15 September 2021

Accepted: 21 October 2021

Published: 9 November 2021

Publisher's Note: MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.



Copyright: © 2021 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction

There are different factors that cause traffic accidents. Among the most common factors that increase the probability of their occurrence are the geometry of the road [1], the climate of the area [2], drunk drivers, and speeding [3,4]. These accidents can cause harm to the people involved and, although most of these present only material damage, each one affects people's quality of life in terms of both traffic mobility and personal safety.

Thanks to technological advances, video cameras have become a resource for controlling and regulating traffic in urban areas. They make it possible to analyze and monitor the traffic flowing within the city [5]. However, the number of cameras needed to perform these tasks has been increasing significantly over time, which makes control difficult if automation mechanisms are not implemented because the number of professionals needed to comply with all the points also increases. Several approaches have been proposed to automate tasks within the control and follow-up process. An example of this is a system based on video camera surveillance in traffic. Through these, it is possible to estimate the speeds and trajectories of the objects of interest [6], with the objective of predicting and controlling the occurrence of traffic accidents in the area.

The scientific community has presented different approaches to detect traffic accidents [7]. These include statistics-based methods [8–10], social network data analysis [11,12], sensor data [13,14], machine learning, and deep learning [15–18]. These latest techniques have presented improvements in various fields of science, including video-based problem solving (video processing). Therefore, it is important to study these tech-

niques in order to approach a solution to the detection and classification of traffic accidents based on video.

With the advent of convolutional layers in the domain of neural networks, better performance has been achieved in the solution of problems involving digital image processing [19]. Deep learning techniques have shown high performance in a large number of problems, especially for image understanding and analysis [20,21]. These layers exploit the spatial relationship that the input data possess and that, due to the size of the information, it is not possible to achieve with dense neural networks [22]. The use of convolutions on input data with a large number of features makes it possible, among other things, to avoid the problem of the curse of dimensionality. This is a very frequent problem when working with data with high complexity, such as images. Likewise, it is important to highlight that the use of several convolutional layers helps the extraction of relevant visual features within the same dataset, which defines the performance of the network [23–25].

On the other hand, there are problems where the spatial relationship of the data is not a determining characteristic. In some problems, the temporal relationship that the data may have is of greater importance. This is because there are events that depend on past and/or future events, that is, on a context of the event in time in order to understand the real event. This is why a new deep learning model has emerged: recurrent neural networks. These networks have a similar architecture to dense artificial neural networks but differ in that at least one neuron has a connection to itself. This allows them to be able to remember what has been previously processed, i.e., it gives them the ability to store information over periods of time (data memory). They specialize in finding the temporal relationships that a set of data may have. Such networks are used to solve problems such as rate-of-change prediction [26], text translation [27], and natural language processing [28], among others. The data processing in these neurons has a higher complexity than the processing performed from a traditional neuron. In addition, these have been improved over the years. One of the most relevant changes was the possibility that the cell can store short and long term memory, called long short-term memory neurons (LSTM). These networks have presented improvements in several problems with respect to past models. Among these are travel time prediction problems [29], language understanding [30], and natural language processing [31].

However, the analysis of video scenes is not a problem that can be solved using one of the two models mentioned above. This is because a video presents both a spatial and a temporal relationship in its content. Therefore, the scientific community has presented several architectures that use both deep learning layers: convolutional layers and recurrent layers [32,33]. Some of the advances they have achieved using these types of architectures are emotion recognition [33], estimation of a person's posture [34], analysis of basketball videos for the automation of tasks such as the score of each team [35], and action recognition [36].

Because of this, a method capable of solving the traffic accident detection problem is proposed. However, the process of detecting traffic accidents is a task that involves a lot of processing and, for this reason, these tasks present many difficulties. The occurrence of a road accident is an event capable of occurring in multiple spatio-temporal combinations. This leaves a large domain of diverse distributions of data to be classified as an accident, which makes it difficult to solve the problem. Similarly, the classification of an accident is a complex problem due to the temporal implications it may present. Therefore, we seek to improve the performance of current approaches with the design of a method capable of detecting traffic accidents through video analysis using deep learning techniques.

The rest of the document is organized as follows: Section 2 describes the previous work carried out. Section 3 describes the proposed method to specify the architecture of the deep neural network proposed in the detection of traffic accidents. Section 4 shows the results and the experiments for the determination of the hyper-parameters of the proposed architecture. Section 5 establishes a discussion regarding bias, generalizability,

and some ethical considerations. Finally, Section 6 establishes the main conclusions, as well as future works.

2. Background

Different authors have proposed various techniques and methods for the detection or classification of accidents. Three study groups are presented based on the taxonomy of the solutions proposed in [7].

In the first group are those works that implement solutions with methods and algorithms based on the theory of vehicular flow and statistics. One of them uses the theory of vehicular flow with wireless communication between vehicles to give alerts on accident sites or road obstructions [37]. Other solutions use probability distributions, using the Poisson statistical distribution [10], based on the analysis of the relationships between heavy vehicle accidents, traffic, and road geometry [8], based on linear regression models with the Poisson distribution [38] to solve the problem.

The next group is based on methods related to machine learning and statistics [7,9,39,40]. Many of these have presented solutions using artificial neural networks [41–43], support vector machines [16,44], probabilistic neural networks [45], autoencoders [46], block clustering [47], Random Forest [15], pattern recognition [18], image processing techniques [48] and the Hidden Markov Model [49], among others. These approaches perform well, and are able to partially deal with an unbalanced dataset.

The last group is oriented to the amount of data collected by all the technology currently available in the community. Among them can be found data from social networks [12,17,50,51], data collected from sensors of a smartphone [13], structured data [52,53], data detected from video cameras [14,54,55], and traffic sensor data [56]. These datasets are widely used by deep learning, hybrid, and extreme learning methods. However, this group has the highest computational expense, which limits its use in various cases.

In terms of the video in which the accident is detected, there are several types. Among them are cameras inside the vehicle (first-person view), cameras located at intersections within cities, and cameras located on highways. In [57], an unsupervised model for the detection of accidents in videos obtained with first-person vision is proposed. In this, vehicles are detected and located on the scene by calculating a Bounding-Box for each one. Then the future position of the vehicle is predicted with the help of some calculated attributes to check if there is a collision between boxes. This approach can present false alarms with the auto-occlusion of objects of interest in the video. Using the same type of video with first-person vision, [40] presents a method for the detection of traffic accidents consisting of three steps: vehicle detection, tracking, and attribute extraction. A mixed Gaussian model is used for the detection of the moving vehicle and, by means of the mean displacement algorithm, the localized vehicle is tracked. Detection is determined by the repeated change of a vehicle's attributes: change in position, acceleration, and direction. By defining a threshold for the sum of the three attributes, it is possible to detect if an accident has occurred at the scene. However, the proposed model shows a deficiency in the detection of the vehicle in cases where the climatic conditions are highly variable.

In order to anticipate traffic accidents in videos with first-person vision, [58] proposes a model composed of spatial dynamic attention and a recurrent neural network. The first is in charge of learning to distribute a level of attention to the objects in the scene in each image of the video. This is in order to identify the objects of interest in the problem posed. The second, using short- and long-term memory cells, makes it possible to relate the signals that each object presents with the probability of an accident so that it is possible to anticipate the accident a few seconds in advance. In [49] a method for the detection of traffic incidents through video is described. The authors explain that surveillance cameras used at intersections present problems if their data are used to track a vehicle. This is because, for the most part, the cameras are located at angles where there is a large amount of occlusion of the objects present. For the detection of accidents, the authors propose an

algorithm based on the Hidden Markov Model. However, the method is limited to using a camera located at a high altitude, which limits the scalability of the implementation.

