IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received 14 November 2023, accepted 3 December 2023, date of publication 7 December 2023,
date of current version 14 December 2023.

Digital Object Identifier 10.1109/ACCESS.2023.3340144

RESEARCH ARTICLE

Real-Time Event-Driven Road Traffic Monitoring System Using CCTV Video Analytics

MEHWISH TAHIR[1], (Graduate Student Member, IEEE), YUANSONG QIAO[1], (Member, IEEE), NADIA KANWAL[2], (Senior Member, IEEE), BRIAN LEE[1], AND MAMOONA N. ASGHAR[3], (Senior Member, IEEE)

[1]Software Research Institute, Technological University of the Shannon: Midlands Midwest, Athlone, County Westmeath, N37 HD68 Ireland
[2]School of Computer Science and Mathematics, University of Keele, ST5 5BG Newcastle, U.K.
[3]School of Computer Science, College of Science and Engineering, University of Galway, Galway, H91 TK33 Ireland

Corresponding author: Mehwish Tahir (mehwish.tahir@tus.ie)

This work was supported by the Presidential Doctoral Scheme (PDS) of the Technological University of the Shannon: Midlands Midwest, Athlone Campus, Ireland, as a part of the Ph.D. work.

ABSTRACT Closed-circuit television (CCTV) systems have become pivotal tools in modern urban surveillance and traffic management, contributing significantly to road safety and security. This paper introduces an effective solution that capitalizes on CCTV video analytics and an event-driven framework to provide real-time updates on road traffic events, enhancing road safety. Furthermore, this system minimizes the storage requirements for visual data while retaining crucial details related to road traffic events. To achieve this, a two-step approach is employed: (1) training a Deep Convolutional Neural Network (DCNN) model using synthetic data for the classification of road traffic (accident) events and (2) generating video summaries for the classified events. Privacy laws make it challenging to obtain extensive real-world traffic data from open-source datasets, and this challenge is addressed by creating a customised synthetic visual dataset for training. The evaluation of the synthetically trained DCNN model is conducted on ten real-time videos under varying environmental conditions, yielding an average accuracy of 82.3% for accident classification (ranging from 56.7% to 100%). The test video related to the night scene had the lowest accuracy at 56.7% because there was a lack of synthetic data for night scenes. Furthermore, five experimental videos were summarized through the proposed system, resulting in a notable 23.1% reduction in the duration of the original full-length videos. Overall, this proposed system holds significant promise for event-based training of intelligent vehicles in Intelligent Transport Systems (ITS), facilitating rapid responses to road traffic incidents and the development of advanced context-aware systems.

INDEX TERMS DCNN, event classification, intelligent transport systems, synthetic data, video summarization.

I. INTRODUCTION

Traffic management authorities are facing significant issues because of the growing urban population, which is significantly increasing traffic congestion and road accidents in urban areas. Road accident responses are delayed when there is no real-time monitoring system in place, which increases the rate of deaths. According to the World Bank, around 56% (approximately 4.4 billion people) of the world's population resides in the cities and is expected to be doubled by 2050 [1]. This population growth in cities is raising the demand for Intelligent Systems (IS) with reduced or no manual input/output in minimal time. IS use advanced technologies to manage and improve traditional systems. One of the emerging applications of IS is Intelligent Transport Systems (ITS) which can be used for traffic management, vehicle/traveler safety, parking management, and accident detection. The purpose of ITS is to reduce traffic congestion and provide rational incident management to enhance the safe mobility of people and vehicles on the roads in smart cities.

The associate editor coordinating the review of this manuscript and approving it for publication was Zhenhua Guo.

The surveillance systems installed in smart vehicles and city infrastructure are the foundation of ITS that can provide brisk services to the residents in case of any calamity. However, considering surveillance systems also brings into focus the privacy of individuals and associated objects, particularly due to the European General Data Protection Regulation (EU-GDPR) [2], [3]. EU-GDPR highlights two concerns: using individuals' personal/associated data without consent and collecting minimal data for providing services. Therefore, it is illegal to utilize real-time data of an individual and their belongings (e.g. vehicles in our case) without prior consent. Here comes artificial intelligence (AI), which plays a noteworthy part in automatic decision-making and self-learning by the system. An event-driven video system utilizing AI could be made compliant with EU-GDPR by training it with synthetic data and storing minimal required data to avoid any privacy concerns.

Moreover, it is hard to get a large number of training data for real-time incidents. The less availability of training data for any application makes it a challenging task to achieve the desired results. Therefore, synthetic data can facilitate the researchers' work in an environment where the simulation of real-world scenarios can provide promising results, even if trained on synthetic data and tested on real-time data. The concept of synthetic data is broadly used in the training of autonomous vehicles [4] and robots [5]. Google's Waymo is an autonomous vehicle project that trains its autonomous vehicles using synthetic data [6].

The storage cost and processing time are important considerations of surveillance systems. The continuous storage resulted in the necessity for additional storage space and also increased the processing time when retrieving video footage. In modern cameras, motion-based recording reduces storage costs and processing time by storing only a fraction of captured frames instead of storing all the frames [7]. Apart from the storage and processing time issues, video footage retrieval by human camera operators is also onerous, as it threatens the identity of the individual/objects. However, these human camera operators can be replaced by automated interpretation, where the system can store and retrieve the video footage as per the requirements. The automated interpretation draws the attention of researchers toward video summarization and context-aware systems [8] that can reduce the storage cost along with the processing time when video footage is retrieved. The machine learning models are extensively used in object detection [9], emotion recognition [10], disease detection [11], and many more to mention. However, this proposed research is about training a machine on a synthetic dataset for image/video scene classification problems. Therefore, a supervised machine learning model called Deep Convolutional Neural Network (DCNN) was used as per the description of the model in Section III-C.