A matrix-based detection model is presented in [59]. First, the authors divide each image of the video into small windows. With this, they are able to calculate two matrices: velocity vector and velocity magnitude. Using the matrices and data collected from both cases (accident and non-accident), they manage to train a model that differentiates the movements of the sub-spaces in a video segment in order to classify the example as a common event or accident. From video recordings of surveillance cameras in [60], the authors describe an algorithm based on vehicle tracking for the detection of traffic accidents. The algorithm is made up of three parts: vehicle detection and extraction, moving vehicle characteristics extraction, and event detection. Four metrics are used as accident rates: acceleration, rate of change in position, rate of change in area, and rate of change in direction. By weighting these four variables, it is possible to detect whether or not an event occurred at the scene.

A different approach is presented in [61]. The authors show that it is possible to detect an accident between two or more vehicles from the sudden change in their speed vector by building a general model of the flow of movement to automatically detect unforeseen changes in the speed of a vehicle. However, the proposed method only shows high performance on the main highways where there is a constant movement and speed of vehicles. It constitutes a limitation in the domain of application due to the complex and different spatial configurations in which an accident can occur, where regular movement or constant speeds are not expected characteristics.

A particular approach for detecting traffic accidents is to consider this as an anomaly within the sequence of a video. Detecting anomalies in the real world has been one of the main problems in the field of computer vision. In [42], the authors present a model for detecting anomalies from a video. They perform a feature extraction using a neural network based on 3D convolution layers for each of the segments. These characteristics serve as input for a deep neural network model, which returns the anomaly score for each example. Finally, given the score obtained by each video segment, a binary classifier based on support vector machines is applied, which allows for distinguishing between the occurrence of an anomaly and a common event. In [46], a model is presented to detect traffic accidents through video using the autoencoder's deep learning architecture. Its framework is divided into two parts that run in parallel: object detection and anomaly detection. The first seeks to detect moving vehicles, track them, and calculate the intersection of the processed objects. The second seeks to exploit the information of the space-time video to later be represented by an autoencoder and obtain the anomaly score. Since accident detection is based on the definition of a threshold, changing the threshold value will also change the performance of the model. This is why the authors conducted tests with different threshold values.

In [43], a model based on two deep convolutional networks is proposed for the detection of accidents in traffic videos. The authors show that a video can be decomposed into two parts: a spatial component and a temporal component. The first is addressed to detect each vehicle on the scene together with its nearby region of accident probability, while the second is in charge of tracking the trajectory of each vehicle found in the video. An accident is detected when two objects collide.

In [62], the authors present a dataset of videos of traffic accidents, together with a predictive model of occurrence. The authors employed a modification to the Faster R-CNN architecture. The modification was made to the pooling layers by implementing context mining. Similarly, in [63], two models based on video analysis are proposed for the detection and classification of vehicles. The first uses a Gaussian method for background removal, plus a support vector machine as a classifier, while the second is based on an architecture named Faster RCNN, for the detection and classification of vehicles simultaneously.

In [56], a video-based model is proposed for the detection of traffic accidents. The dataset used was collected by the authors through the YouTube platform. These data are

made up of a total of 324 examples of training with six different types of collisions. Each category contains approximately 53 examples.

Recently, in [64], an automated anomaly detection technique based on deep learning in pedestrian walkways (DLADT-PW) was described. This approach aims to recognize and categorize the dissimilar anomalies present in pedestrian walkways. The analysis is based only on information per frame and does not consider exploiting the natural time component in the input that is used, which is the video.

3. Method for Automatic Detection of Traffic Accidents

The proposed method is based on techniques used in video analytics. In particular, deep learning neural networks architectures trained to detect the occurrence of a traffic accident are used. Before describing the architecture, it was necessary to define the network input. Since a video must be processed, it is separated into segments. Therefore, the temporal segmentation of the video required a basic analysis to determine which was the most appropriate scheme to generate the segments, considering a tradeoff between the computational cost of processing the segment and the generation of enough visual characteristics to extract patterns that the network learned. Once the input was defined, the accident event was built as the occurrence in time of a set of visual patterns. For this, the architecture has two parts. The first one extracts a vector of visual characteristics using a modified Inception V4 architecture; this set of characteristics is processed by a recurrent component to extract the temporal component associated with the occurrence of the event.

Next, we describe the two stages: temporal video segmentation and automatic detection of traffic accidents.

3.1. Temporal Video Segmentation

Temporal video segmentation is a problem that has been studied for many years by the scientific community since it is the first step towards the development of more general solutions, such as scene understanding of videos [65]. A video is a sequence of consecutive images with a particular order. When these images are viewed in the correct order and at a specific speed, it is possible to observe the animated event represented by the recorded video.

A video camera can capture, with the help of a mechanism, an event that is happening at the moment, in order to store, observe, and process it in the future. Using the same concept of a digital camera, a video camera makes it possible to capture a number of photographs per second, thus allowing the event that is occurring to be digitally recorded. These images, which represent the video, are known as frames. Video cameras allow recording at different numbers of frames per second (FPS). This means that the higher the number of FPS, the more fluid the movement of the objects on screen. The most commonly used FPS values are 30, 60, and 120 [66].

Video segmentation can be divided into two categories: spatial and temporal. Spatial segmentation seeks to visually classify objects of interest in the video in order to spatially locate objects in different frames. This type of segmentation is very useful when tracking objects in a video [67].

On the other hand, time segmentation seeks to solve a different problem. One of these is the reduction in the video time. This can be achieved by dividing the video into multiple fixed time windows in order to transform a long-duration video into a finite number of short-duration videos. However, there are also cases where it is not possible to divide the video into fixed time windows because it may contain multiple events in it, and, if a fixed window division is performed, it is possible that an event will be divided into different segments. In order to be able to discriminate between the multiple situations that the video describes, the scientific community has developed techniques to temporally segment this data, taking into account the scene change frame. This allows a single video to be divided into N temporal segments, where N is the number of scenes that can be observed.

Traffic accidents are rare events of short duration. In order to be able to detect them correctly through a video, it is important to preprocess the original recording considering that the video (using a static surveillance camera) will contain many frames with a high similarity index. For this reason, several temporal video segmentation techniques are proposed in order to compare them with the results of the detector model.

Three techniques were used to increase the variety of the data segments. The first is based on the metric named the Structural Similarity Index Measure, SSIM (Equation (1)), applied between consecutive frames in order to eliminate those that exceed an empirically defined threshold δ_{ss} . The threshold is arbitrarily set at a high value ($\delta_{ss} \geq 0.98$), representing a high similarity of the frames, which implies no additional information for the analysis. With this, we could significantly reduce the number of similar frames in the video segment.

$$SSIM(F_a, F_b) = \frac{(2\mu_{F_a}\mu_{F_b} + c_1)(2\sigma_{F_a F_b} + c_2)}{(\mu_{F_a}^2 + \mu_{F_b}^2 + c_1)(\sigma_{F_a}^2 + \sigma_{F_b}^2 + c_2)}, \quad (1)$$

where F_a and F_b are two consecutive frames from the video segment, and μ_{F_i} is the intensity mean of all the pixels from frame i . The $\sigma_{F_a F_b}$ factor is the covariance between the pixel value from frame a and b , and $\sigma_{F_i}^2$ is the variance from frame F_i . The $c_1 = (k_1 L)^2$ and $c_2 = (k_2 L)^2$. L is the dynamic range from pixel values ($2^{\#bit_per_pixel} - 1$), $k_1 = 0.01$ and $k_2 = 0.03$. The next technique used consists of a comparison very similar to the previous one. However, this one uses pixel-to-pixel comparison (Equations (2) and (3)) on consecutive frames, thus eliminating those that exceed the empirically defined threshold ($\delta_{pp} \geq 0.9$).

$$PtP(F_a, F_b) = \frac{1}{whk} \sum_{i=1}^w \sum_{j=1}^h \sum_{c=1}^k int(1 - |F_{aijc} - F_{bjc}|) \quad (2)$$

$$int(v) = \begin{cases} 1 & \text{if } v = 1 \\ 0 & \text{else} \end{cases} \quad (3)$$

where F_a and F_b are two consecutive frames of w rows, h columns, and k layers. Therefore, F_{qijc} is the pixel value from the q frame in the position (i, j) in layer c .