This paper considered a critical application for ITS i.e., road traffic accident detection which is one of the leading causes of casualties globally. According to the World Health Organization (WHO), approximately 1.3 million people expire every year as a result of road accidents [12]. The number of fatalities can be reduced if timely emergency assistance is provided to injured people. The usage of surveillance systems installed in smart vehicles [13], [14], [15] and the city's infrastructure for the quick detection of accidents can save many lives. Upon detection of an accident, an alert could be sent to the relevant services (hospital, police, etc.) to take immediate action. This can also let vehicles change their lane in traffic congestion.

In the light of aforementioned issues regarding EU-GDPR, large-scale CCTV (closed-circuit television) footage storage optimization, and video analytics for accident detection, the following research questions were identified for pursuing this research work:

RQ1: Is it adequate to train a Deep Convolutional Neural Network (DCNN) model on video frames to identify events in a video, as opposed to training it on objects?

RQ2: How can the extensive storage challenges associated with recorded CCTV data be addressed in the EU-GDPR era?

RQ3: How could the event-driven CCTV-captured videos be helpful in the foundation of a context-aware ITS?

By considering the aforesaid questions, the contributions of this research article are as follows:

C1: Created new annotated training datasets (synthetic visual data) containing accident and non-accident video frames, which are now publicly available on the Kaggle repository [16] for aspiring researchers. (Section III-B)

C2: An in-house sequential DCNN model was built and trained on the synthetic visual datasets to achieve optimal results when evaluated on real-time videos. (Section III-C)

C3: An Event-driven video system is proposed to resolve the storage and processing time issues of large-scale visual data. (Section III-D)

The remainder of this paper is organized as Section II sheds light on the literature and related work. The methodology of the proposed system is elaborated in Section III and the implementation details are given in Section IV. The performance analysis is presented in Section V and the limitations are discussed in Section VI. The future directions are discussed in Section VII and in the last, Section VIII concludes this research.

II. LITERATURE AND RELATED WORK

This section highlights several techniques that made use of smart vehicles and smart cities for road accident detection and video summarization for efficient storage space utilization in ITS.

A. ROAD ACCIDENT DETECTION TECHNIQUES

Over the past few decades, various systems for traffic accident detection have developed to enhance the safety of people in ITS. An accident detection system was proposed by [17] to provide timely services to the drivers of head-on and single vehicles. The author used sensors for collision detection and a dashcam to record the footage to send a timely

emergency notification to the rescue team. An unsupervised model utilized dashboard-mounted cameras for accident detection [18]. The novelty of the proposed technique was to detect anomalies by predicting the next position of the road participants. A technique based on DCNN was used by [19] to classify four classes i.e., accident, dense traffic, fire, and sparse traffic. The model architecture included 4 convolutional layers and 3 fully connected layers. It was trained on a total of 4400 images (1100 from each class) for 100 epochs of batch size 32. Overall training accuracy achieved by the model was 94.4%, whereas, testing accuracy was 91.64%.

Another accident detection system based on deep learning was proposed by [20], which focused on accidents in India. The author collected 5000 images from Google representing two classes: accident and non-accident with 2500 images each. An overall 85% accuracy was achieved through a Convolutional Neural Network (CNN) and the model was created using sequential API. A novel technique for classifying and detecting accidents based on trajectory data from multiple vehicles was presented by [21] that detected six types of accidents including four rear-end collisions (DN, NA, SN, and LR) and two side collisions (CP and SL). Another technique achieved promising results that worked on coarse and fine detection for temporal and spatial encoding, respectively in the vehicular ad hoc networks (VANET) environment [22]. The technique used trajectory information along with the position from Global Navigation Satellite System (GNSS) to detect and classify an accident. The research presented by [23] identified the damaged vehicle and 12 types of damages were targeted in the proposed research.

The small sample sizes and imbalanced data through generative adversarial network (GAN) [24] were handled through the technique proposed by [25]. The extraction of temporal and spatial correlations of traffic flow and the detection of incidents were performed using a temporal and spatially stacked auto-encoder (TSSAE). A graph convolutional adversarial network for anomaly detection like traffic accidents through spatio-temporal characteristics was proposed by [26]. Following adversarial training, the generator and discriminator could be used individually as detectors. The generator was used for modeling normal traffic dynamics patterns, whereas, the detection criteria were provided by the discriminator. The strength of both detectors was combined to develop an anomaly score. Then, considering how unpredictable traffic dynamics can trick the discriminator, a novel anomaly score was developed by combining the strengths of two detectors to differentiate between normal and abnormal data.

Automatic detection of accidents through surveillance cameras was proposed by [27]. The CNN model was used to detect accidents or anomalies from the videos captured by the video traffic surveillance system. Further, a rolling prediction algorithm was applied to achieve high accuracy. The results showed 82% accuracy in detecting accidents successfully. In another technique, road accident detection in ITS was proposed by training the model using synthetic data captured from multiple perspectives named Multi-Perspective Road Accident Dataset (MP-RAD) [28]. The training dataset consists of 400 accidents in a total of 2000 videos. The similarity of features was estimated and based on the rank of those features, videos were divided into similar sample groups. The spatio-temporal features were extracted from each group using two-branch DCNNs, which were fused using a rank-based weighted average pooling technique, and then classification was performed. The authors performed two types of cross-validation to show the relevance between real and synthetic datasets. The model was trained with the MP-RAD dataset and tested on real-time videos and vice-versa. The model outperformed when trained on synthetic data and tested on real-time videos.