The last technique performs a selection using a fixed skip window. That is, a value is defined for K , which represents the number of frames that must be eliminated before selecting the next candidate to form the segment. If K is equal to 1, it means that the frames to be selected should be those with odd indices (1, 3, 5, 7, 9, ... , etc.). All this continues until the maximum segment number is reached, for which the following values were defined: 10, 15, 30, 45, and 60.

3.2. Automatic Detection of Traffic Accidents

In order to interpret a video segment to detect whether an event occurs, the data must be exploited in two main ways: visually and temporally.

The convolutional-based architectures [68] are the most important techniques for visual analysis of images. These are a significant improvement over traditional artificial neural networks in the performance of image classification solutions. However, convolutional layers do not solve all problems. One of the weaknesses of convolutional layers is that they are not good at extracting temporal features from data. Although convolutional layers are powerful in exploiting the spatial characteristics of the data, recurrent neural networks were designed to exploit the temporal characteristics of the data. Convolutional layers are able to process the data in such a way that the spatial information changes to a more abstract representation saving computational cost. Currently, these architectures are used as automatic extractors of image features due to their performance reducing the dimensionality of the input data. However, spatial data is not everything in a video.

Sequential data is of importance in understanding an event that happens over a time span. Recurrent neural networks perform better when processing a sequence over time

compared to feed-forward artificial neural networks. There are solutions that use both architectures in order to improve performance in solving video comprehension problems [69]. However, the scientific community has presented a design capable of exploiting both types of data: the Convolutional LSTM (ConvLSTM) layers. These are a special type of architecture where the cells follow the same operations as a Long Short-Term Memory neuron but differ in that the input operations are convolutions instead of basic arithmetic operations. This architecture has shown high performance in problems with video compression.

To solve the traffic accident detection problem, the first part of the architecture is designed as an automatic image feature extractor to process each frame of the video segment. Then, this new representation of the data is used as input data in an empirically designed recurrent neural network to extract temporal information from the input data. Finally, a dense artificial neural network block is used to perform the binary classification of detecting an accident, as shown in Figure 1.

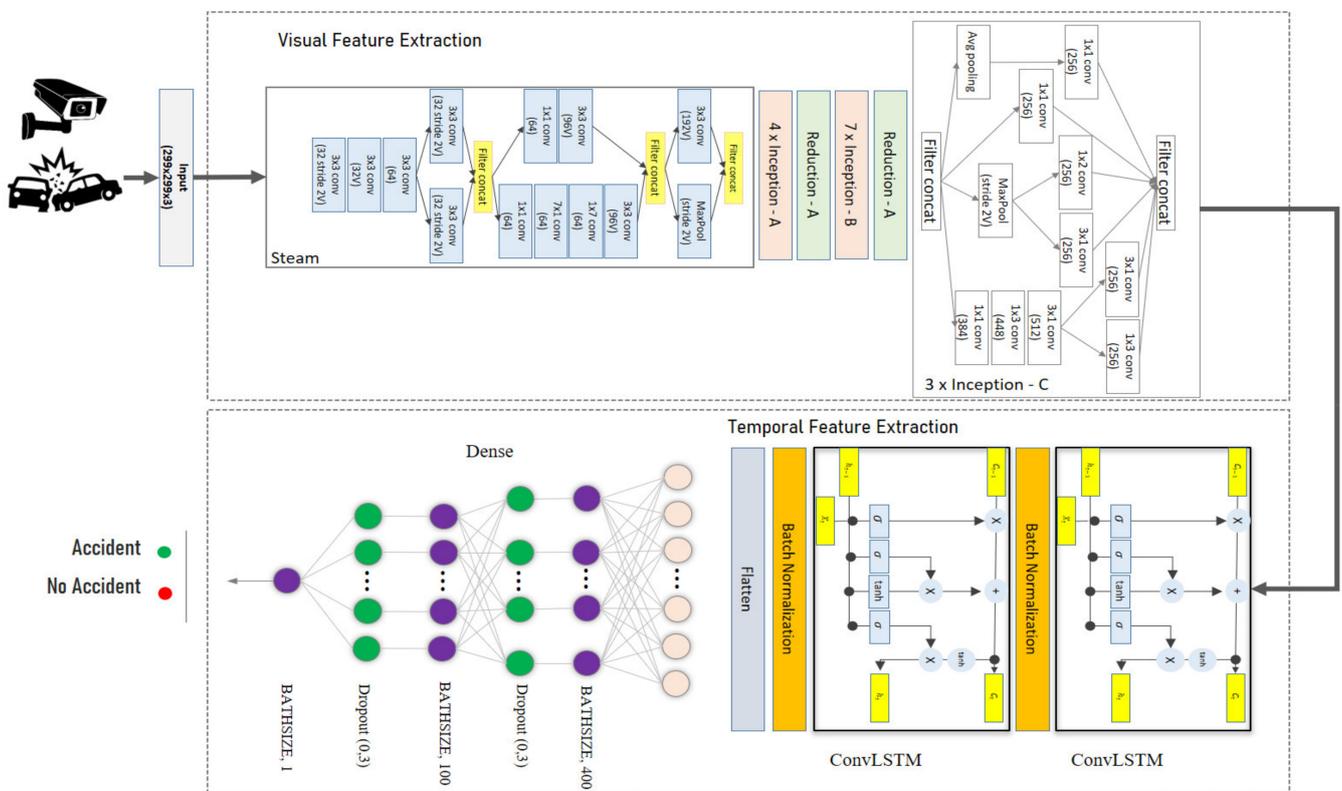


Figure 1. Architecture for video analysis, visual feature extractor based on the InceptionV4 architecture (**top**) and temporal feature extractor (**bottom**).

The proposed model consists of three parts: a spatial feature extractor, a temporal feature extractor, and a binary classifier. The first part uses an architecture named InceptionV4, proposed in [70]. This model was trained with the ImageNet dataset, which showed high performance in solving this problem by classifying images among a thousand different categories. However, this pre-trained model does not show good results when detecting a traffic accident in images because the model was trained with a completely different task. Therefore, when applying transfer learning, it seeks to compensate for the acquisition of knowledge in a new task. For this reason, an adjustment is made to the model using a new dataset with examples of traffic accidents in images.

To obtain more information from video type data, it is necessary to know the temporal and the spatial features; therefore, it is required to use a model capable of extracting these kinds of characteristics. Therefore, a ConvLSTM layer-based neural network archi-

texture is proposed that receives as input the feature vector computed by the adjusted InceptionV4 architecture.

Finally, it is necessary to detect whether the video segment contains a traffic accident. For this, a dense artificial neural network block using regularization methods is proposed so that the final model is able to generalize the solution.

4. Results

4.1. Dataset

For the solution proposed, two sets of data were used. The first one consists of images used for the fine tuning of the visual feature vector extractor. The second one consists of videos that present traffic accidents (positive and negative cases) for training the temporal feature extractor.

The image dataset was built from scratch, applying the web scraping technique to populate the dataset. For this, a series of logical steps were proposed. First, we identified the sources on the web where the image search was performed. Next, we defined the set of keywords for the searches. For this process, the following keywords were selected: Traffic accidents, Car accidents, Motorcycle accidents, and Truck accidents. Then, the automation stage was performed. The application was developed in the Python programming language together with the Selenium library, which contains useful functions to perform this process. Finally, a manual validation of all the collected images was carried out together with an image transformation in order to standardize the size and format used.

The videos dataset was formed from two different data sources. The first one is the CADP dataset [62]. It has a total of 1416 traffic accident videos, i.e., positive examples of the traffic accident problem. This dataset adds up to a total duration of 5.2 h, with an average number of frames of 366. This source was chosen instead of others in the literature, such as [42,58], due to the number of positive cases that the CADP dataset presents (100%) and the position of the video camera (CCTV), which allows for a third-person perspective. The second source used for the video dataset only contains negative cases of the presented problem, i.e., videos where no traffic accidents are presented [71]. It has a total of 100 videos from different locations in China, with a spatial resolution of 960×540 pixels. Some examples of frames belonging to these datasets are shown in Figure 2.

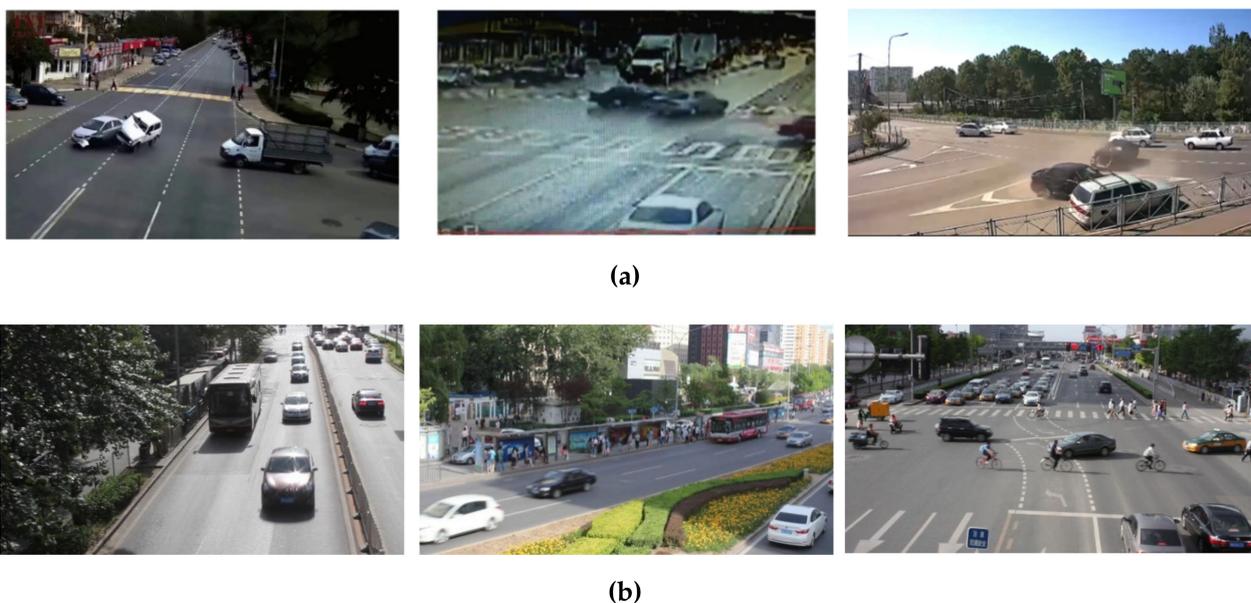


Figure 2. Examples of frames from videos in the datasets: (a) frames from positive class (accidents), (b) frames from negative class (no accident).

4.2. Temporal Video Segmentation

A video is segmented in order to obtain a greater number of examples with a certain number of constant frames and, in turn, a segment with shorter duration. This is because traffic accidents have a short average duration (10 frames) [58], which allows for processing of the original video in a more efficient way.

In order to select the segmentation technique for the input data, some experiments were performed on the videos taken from the dataset. The four techniques to be evaluated were compared using the same videos in each case. The first technique consists of a segmentation without frame discrimination. Therefore, all consecutive images of the video are selected until the maximum time of the segment is reached. This technique has an average reading time of 0.18 s. The second technique used seeks to skip frames in order to reduce the redundancy that can be observed when using very close images in the video. This is because when the video has been recorded with a traditional camera, the number of similar consecutive frames is very high. For this reason, we experimented by skipping one frame for each frame selected. That is, in this case, the images with an odd index were chosen from the video, until the maximum length of duration established for the segment was reached. The third and fourth techniques presented are based on discriminating consecutive frames with respect to an SSIM. For the third technique, a pixel-to-pixel comparison of two consecutive images is calculated. For decision making, a threshold of 0.9 was set. Therefore, if a consecutive frame exceeds this threshold, the candidate is not chosen and moves to the next frame in the video, for which the same process is performed. Finally, the fourth technique number four shows a similar process to the third technique. However, in this one, the threshold was defined at 0.98, and the matching operation used is the SSIM image-matching metric [72]. A maximum segment length of 45 frames was set for the tests. The comparison between techniques is presented in Tables 1 and 2. The technique chosen was the first described: “No selection”.

Table 1. Comparison between segment generation techniques.

Method	Advantages	Disadvantages
No selection	Low runtime, no data loss	High similarities between adjacent frames
Skip frame ($n = 1$)	Low runtime, with medium/high similarity in adjacent frames	Possible data loss
Pixel similarity	Low runtime, with medium/high similarity in adjacent frames	Possible data loss
Structural similarity	Low similarity in adjacent frames	High execution time

Table 2. Results between segment generation techniques.

Method	Frames	Execution Time ¹	Similarity
No selection	45	0.918	0.824
Skip frame ($n = 1$)	45	0.971	0.761
Pixel similarity	45	1.213	0.874
Structural similarity	45	2.456	0.822

¹ Average value measured in seconds.

4.3. Automatic Detection of Traffic Accidents

The solution presented is based on a visual and a temporal feature extractor. The first stage of the model consists of the InceptionV4 architecture (pre-trained with the ImageNet dataset) [70] truncated. That is, all the Inception cells (convolutional layers) were used, eliminating the multilayer perceptron at the end of this architecture. This is to use this part of the model only as a visual feature extractor as in Figure 1, upper part.

However, by performing multiple experiments, it was concluded that the pre-trained model does not differentiate between a vehicle at rest and a vehicle hit by a traffic accident. Therefore, the images dataset was used for training in order to adjust the weights of this pre-trained network. In this process, all the weights of the initial layers of the architecture were frozen, and only those of the last convolutional cell of InceptionV4 were adjusted. To adjust the feature extractor, multiple experiments were performed. This was done using

regularization techniques, data augmentation, and hyper-parameter modifications. The results of the tests performed are described in Figure 3.

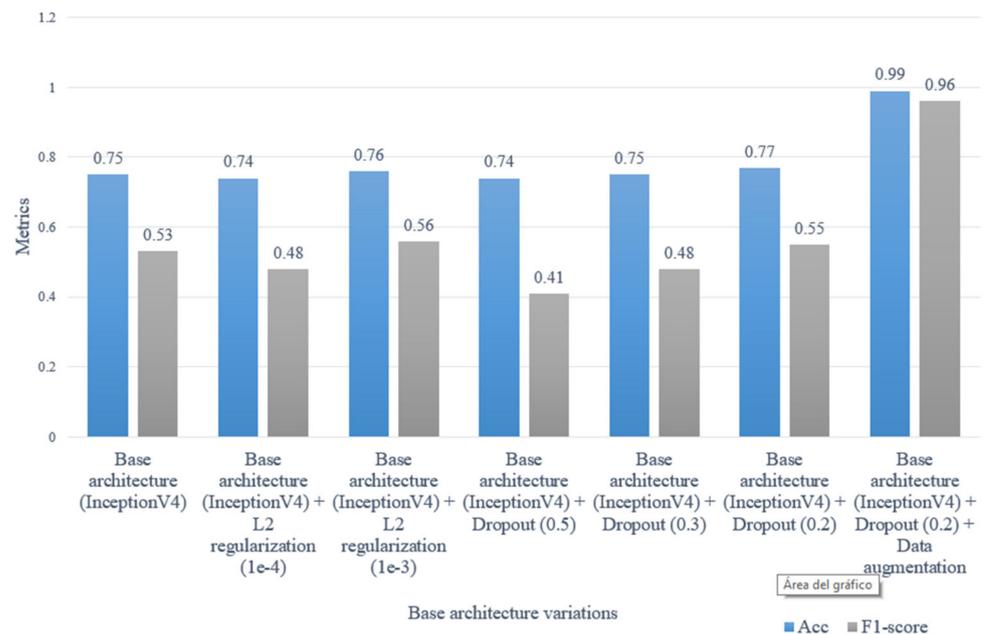


Figure 3. Visual feature extractor experiment.