The synthetic dataset was used to train You Only Look Once (YOLO) followed by testing on real-time videos by [29]. The synthetic dataset was annotated as per the requirement of YOLO. The results showed that the use of YOLO is not only for object detection but alternately, it can be trained on a customised dataset for event detection as well. The results were made useful by providing privacy-preserved summarized videos of road accidents in ITS, which can only be accessible to authorized stakeholders. Table 1 shows the comparison of the proposed system with existing accident detection techniques.

B. VIDEO SUMMARIZATION TECHNIQUES

Video summarization has a significant role in space and time utilization. The research conducted in this area is mostly on moving object-based video summarization. A real-time traffic accident approach based on vision was presented by [30]. The vehicles were extracted using the Gaussian Mixture Model (GMM) and then tracked from the videos through the mean shift algorithm. The research focused on the position, direction, and speed of the moving vehicles to decide whether the accident occurred or not. One of the moving/non-moving object-based research was conducted by [31], in which non-moving objects were discarded first, and then a group of moving objects in inter and intra-frames was created to summarize the video. A perceptual video summarization technique also detects an accident through the features of moving objects [32].

Another technique proposed by [33] was based on the estimation of optical flow and heuristic calculation for adaptive threshold. The experiments proved the effectiveness and applicability of the proposed algorithm. Situations such as tunnels and collisions at the intersections were not observed. The video summarization technique using Faster Regions with Convolutional Neural Networks (R-CNN) summarized the videos to detect traffic rules violators [34]. A superframe segmentation-based technique for retrieving an event from long videos was proposed by [35]. The motion amplitude was used to remove redundant frames from long videos and the

TABLE 1. Comparison of the proposed system with existing accident detection techniques.

Reference	Input	No. of Classes	Names of Classes	Training Dataset	Epochs	Batch Size	No. of Layers
B. Kumeda et al. (2019) [19]	Video frames	4	Accident, Dense traffic, Fire, Sparse traffic	4400	100	32	7
G. Rajesh et al. (2020) [20]	Video frames	2	Accident, Non-accident	4000	30	100	4
D. Yang et al. (2021) [21]	Trajectory related information of two cars (9 variables for each class)	7	NM, DN, NA, SN, LR, CP, SL	3500	–	–	6
T. Kosambia et al. (2021) [37]	Video frames	3	Accident, pre-accident, post-accident	–	100	100	152
S. W. Khan et al. (2022) [27]	Video frames	1	Accident	1360	10	32	–
M. Tahir et al. (2023) [29]	Images and video frames	2	Accident, Non-accident	600	30	16	157
Proposed System	Video frames	2	Accident, Non-accident	13,228	25	32	6

text questions were matched with the answers generated from the trained semantic model. Similarly, a hybrid model based on machine learning for the summarization of cricket videos was proposed by [36].

A frame-based video summarization for accident detection using ResNet (deep learning) was proposed by [37]. The model used the ImageNet dataset for training on three classes known as pre-accidents, accidents, and post-accidents. An Object-of-Interest (OoI) based video summarization was proposed by [38] which summarized the video by taking two inputs: video and OoI. Deep learning was used to detect OoI from the video and then the video was summarized if the relevant object was detected in the video. Z-numbers-based spatio-temporal rough fuzzy granulation (Z-STRFG) is a video summarization technique that resolved the issue of identifying uncertainty between collision and near-miss in accidents [39]. The approximate anomaly-prone regions in terms of granules were obtained using Z-STRFG by computing various spatio-temporal properties over the video frames. These regions may have uncertainty among the crash, near-miss, and regular traffic scenarios. To distinguish between the aforementioned three cases, two types of rough fuzzy granules (RFGs) were computed along with their roughness scores. Another technique proposed video summarization by training YOLO using synthetic data [40].

Existing literature focuses mostly on accident detection, while event-based video summarization is hardly considered. By considering identified research gaps, this paper proposes an optimized solution for event classification (accident detection) and then video summarization for large-scale visual data storage issues. Through the literature survey, it is evident that several proposed techniques used VANETs [41] for predicting and detecting accidents. One of the limitations of these techniques is the installation of specific devices in the vehicles to detect accidents. This paper proposes a framework for event (road accidents) detection and video summarization through CCTV recordings utilizing the infrastructure of smart cities i.e., ITS. Furthermore, enhancing traditional network video recorders (NVR) to incorporate event-driven video storage would contribute to increased efficiency.

III. METHODOLOGY OF PROPOSED SYSTEM

This section provides a detailed explanation of the core functionalities of the basic building blocks in the proposed system. It also delves into the data collection and annotation along with the DCNN model architecture and outlines the stages involved in the event-driven video system.