The temporal feature extraction is based on recurrent neural networks. The architecture proposed for this stage consists of two ConvLSTM layers. These were created to extract temporal information in data of more than one dimension, using the convolution operation. Between these layers, a BatchNormalization is added, and the various hyper-parameters are adjusted. The ConvLSTM layers used consist of 64 neurons each, a kernel size of 3×3 , a dropout of 0.2 and a recurrent dropout of 0.1. The results obtained are presented in Figure 4, while Figure 5 shows the accuracy of the model in the training stage.

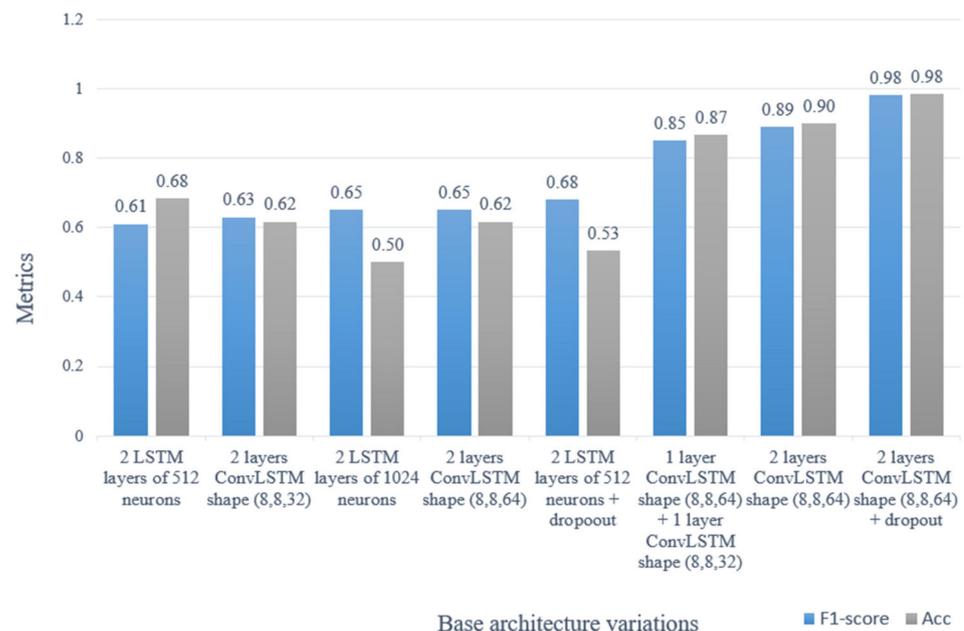


Figure 4. Experimenting with the temporal feature extractor.

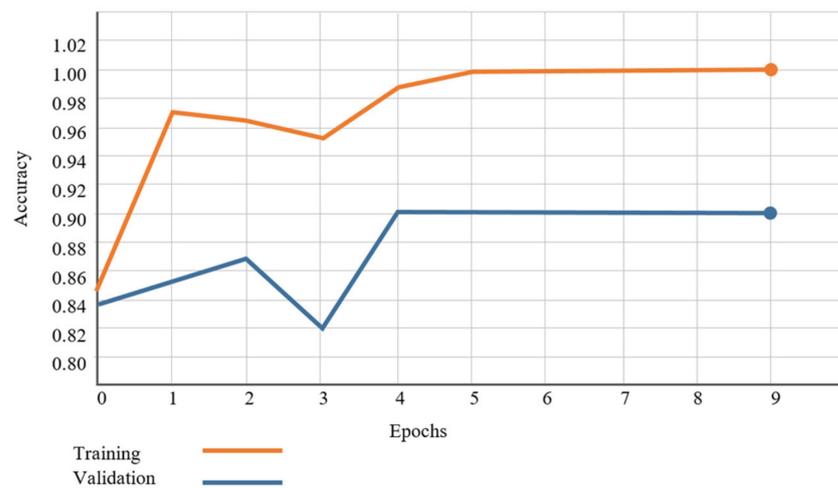


Figure 5. Behavior of the model's accuracy by epochs with the training set and the validation set.

The last stage of the accident detection process is given by a densely layered block. The proposed neural network consists of a total of three hidden layers and one output layer, plus a regularization technique called dropout with a value of 0.3. The distribution of the neurons in the mentioned layers is as follows: four hundred, one hundred, and one neurons, where the first two layers use the hyperbolic tangent activation function while the last layer (output layer) uses the sigmoid activation function in order to perform a binary classification (accident or non-accident). The training and validation results are presented in Table 3 (Note: the dataset is distributed as 54% accident and 46% non-accident). The established hyper-parameter values are presented in Table 4. The model was trained on a computer with a 5th generation I7 4820k@3.70GHz processor, 64 GB of RAM memory, and two Nvidia 1080TI video cards with 11 GB of GDDR 5X RAM at 405 MHz.

Table 3. Confusion matrix of the complete model on the validation set.

		Predicted Class	
		No Accident	Accident
Actual class	No accident	0.99	0.01
	Accident	0.03	0.97

Table 4. Hyper-parameters of the proposed neural network.

Hyper-Parameter	Value
Input size	45 frames
Batch size	4
Loss function	Binary cross-entropy
Optimizer	Adam optimization
Weight initialization	Xavier initialization
Learning rate	0.0001
Number of epochs	10

5. Discussion

In relation to model bias, the model could be biased towards accidents involving vehicles. This verification is not trivial due to limitations of the validation dataset, which is composed of a majority of accidents involving vehicles. However, the feature extractor was trained considering different types of vehicles, including cars, trucks, and motorcycles but excluding pedestrians and cyclists where there are visible human interventions. Additionally, relating to weather conditions, the bias is present in diurnal accidents due to

dataset limitations. There were not enough videos with rain or snow or at night, among other weather conditions.

Regarding the model's generalization capacity, the model is independent of a particular camera-viewpoint, the structure of the street, or aspects such as vehicle density. We do not describe the technical parameters of the cameras because we used public video datasets for the model validation. However, an adequate analysis in this address will permit defining hardware limitations for a correct model operation. However, it is difficult to perform analysis at the level of device specification, mainly because obtaining a dataset that includes a large number of images from cameras with different lenses, acquisition sensors, and even spatial positions is impractical. In this context, separating or analyzing the effect of the camera parameters is almost impossible [73]. However, Deep Learning models have the characteristic of being robust to small variations in their input. They require minimal preprocessing and do not need the selection of an extractor of specific characteristics [74]. In this context, we consider that there are no significant camera parameters restrictions in the model due to the used datasets that include different cameras in multiple positions. Therefore, we assume that the model can operate correctly in the most popular devices used for vehicular traffic systems.

The feature extractor was trained with visual patterns associated with accidents that had already occurred, so the model cannot predict an accident, but it is capable of identifying visual patterns relating to the occurrence of an accident. The temporal feature extractor was trained to recognize the appearance in time of these visual patterns, which strengthens the accident identification only when based on visual patterns. We considered that it is not possible to predict in advance the occurrence of accidents with this configuration.