A. FUNCTIONALITY OF THE BUILDING BLOCKS

The proposed event-driven road traffic monitoring system has the following five fundamental building blocks, which were considered for the foundation and functionality of the proposed system, shown in Fig. 1. The importance of each building block is described below:

1) **Security and Privacy** are the main concerns of all kinds of data management system that deals with individuals and objects. To make the system EU-GDPR compliant, the privacy of individuals and objects was considered in the proposed system by utilizing synthetic data for training the DCNN model.
2) **Model Integration** is vital to perform a specific task or to solve a given problem by the machine, like road traffic event classification in our case. A DCNN model was trained on customised synthetic data in the proposed system.
3) **Sensor Integration** is another important aspect to address. Several types of sensors and data sources are used to collect and analyze different types of data like

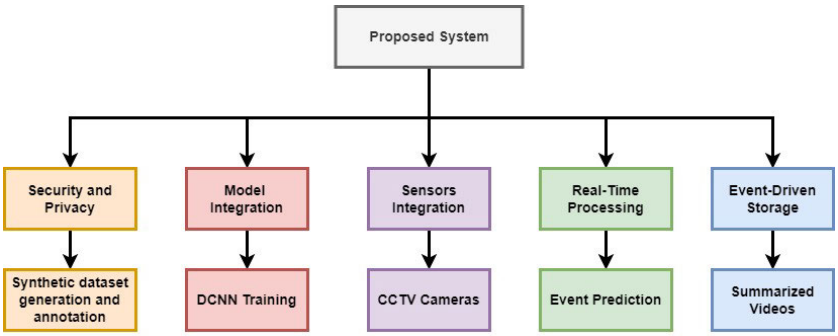

FIGURE 1. Building blocks and functionality of the proposed event-driven road traffic monitoring system.

Global Positioning Systems (GPS), accelerometers, microphones, and CCTV. The CCTV cameras installed in the smart cities were used for collecting visual data in the proposed system followed by the analysis of visual data and extraction of the contextual information from the CCTV for further use.

4) **Real-time Processing** is the building block that evaluates the system on real-time data. For example, the proposed system was evaluated on real-time CCTV-captured videos.
5) **Event-driven Storage** is another requisite functionality of our system. It stored the segment of the CCTV-captured video in which the relevant event was detected along with the contextual information like camera ID, location, video ID, time, and date. It also ensured EU-GDPR compliance by storing minimal required event-classified data for future operations.

B. DATA COLLECTION AND ANNOTATION

The proposed system used the synthetic dataset for training purposes and then evaluation was performed on real-time videos for event detection. Consequently, the synthetic data videos were downloaded in higher resolution from BeamNG Drive [42] YouTube channel after getting permission from the authorized stakeholder. The frames from these downloaded synthetic videos were extracted, manually annotated into the accident and non-accident categories, and made publicly available for future researchers at [16]. After careful annotation of the frames, a total of 13,228 (6,614 from each class) frames were selected for training the model as mentioned in Table 2. The sample synthetic dataset frames of both classes (accident and non-accident) are shown in Fig. 2.

TABLE 2. Test data details.

Total Frames	Accident	Non-accident
13,328	6,614	6,614

C. DEEP CNN MODEL ARCHITECTURE

Certainly, compared to a pre-trained DCNN model to classify test data, this is generally seen as a more spectacular accomplishment. The design of a sequential (API) DCNN

FIGURE 2. Sample annotated training dataset (synthetic data).

model is easy to understand in which the layers are stacked one on top of the other in order, but it is only compatible with stacks of layers that have a single input and output tensor. The number of layers varies and could be added to the model as per the requirement because several convolutional and pooling layers make up a DCNN. Keras is a high-level deep learning API for the implementation of neural networks, therefore, the proposed custom sequential DCNN model is implemented in Keras. The features representing accidents and non-accidents in the training data were extracted through the Conv2D layer and then the frame was down-sampled using Max_Pooling2D.

The architecture of the custom sequential DCNN model is shown in Fig. 3 and is trained from scratch with a customised

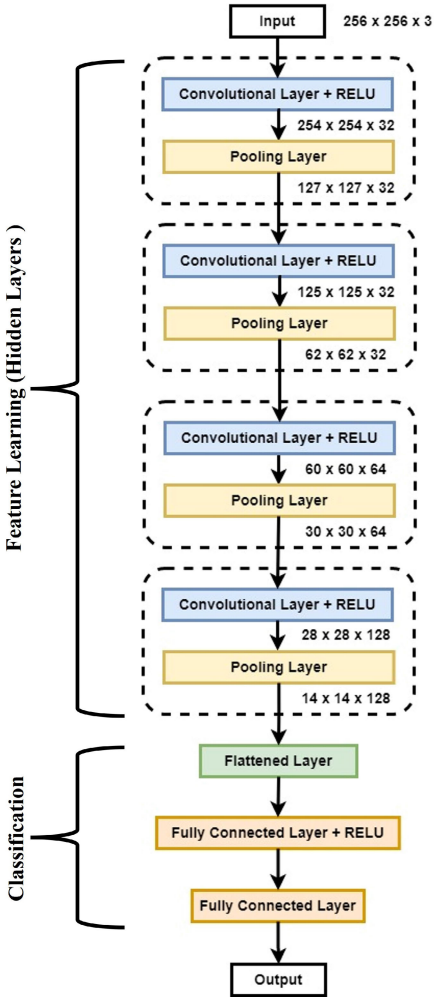

FIGURE 3. CNN architecture for the proposed system.

synthetic training dataset. There are a total of 11 layers in the model including four Conv2D layers, four Max_Pooling, 1 flattened layer, and 2 dense layers. However, the model architecture has 6 distinct layers (excluding pooling and flattening). The model is divided into two main parts known as feature learning and classification. The details of both parts are as follows:

1) FEATURE LEARNING (HIDDEN LAYERS)

Firstly, feature learning is performed by using convolutional and pooling layers. As previously mentioned, our model is trained on 4 convolutional and 4 pooling layers. The purpose of the convolutional layer is to divide the image into features and then see if these images are available in the test image. It matches the features rather than pixel locations. Afterwards, filters are created and in a convolutional layer, an image becomes a stack of filtered images. The number of filtered images varies according to the number of filters. The output of each layer is calculated by:

$$\frac{n+2(p)-k}{s}+1\times\frac{n+2(p)-k}{s}+1\times filters \quad (1)$$

where n represents the dimension of the image, p is padding, k is kernel size, and s is stride.

The images are input one by one to the model for training. In our proposed system the image size is $256 \times 256 \times 3$. The value $p = 0$ and $s = 1$ in the convolutional layers and $s = 2$ in the pooling layers. All the layers have a fixed $k = 3$. The number of filters used is the same in the first two convolutional and pooling layers which is 32, however, in the third and fourth layers it is 64 and 128, respectively.

The Relu function shown in (2) was used as an activation function in all the layers except the last layer. It returns 0 if the value is negative otherwise it returns the value itself.

$$Relu(a) = \begin{cases} 0 & a < 0 \\ a & a \geq 0 \end{cases} \quad (2)$$

2) CLASSIFICATION (FLATTENED AND DENSE LAYERS)

After the feature learning extracted the relevant features from the input image, the flattened layer transformed the 3D tensor ($14 \times 14 \times 128$) into a 1D vector of length 25088. Each element in the 1D vector represents a feature or a combination of features learned by the previous layer. Therefore, this vector is then presented as input to the next layer i.e., the dense layer or fully connected layer. In mathematical notation, input to a dense layer with M units (neurons) is represented as a vector having $N \times 1$ dimensions, if there is a flattened vector (x) with length N. Subsequently, the weights of the dense layer are applied to each element of the vector to produce another output vector (y) of length M. The following equation is used to calculate the y:

$$y = W \cdot x + b \quad (3)$$

where W is the weight matrix, and b is the bias vector (provides an additional degree of freedom for the model to learn). The last two layers were dense layers in the proposed system and the purpose of dense layers was to select features that belong to the accident or non-accident class. The first dense layer had $W = 256$, $x = 25088$, and $b = 256$, therefore, $y = 6422784$. However, in the second dense layer $W = 1$, $x = 256$, and $b = 1$, therefore, $y = 257$. The model used Sigmoid probability to classify the frame into the accident or non-accident class. The following equation of the Sigmoid activation function was used to classify the frame:

$$Sigmoid(a) = \frac{1}{1+e^{-a}} \quad (4)$$

D. PROPOSED EVENT-DRIVEN VIDEO SYSTEM

Fig. 4 and Algorithm 1 show the phases of the adopted video summarization methodology with pseudo-code, respectively. The proposed methodology is comprised of three phases such as 1) buffering, 2) classification, and 3) storage, which are narrated below:

1) **Buffering:** All the temporally dependent frames captured within one second through the CCTV camera

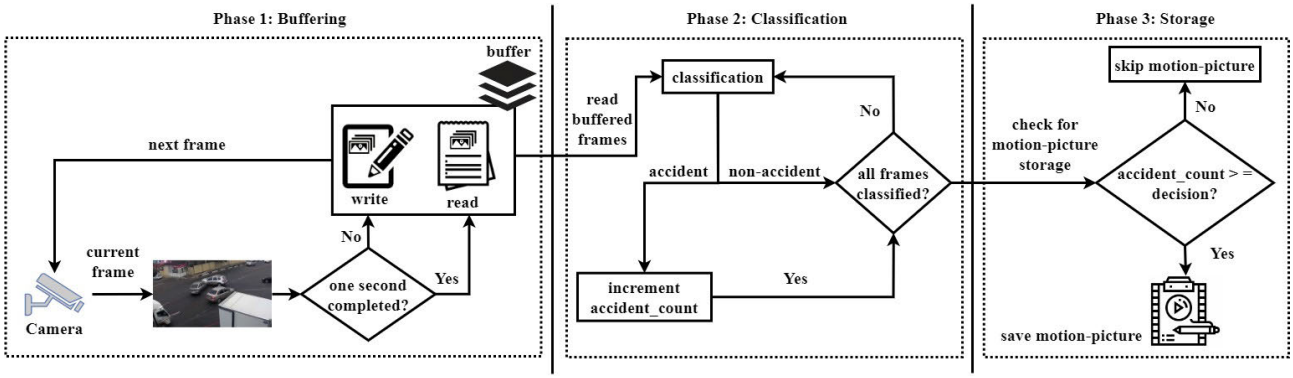

FIGURE 4. Phases of the proposed event-driven road traffic monitoring system.

were sent sequentially to the buffer for provisional storage. A batch of frames that were captured during one second was stored in the buffer. Thenceforth, the charge was handed over to the next phase, i.e., classification. For instance, if the frame rate of a CCTV camera is 30, the buffer would hold a total of 30 frames which would be classified in the next phase.

2) **Classification:** The model architecture explained in Section III-C, trained on customised synthetic data was used in this phase. The already trained model is loaded in this phase which sequentially reads all the frames from the buffer. Then uses spatial dependencies in the frames to classify them as accidents or non-accidents. When a frame was read from the buffer, it had to be prepared for classification. Steps 14 to 17 in Algorithm 1 show the preparation of the frame which includes resizing, conversion to a NumPy array, normalization, and reshaping. Step 18 predicted the class for that prepared frame by loading the saved model and weights. A variable named *accident_count* was used to count the number of frames (captured in one second) classified as accidents. If a frame was classified as an accident, the variable *accident_count* was incremented by 1 as shown in (5), otherwise, the next frame was read without increment in *accident_count*.

$$accident\_count \leftarrow accident\_count + 1 \tag{5}$$

This process resumed until all the frames in the buffer were classified. Afterward, the control was bequeathed to the storage phase. For example, if a video had 30 fps, 10 frames were identified as accident frames in the first second then the *accident_count* value was 10.