Addressing ethical considerations, the group of accidents involving pedestrians was not considered because the nature of the training in the fine-tuning process of the feature extractor model required images that included injured bodies. To include this category (pedestrians), consideration should be given to obtaining a representative group of images to avoid biases due to aspects such as age, height, or skin color.

6. Conclusions and Future Work

Pre-trained neural networks are not able to compute a vector with relevant features for very specific problems. Therefore, it is necessary to adjust the weights of these models using examples of the problem to be solved.

The technique that best represents a temporal segment of a traffic accident does not eliminate any data, because the similarity values between the segments of the techniques with frame selection present negligible differences between them, while the computational cost, processing time and accuracy in accident detection present better results by not conditioning the selection of frames to a metric.

Artificial vision has made great advances in the understanding of video scenes. One of the best-performing techniques is artificial neural networks. Many of these models are based on architectures composed of convolutional layers and recurrent layers, in order to extract as much information as possible from the input data. The proposed method is based on this type of architecture and achieves a high performance when detecting traffic accidents in videos, achieving an F1 score of 0.98 and an accuracy of 98%.

The proposed model shows high performance for video traffic accident detection. However, due to the paucity of such datasets in the scientific community, the conditions under which the model works are limited. The solution is restricted to vehicular collisions, excluding motorcycles, bicycles, and pedestrians due to the negligible number of these types of examples available. In addition, the model has errors in determining accident segments with low illumination (such as nighttime videos) or low resolution and occlusion (low quality video cameras and locations).

Author Contributions: Conceptualization, S.R.-S. and G.S.-T.; methodology, S.R.-S., G.S.-T. and J.B.-B.; software, S.R.-S.; validation, G.S.-T., J.B.-B. and S.R.-S.; formal analysis, S.R.-S.; investigation, S.R.-S.; resources, S.R.-S.; data curation, S.R.-S. and G.S.-T.; writing—original draft preparation, S.R.-S. and G.S.-T.; writing—review and editing, J.B.-B.; visualization, G.S.-T.; supervision, J.B.-B. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by Universidad Nacional de Colombia, sede Medellín, grant number BEDA2019I.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Authors can confirm that all relevant data are included in the article.

Conflicts of Interest: The authors declare no conflict of interest.

References

- Li, M.Z. The Road Traffic Analysis Based on an Urban Traffic Model of the Circular Working Field. *Acta Math. Appl. Sin.* **2004**, *20*, 77–84. [[CrossRef](#)]
- Chu, W.; Wu, C.; Atombo, C.; Zhang, H.; Özkan, T. Traffic Climate, Driver Behaviour, and Accidents Involvement in China. *Accid. Anal. Prev.* **2019**, *122*, 119–126. [[CrossRef](#)]
- Guimarães, A.G.; da Silva, A.R. Impact of Regulations to Control Alcohol Consumption by Drivers: An Assessment of Reduction in Fatal Traffic Accident Numbers in the Federal District, Brazil. *Accid. Anal. Prev.* **2019**, *127*, 110–117. [[CrossRef](#)] [[PubMed](#)]
- Nishitani, Y. Alcohol and Traffic Accidents in Japan. *IATSS Res.* **2019**, *43*, 79–83. [[CrossRef](#)]
- Mahata, D.; Narzary, P.K.; Govil, D. Spatio-Temporal Analysis of Road Traffic Accidents in Indian Large Cities. *Clin. Epidemiol. Glob. Health* **2019**, *7*, 586–591. [[CrossRef](#)]
- Sheng, H.; Zhao, H.; Huang, J.; Li, N. A Spatio-Velocity Model Based Semantic Event Detection Algorithm for Traffic Surveillance Video. *Sci. China Technol. Sci.* **2010**, *53*, 120–125. [[CrossRef](#)]
- Parsa, A.B.; Chauhan, R.S.; Taghipour, H.; Derrible, S.; Mohammadian, A. Applying Deep Learning to Detect Traffic Accidents in Real Time Using Spatiotemporal Sequential Data. *arXiv* **2019**, arXiv:1912.06991.
- Joshua, S.C.; Garber, N.J. Estimating Truck Accident Rate and Involvements Using Linear and Poisson Regression Models. *Transp. Plan. Technol.* **1990**, *15*, 41–58. [[CrossRef](#)]
- Arvin, R.; Kamrani, M.; Khattak, A.J. How Instantaneous Driving Behavior Contributes to Crashes at Intersections: Extracting Useful Information from Connected Vehicle Message Data. *Accid. Anal. Prev.* **2019**, *127*, 118–133. [[CrossRef](#)] [[PubMed](#)]
- Jovanis, P.P.; Chang, H.L. Modeling the Relationship of Accidents To Miles Traveled. *Transp. Res. Rec.* **1986**, 42–51.
- Xu, S.; Li, S.; Wen, R. Sensing and Detecting Traffic Events Using Geosocial Media Data: A Review. *Comput. Environ. Urban Syst.* **2018**, *72*, 146–160. [[CrossRef](#)]
- Gu, Y.; Qian, Z.; Chen, F. From Twitter to Detector: Real-Time Traffic Incident Detection Using Social Media Data. *Transp. Res. Part C Emerg. Technol.* **2016**, *67*, 321–342. [[CrossRef](#)]
- Fernandes, B.; Alam, M.; Gomes, V.; Ferreira, J.; Oliveira, A. Automatic Accident Detection with Multi-Modal Alert System Implementation for ITS. *Veh. Commun.* **2016**, *3*, 1–11. [[CrossRef](#)]
- Maha, V.C.; Rajalakshmi, M.; Nedunchezian, R. Intelligent Traffic Video Surveillance and Accident Detection System with Dynamic Traffic Signal Control. *Clust. Comput.* **2018**, *21*, 135–147. [[CrossRef](#)]
- Ozbayoglu, M.; Kucukayan, G.; Dogdu, E. A real-time autonomous highway accident detection model based on big data processing and computational intelligence. In Proceedings of the 2016 IEEE International Conference on Big Data, Big Data, Washington, DC, USA, 5–8 December 2016; pp. 1807–1813. [[CrossRef](#)]
- Dong, N.; Huang, H.; Zheng, L. Support Vector Machine in Crash Prediction at the Level of Traffic Analysis Zones: Assessing the Spatial Proximity Effects. *Accid. Anal. Prev.* **2015**, *82*, 192–198. [[CrossRef](#)] [[PubMed](#)]
- Zhang, Z.; He, Q.; Gao, J.; Ni, M. A Deep Learning Approach for Detecting Traffic Accidents from Social Media Data. *Transp. Res. Part C Emerg. Technol.* **2018**, *86*, 580–596. [[CrossRef](#)]
- Yu, R.; Wang, G.X.; Zheng, J.Y.; Wang, H.Y. Urban Road Traffic Condition Pattern Recognition Based on Support Vector Machine. *Jiaotong Yunshu Xitong Gongcheng Yu Xinxi/J. Transp. Syst. Eng. Inf. Technol.* **2013**, *13*, 130–136. [[CrossRef](#)]
- Albawi, S.; Mohammed, T.A.M.; Alzawi, S. Understanding of a convolutional neural network. In Proceedings of the 2017 International Conference on Engineering and Technology (ICET), Antalya, Turkey, 21–23 August 2017; pp. 1–6.
- Chan, T.H.; Jia, K.; Gao, S.; Lu, J.; Zeng, Z.; Ma, Y. PCANet: A Simple Deep Learning Baseline for Image Classification? *IEEE Trans. Image Process.* **2015**, *24*, 5017–5032. [[CrossRef](#)]
- Wu, J.; Yu, Y.; Huang, C.; Yu, K. Deep Multiple Instance Learning for Image Classification and Auto-Annotation. *J. Reconstr. Microsurg.* **1985**, *1*, 287–289. [[CrossRef](#)]
- Rawat, W.; Wang, Z. Deep Convolutional Neural Networks for Image Classification: A Comprehensive Review. *MIT Press J.* **2018**, 2733, 2709–2733. [[CrossRef](#)]