3) **Storage:** In this phase, a decision was taken whether the frames classified by the previous stage were to be the part of motion-picture or not. It was done by comparing two variables *accident_count* and *decision* as $accident\_count > decision$. The *decision* was calculated by dividing the *fps* by 4 (one-quarter of the frames captured in one second) as shown in (6), therefore, in the case of 30 *fps*, *decision* was 7.

$$decision = \frac{fps}{4} \tag{6}$$

The example mentioned in the previous phase had $accident\_count = 10$, which means *decision* was greater than *accident_count*. According to the Algorithm 1 proposed for this framework, all the frames in that one second were saved as motion-picture for scene formation (video summarization). The storage of all the frames captured in one second to make motion-picture through a sequence of images, where x, f, and n represent frame number, frame, and *fps*-1, respectively, is denoted by a set builder notation as:

$$S^{fps} = \{f \in V \mid accident\_count > decision \; and \; 0 \leq x \leq n\} \tag{7}$$

The summary of the frames captured in one second (S^{fps}) contains all the frames that are extracted from the original video (V) footage. For a f to be a part of S^{fps}, $accident\_count > decision$ and $0 \leq x \leq n$, both should be true. Thus, this logic will give context to the detected event in the full summarized video (S) by storing a few frames before or after the event. The logical relationship between S and V is represented by the Venn diagram in Fig. 5. If video footage is very small in length (just a few seconds), it is possible that all the frames are part of S and in that case, S would be the subset of V i.e., $S \subseteq V$. If we take the union of S and V, it would be equal to V:

$$S \cup V = V \tag{8}$$

Contrarily, for a longer video (minutes or hours) the possibility of detecting an event in V is non-viable, therefore, S would be a shorter part of V. In the latter scenario, S would be the proper subset of V i.e., $(S \subset V)$, and all the frames of the video would not be part of the original video. In this case, if we take the intersection of S and V, it would be equal to S:

$$S \cap V = S \tag{9}$$

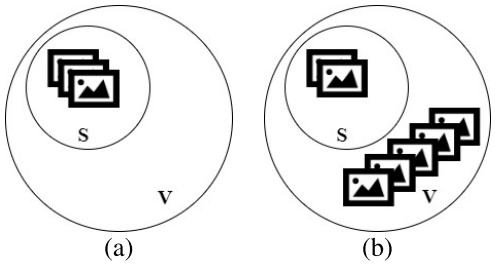

FIGURE 5. (a) Original video footage (V) is small, and most frames are detected as accidents, therefore, summary (S) is the subset of V, and (b) V is a large CCTV footage and accident is detected in a segment of the video, therefore, S is the proper subset of V.

TABLE 3. System specifications used for conducting experiments.

System	Specifications
System Model	G5 5000
Processor	Intel(R) Core(TM) i7-10700F CPU @ 2.90GHz 2.90 GHz
GPU	NVIDIA Geforce RTX 3070
System Type	64-bit operating system, x64-based processor
Installed RAM	16.0 GB
Operating System	Windows 11

IV. IMPLEMENTATION

A. EXPERIMENTAL SETUP

The test-bed for the proposed event-driven video system was developed in Python language on the system specifications given in Table 3.

B. DEEP LEARNING MODEL TRAINING

1) **Training Dataset Preparation:** The pre-processing of images is the prime and essential step that ought to be performed when visual data is employed. The proposed model extended the training data through AI using image augmentation which includes: rescale = 1./255, shear_range = 0.2, zoom_range = 0.2, and horizontal_flip = True. These training data images are now used to train the images.
2) **Model Training:** The training data images prepared in the last step are used for model training. The model was trained on a batch size of 32 for 25 epochs. The most common Adam optimizer was used to compile the model along with the binary_crossentropy loss function, and the accuracy metric was used to monitor the training of the model. There was a total of 6,525,537 parameters that were all trainable and none of the parameters were non-trainable. The values of trainable parameters were those that were updated or adjusted according to their gradient during training. Due to the synthetic nature of the data, the increase in training accuracy and decrease in training loss was abrupt as shown in Fig. 6. Once the model was trained, the model architecture and weights were stored for testing on real-time data. The model architecture and the trained weights took 8KB and 24.9 MB of space on the disk, which is very less. The model was trained in approximately 3.25 hours.

Algorithm 1 Pseudo-Code of the Proposed Event-Driven Road Traffic Monitoring System

```
/* Load architecture and weights of the
   trained model along with video to be
   summarized                                   */
Load and Compile: saved model along with weights;
Input: video;
Read: height, weight, fps;
Calculate: decision = fps/4, where 4 represents one-quarter
frames;
while (Read video) do
    /* read video frame                        */
    read frame from video;
    if frame_count <= fps then
        write frame in buffer;
        increment frame_count by 1;
    end
    if frame_count == fps then
        for all the frames stored in buffer do
            /* read and prepare frame for
               classification                   */
            frame ← read frame from the buffer;
            F1 ← resize frame to 256 × 256;
            F2 ← create Numpy array of F1;
            F2 ← divide F2 by 255 to normalize;
            F2 ← reshape F2;
            /* predict class of the frame       */
            P ← predict class of the frame F2;
            if accident is True then
                increment accident_count by 1;
            end
        end
    end
    /* Decision about summary writing           */
    if accident_count > decision then
        reset accident_count to 0;
        /* read frames from the buffer and
           write them in video format           */
        for all the frames stored in buffer do
            read frames from buffer;
            write summary;
        end
    end
end
video.release();
summary.release();
/* Summary of the video                         */
Output: Event-driven video
```

V. PERFORMANCE ANALYSIS

This section describes the detailed performance analysis of the proposed system.