23. Howard, A.G. Some improvements on deep convolutional neural network based image classification. In Proceedings of the 2nd International Conference on Learning Representations (ICLR 2014), Banff, AB, Canada, 14–16 April 2014.
24. Krizhevsky, A.; Sutskever, I.; Hinton, G.E. ImageNet Classification with Deep Convolutional Neural Networks. *Commun. ACM* **2017**, *60*, 84–90. [[CrossRef](#)]
25. Brinker, T.J.; Hekler, A.; Enk, A.H.; Klode, J.; Hauschild, A.; Berking, C.; Schilling, B.; Haferkamp, S.; Schadendorf, D.; Fröhling, S.; et al. A Convolutional Neural Network Trained with Dermoscopic Images Performed on Par with 145 Dermatologists in a Clinical Melanoma Image Classification Task. *Eur. J. Cancer* **2019**, *111*, 148–154. [[CrossRef](#)]
26. Panda, C.; Narasimhan, V. Forecasting Exchange Rate Better with Artificial Neural Network. *J. Policy Modeling* **2007**, *29*, 227–236. [[CrossRef](#)]
27. Song, H.J.; Kim, A.Y.; Park, S.B. Translation of natural language query into keyword query using a Rnn encoder-decoder. In Proceedings of the SIGIR 2017—40th International ACM SIGIR Conference on Research and Development in Information Retrieval, Tokyo, Japan, 7–11 August 2017; pp. 965–968. [[CrossRef](#)]
28. Kahuttanaseth, W.; Dressler, A.; Netramai, C. Commanding mobile robot movement based on natural language processing with RNN encoder-decoder. In Proceedings of the 2018 5th International Conference on Business and Industrial Research: Smart Technology for Next Generation of Information, Engineering, Business and Social Science, ICBIR 2018, Bangkok, Thailand, 17–18 May 2018; pp. 161–166. [[CrossRef](#)]
29. Duan, Y.; Lv, Y.; Wang, F.Y. Travel time prediction with LSTM neural network. In Proceedings of the IEEE Conference on Intelligent Transportation Systems, ITSC 2016, Rio de Janeiro, Brazil, 1–4 November 2016; pp. 1053–1058. [[CrossRef](#)]
30. Sundermeyer, M.; Ney, H.; Schluter, R. From Feedforward to Recurrent LSTM Neural Networks for Language Modeling. *IEEE Trans. Audio Speech Lang. Process.* **2015**, *23*, 517–529. [[CrossRef](#)]
31. Venugopalan, S.; Xu, H.; Donahue, J.; Rohrbach, M.; Mooney, R.; Saenko, K. Translating videos to natural language using deep recurrent neural networks. In Proceedings of the 2015 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (NAACL HLT 2015), Denver, CO, USA, 31 May–5 June 2015; pp. 1494–1504. [[CrossRef](#)]
32. Ullah, A.; Ahmad, J.; Muhammad, K.; Sajjad, M.; Baik, S.W. Action Recognition in Video Sequences Using Deep Bi-Directional LSTM with CNN Features. *IEEE Access* **2017**, *6*, 1155–1166. [[CrossRef](#)]
33. Fan, Y.; Lu, X.; Li, D.; Liu, Y. Video-based emotion recognition using CNN-RNN and C3D hybrid networks. In Proceedings of the ICMI 2016—18th ACM International Conference on Multimodal Interaction, Tokyo, Japan, 12–16 November 2016; pp. 445–450. [[CrossRef](#)]
34. Nishani, E.; Cico, B. Computer vision approaches based on deep learning and neural networks: Deep neural networks for video analysis of human pose estimation. In Proceedings of the 2017 6th Mediterranean Conference on Embedded Computing (MECO), Bar, Montenegro, 11–15 June 2017; pp. 11–14.
35. Liu, W.; Yan, C.C.; Liu, J.; Ma, H. Deep Learning Based Basketball Video Analysis for Intelligent Arena Application. *Multimed. Tools Appl.* **2017**, *76*, 24983–25001. [[CrossRef](#)]
36. Lee, I.; Kim, D.; Kang, S.; Lee, S. Ensemble deep learning for skeleton-based action recognition using temporal sliding LSTM networks. In Proceedings of the 2017 IEEE International Conference on Computer Vision, Venice, Italy, 22–29 October 2017; pp. 1012–1020. [[CrossRef](#)]
37. Chen, A.; Khorashadi, B.; Chuah, C.N.; Ghosal, D.; Zhang, M. Smoothing vehicular traffic flow using vehicular-based ad hoc networking & Computing grid (VGrid). In Proceedings of the IEEE Conference on Intelligent Transportation Systems, ITSC 2006, Toronto, ON, Canada, 17–20 September 2006; pp. 349–354. [[CrossRef](#)]
38. Miaou, S.P.; Lum, H. Modeling Vehicle Accidents and Highway Geometric Design Relationships. *Accid. Anal. Prev.* **1993**, *25*, 689–709. [[CrossRef](#)]
39. Parsa, A.B.; Movahedi, A.; Taghipour, H.; Derrible, S.; Mohammadian, A. (Kouros) Toward Safer Highways, Application of XGBoost and SHAP for Real-Time Accident Detection and Feature Analysis. *Accid. Anal. Prev.* **2020**, *136*, 105405. [[CrossRef](#)]
40. Hui, Z.; Xie, Y.; Lu, M.; Fu, J. Vision-based real-time traffic accident detection. In Proceedings of the World Congress on Intelligent Control and Automation (WCICA), Shenyang, China, 29 June–4 July 2014; pp. 1035–1038. [[CrossRef](#)]
41. Motamed, M. Developing A Real-Time Freeway Incident Detection Model Using Machine Learning Techniques. Ph.D. Thesis, The University of Texas at Austin, Austin, TX, USA, 2016.
42. Sultani, W.; Chen, C.; Shah, M. Real-world anomaly detection in surveillance videos. In Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18–22 June 2018; pp. 6479–6488. [[CrossRef](#)]
43. Huang, X.; He, P.; Rangarajan, A.; Ranka, S. Intelligent Intersection: Two-Stream Convolutional Networks for Real-Time near Accident Detection in Traffic Video. *arXiv* **2019**, arXiv:1901.01138. [[CrossRef](#)]
44. Mokhtarimousavi, S.; Anderson, J.C.; Azizinamini, A.; Hadi, M. Improved Support Vector Machine Models for Work Zone Crash Injury Severity Prediction and Analysis. *Transp. Res. Rec.* **2019**, *2673*, 680–692. [[CrossRef](#)]
45. Parsa, A.B.; Taghipour, H.; Derrible, S.; Mohammadian, A. (Kouros). Real-Time Accident Detection: Coping with Imbalanced Data. *Accid. Anal. Prev.* **2019**, *129*, 202–210. [[CrossRef](#)]
46. Singh, D.; Mohan, C.K. Deep Spatio-Temporal Representation for Detection of Road Accidents Using Stacked Autoencoder. *IEEE Trans. Intell. Transp. Syst.* **2019**, *20*, 879–887. [[CrossRef](#)]