A. COMPUTATIONAL COMPLEXITY

To show the effectiveness of the proposed system, the computational complexity of the implemented Algorithm 1 is calculated as:

- Steps 2-4 have a constant time complexity of $O(1)$.

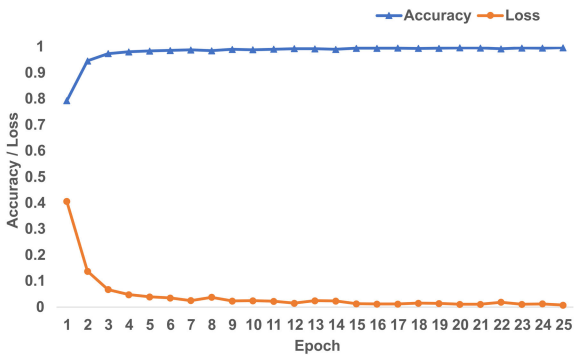

FIGURE 6. Accuracy and loss graph for training on the synthetic dataset.

- The while loop starting at line 5 iterates over all the frames in the video. The time complexity of the loop depends on the number of frames in the video. Let's assume there are N frames in the video, then the time complexity is $O(N)$.
- Steps 6-10 and 11-22 are executed N/fps times. Therefore, their time complexity can be expressed as $O(N/fps)$.
- Step 24 has a constant time complexity of $O(1)$.
- The for loop starting at Step 26 is executed only when the accident count exceeds the decision threshold. The maximum number of frames in the buffer is fps. Therefore, the time complexity of this loop can be expressed as $O(fps)$.

Overall, the time complexity of the algorithm can be expressed as $O(N/fps \times fps) = O(N)$. Therefore, the proposed algorithm has a linear time complexity which is $O(N)$, where N is the number of frames in the video.

B. EVALUATION ON REAL-TIME VIDEOS

The proposed system was evaluated on real-time publicly available CCTV videos on YouTube [43] comprising accident and non-accident classes. The model architecture and trained weights were loaded to perform classification. The test frames were taken from ten different real-time videos that were unseen for the trained model. The details of videos like video number, number of frames, frames per second (fps), and resolution along with test results like a true accident, false accident, true non-accident, false non-accident, and accuracy are presented in Table 4. The results achieved from the classification show that on average 82.3% frames were correctly classified. Fig. 7 presents sample classification on frames from the test videos for accident and non-accident video frames along with the classification accuracy.

If the total number of frames in one second is 30, the model will consume approximately 0.45 seconds to classify all the frames captured in one second, which is defined as:

$$Time = \sum_{f=1}^{n} (t_f, t_{f+1}, t_{f+2}, \ldots, t_n) \tag{10}$$

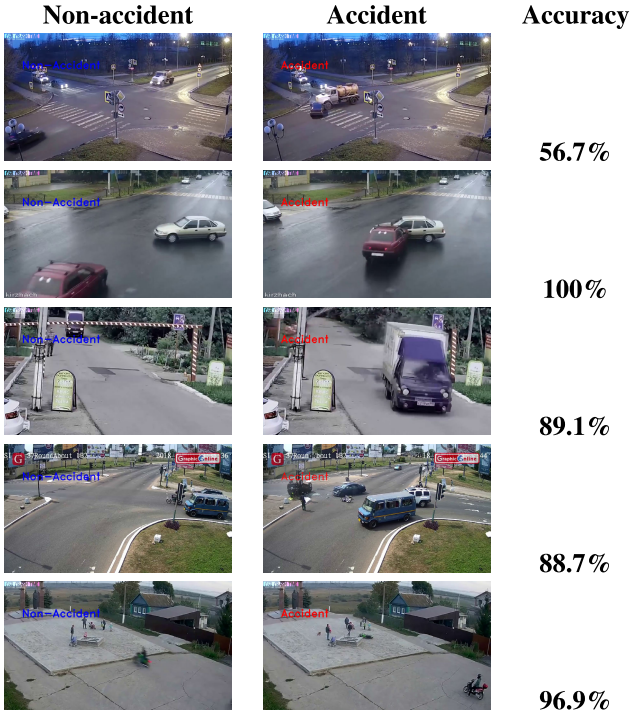

FIGURE 7. Test results along with detection accuracy on each real-time video footage, where non-accident and accident frames are labelled in blue and red colors, respectively.

where f is representing frames and n is the last frame number. The average time $\overline{Time}$ taken by the model to classify one frame is approximately 0.015 seconds, where $Time$ is the total time:

$$\overline{Time} = \frac{Time}{fps} \tag{11}$$

The time taken by the model to classify all the frames of each video is shown in the graph Fig. 8. The time is directly proportional to the length of the video.

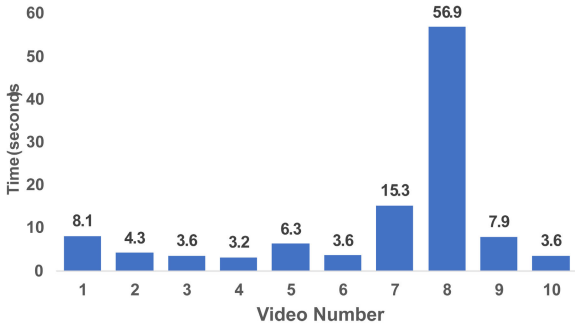

FIGURE 8. Time taken by each video footage for classification.