47. Rahimi, A.; Azimi, G.; Asgari, H.; Jin, X. Clustering Approach toward Large Truck Crash Analysis. *Transp. Res. Rec.* **2019**, *2673*, 73–85. [[CrossRef](#)]
48. Marimuthu, R.; Suresh, A.; Alamelu, M.; Kanagaraj, S. Driver Fatigue Detection Using Image Processing and Accident. *Int. J. Pure Appl. Math.* **2017**, *116*, 91–99. [[CrossRef](#)]
49. Zou, Y.; Shi, G.; Shi, H.; Wang, Y. Image sequences based traffic incident detection for signaled intersections using HMM. In Proceedings of the 2009 9th International Conference on Hybrid Intelligent Systems, HIS 2009, Shenyang, China, 12–14 August 2009; Volume 1, pp. 257–261. [[CrossRef](#)]
50. Gutiérrez, C.; Figueiras, P.; Oliveira, P.; Costa, R.; Jardim-Goncalves, R. An Approach for Detecting Traffic Events Using Social Media. *Stud. Comput. Intell.* **2016**, *647*, 61–81. [[CrossRef](#)]
51. Ghandour, A.J.; Hammoud, H.; Dimassi, M.; Krayem, H.; Haydar, J.; Issa, A. Allometric Scaling of Road Accidents Using Social Media Crowd-Sourced Data. *Phys. A Stat. Mech. Appl.* **2020**, *545*. [[CrossRef](#)]
52. Weil, R.; Wootton, J.; García-Ortiz, A. Traffic Incident Detection: Sensors and Algorithms. *Math. Comput. Model.* **1998**, *27*, 257–291. [[CrossRef](#)]
53. Xiao, J. SVM and KNN Ensemble Learning for Traffic Incident Detection. *Phys. A Stat. Mech. Appl.* **2019**, *517*, 29–35. [[CrossRef](#)]
54. Liu, X.M.; Zhang, Z.H.; Li, G.Y.; Lv, T.J. Research on Technology of Traffic Video Incidents Detection under Highway Condition. *J. China Univ. Posts Telecommun.* **2010**, *17*, 79–83. [[CrossRef](#)]
55. Zhang, Y.; Zhu, Y. A novel storing and accessing method of traffic incident video based on spatial-temporal analysis. In Proceedings of the International Conference on Algorithms and Architectures for Parallel Processing, Zhangjiajie, China, 18–20 November 2015; Volume 9529, pp. 315–329. [[CrossRef](#)]
56. Chen, Y.; Yu, Y.; Li, T. A vision based traffic accident detection method using extreme learning machine. In Proceedings of the ICARM 2016—2016 International Conference on Advanced Robotics and Mechatronics, Macau, China, 18–20 August 2016; pp. 567–572. [[CrossRef](#)]
57. Yao, Y.; Xu, M.; Wang, Y.; Crandall, D.J.; Atkins, E.M. Unsupervised traffic accident detection in first-person videos. In Proceedings of the 2019 IEEE/RISJ International Conference on Intelligent Robots and Systems (IROS), Macau, China, 3–8 November 2019.
58. Chan, F.H.; Chen, Y.T.; Xiang, Y.; Sun, M. Anticipating accidents in dashcam videos. In Proceedings of the 13th Asian Conference on Computer Vision, Taipei, Taiwan, 20–24 November 2016; Volume 10114, pp. 136–153. [[CrossRef](#)]
59. Xia, S.; Xiong, J.; Liu, Y.; Li, G. Vision-based traffic accident detection using matrix approximation. In Proceedings of the 2015 10th Asian Control Conference: Emerging Control Techniques for a Sustainable World, ASCC 2015, Kota Kinabalu, Malaysia, 31 May–3 June 2015. [[CrossRef](#)]
60. Ki, Y.K.; Lee, D.Y. A Traffic Accident Recording and Reporting Model at Intersections. *IEEE Trans. Intell. Transp. Syst.* **2007**, *8*, 188–194. [[CrossRef](#)]
61. Maaloul, B.; Taleb-Ahmed, A.; Niar, S.; Harb, N.; Valderrama, C. Adaptive video-based algorithm for accident detection on highways. In Proceedings of the 2017 12th IEEE International Symposium on Industrial Embedded Systems, SIES 2017, Toulouse, France, 14–16 June 2017. [[CrossRef](#)]
62. Shah, A.P.; Lamare, J.B.; Nguyen-Anh, T.; Hauptmann, A. CADP: A novel dataset for CCTV traffic camera based accident analysis. In Proceedings of the AVSS 2018—2018 15th IEEE International Conference on Advanced Video and Signal-Based Surveillance, Auckland, New Zealand, 27–30 November 2019. [[CrossRef](#)]
63. Arinaldi, A.; Pradana, J.A.; Gurusanga, A.A. Detection and Classification of Vehicles for Traffic Video Analytics. *Procedia Comput. Sci.* **2018**, *144*, 259–268. [[CrossRef](#)]
64. Pustokhina, I.V.; Pustokhin, D.A.; Vaiyapuri, T.; Gupta, D.; Kumar, S.; Shankar, K. An Automated Deep Learning Based Anomaly Detection in Pedestrian Walkways for Vulnerable Road Users Safety. *Saf. Sci.* **2021**, *142*, 105356. [[CrossRef](#)]
65. Jiang, H.; Zhang, G.; Wang, H.; Bao, H. Spatio-Temporal Video Segmentation of Static Scenes and Its Applications. *IEEE Trans. Multimed.* **2015**, *17*, 3–15. [[CrossRef](#)]
66. MacKin, A.; Zhang, F.; Bull, D.R. A Study of High Frame Rate Video Formats. *IEEE Trans. Multimed.* **2019**, *21*, 1499–1512. [[CrossRef](#)]
67. Redmon, J.; Divvala, S.; Girshick, R.; Farhadi, A. You Only Look Once: Unified, Real-Time Object Detection. *arXiv* **2019**, arXiv:1506.02640.
68. LeCun, Y.; Haffnet, P.; Leon, B.; Bengio, Y. Object recognition with gradient-based learning. In *Shape, Contour and Grouping in Computer Vision*; Springer: Berlin/Heidelberg, Germany, 1999; pp. 1–27.
69. Lim, W.; Jang, D.; Lee, T. Speech emotion recognition using convolutional recurrent neural networks and spectrograms. In Proceedings of the Canadian Conference on Electrical and Computer Engineering, London, ON, Canada, 30 August–2 September 2020.
70. Szegedy, C.; Ioffe, S.; Vanhoucke, V.; Alemi, A.A. Inception-v4, inception-ResNet and the impact of residual connections on learning. In Proceedings of the 31st AAAI Conference on Artificial Intelligence, AAAI 2017, San Francisco, CA, USA, 4–9 February 2017; pp. 4278–4284.
71. Lyu, S.; Chang, M.C.; Du, D.; Li, W.; Wei, Y.; Coco, M.D.; Carcagni, P.; Schumann, A.; Munjal, B.; Dang, D.Q.T.; et al. UA-DETRAC 2018: Report of AVSS2018 IWT4S challenge on advanced traffic monitoring. In Proceedings of the AVSS 2018—2018 15th IEEE International Conference on Advanced Video and Signal-Based Surveillance, Auckland, New Zealand, 27–30 November 2019; pp. 1–7. [[CrossRef](#)]

-
72. González-Díaz, I.; Martínez-Cortés, T.; Gallardo-Antolín, A.; Díaz-De-María, F. Temporal Segmentation and Keyframe Selection Methods for User-Generated Video Search-Based Annotation. *Expert Syst. Appl.* **2015**, *42*, 488–502. [[CrossRef](#)]

-
73. Liu, Z.; Lian, T.; Farrell, J.; Wandell, B.A. Neural Network Generalization: The Impact of Camera Parameters. *IEEE Access* **2020**, *8*, 10443–10454. [[CrossRef](#)]
 74. Yang, B.; Guo, H.; Cao, E. Chapter Two—Design of cyber-physical-social systems with forensic-awareness based on deep learning. In *Advances in Computers*; Hurson, A.R., Wu, S., Eds.; Elsevier: Amsterdam, The Netherlands, 2021; Volume 120, pp. 39–79.