C. PERFORMANCE METRIC FOR REAL-TIME VIDEOS

There is a total of 7,510 frames from ten different real-time videos that are used for evaluation purposes. The average time consumed for the classification of one frame is 0.015 seconds. The performance of the DCNN model

TABLE 4. Real-time test-data specifications along with the event detection accuracy.

Video No.	No. of Frames	fps	Resolution	True accident	False accident	True Non-accident	False Non-accident	Accuracy
1	541	30	1280x720	382	2	140	17	96.9
2	284	30	1280x720	163	18	90	13	89.1
3	239	30	1280x720	162	0	77	0	100
4	210	30	1280x720	155	55	0	0	73.8
5	420	30	1280x720	247	46	104	23	83.1
6	240	30	1280x720	128	61	51	0	74.6
7	1017	30	1920x1080	787	41	99	90	87.1
8	3792	30	538x360	2644	326	144	678	73.5
9	529	23	1280x720	424	6	45	54	88.7
10	238	30	1280x720	122	103	13	0	56.7

used in the proposed system was statistically measured by accuracy, sensitivity, specificity, precision, recall, and F1_score is shown in Table 5. As this research has focused on two classes; accident and non-accident, the mentioned performance metrics are explained below by keeping in view the application used in this research. T_A and T_N represent the total number of images that were correctly classified as accidents and non-accidents, respectively. Whereas F_A and F_N represent the total number of images that were incorrectly classified as accidents and non-accidents, respectively.

Accuracy is the number of frames that were correctly identified as accidents and non-accidents from the total number of frames provided to the model for testing. According to (12), the overall accuracy of the model was 82.3% on all of the test images.

$$Accuracy = \frac{T_A + T_N}{T_A + F_A + T_N + F_N} \tag{12}$$

Sensitivity is the ability of the trained model to predict how many frames were correctly classified as accidents. The sensitivity value obtained using (13), was 85.6%.

$$Sensitivity = \frac{T_A}{T_A + F_N} \tag{13}$$

Specificity is the ability of the trained model to predict how many frames were correctly classified as non-accident. The specificity value achieved using (14), was 53.7%.

$$Specificity = \frac{T_N}{T_N + F_A} \tag{14}$$

Precision quantifies the number of accident class predictions that actually belonged to the accident class. The precision value calculated using (15), was 88.8%.

$$Precision = \frac{T_A}{T_A + F_A} \tag{15}$$

Recall is calculated as the ratio between the number of accidents class correctly predicted as accidents and the total number of frames in the accident class. The recall value is directly proportional to the correct prediction of the accident class. The recall value calculated using (16), was 85.6%.

$$Recall = \frac{T_A}{T_A + F_N} \tag{16}$$

F1_score is the harmonic mean of precision and recall which is calculated using (17). The F1_score for the proposed model was 87.1%.

$$F1\_score = 2 \cdot \frac{Precision \cdot Recall}{Precision + Recall} \tag{17}$$

D. EXPERIMENTAL RESULTS

The reduction in length of five videos captured in different environmental conditions is shown in Table 6. The original video and summary duration are given in seconds along with a reduction of video duration in percentage. The analysis shows that the total duration of the five videos was 56.3 seconds which was reduced to 43.3 seconds (23.1%) after summarization through the proposed system. Fig. 9 shows the timelines for test results on these real-time videos representing which part of the videos were considered to be kept/discarded in summary after accident detection. The discarded frames, incorrect detection in numerous frames and few frames, and the final summary of each video are highlighted in the figure.

1) EXPERIMENT 1

A camera recording of a no-vehicle entry was shown in this video. A biker attempted to enter an area where some kids were playing and caused an accident. This video was approximately 19 seconds, 541 frames, 30 *fps*, and 1280 × 720 resolution. The classification results achieved for this video were as follows:

- The first 5 seconds were not considered the part of summarization, as most frames were true non-accident.
- Then the next 13 seconds had the majority of true accident frames in each second, so all the frames were saved in the summarized video as per the ground truth.

Finally, the original length of this video was 18 seconds which was reduced to 13 seconds (27.7%).

2) EXPERIMENT 2

This was a video of an area where a barrier was used as a safety measure before allowing the vehicles to pass that area. The video shows that the driver attempted to cross the barrier without waiting to open it after the security check and broke

TABLE 5. Performance metric for real-time videos.

Frames	Accuracy	Sensitivity	Specificity	Precision	Recall	F1_score	Time (sec)
7510	82.3%	85.6%	53.7%	88.8%	85.6%	87.1%	0.015

TABLE 6. Reduction in the length of real-time videos.

Video No.	VL (sec)	SL (sec)	Reduction (%)
1	18	13	27.7
2	9.4	6.4	31.9
3	7.9	5.9	25.3
4	7	7	0
5	14	11	21.4

the barrier. This incident was recorded as an unwanted road event known as an accident. The video had 284 frames, 30 *fps*, 1280 × 720 resolution, and 9.4 seconds duration. According to the results:

- In the first 3 seconds, all the frames were non-accident.
- In the fourth second, 18 frames were classified as false accidents, and 12 frames were classified as true accidents which were correct according to the ground truth.
- According to the classification in the fifth second, 17 frames were classified as true accidents and 13 frames were classified as false non-accident.
- The results in the remaining 5 seconds showed all the frames correctly classified as a true accident as per the ground truth.

The proposed system reduced the length of video 2 from 9.4 to 6.4 seconds i.e., a 31.9% reduction.

3) EXPERIMENT 3

This video was footage of a location where an accident occurred in rainy weather. The specifications of the video were 239 frames, 30 *fps*, 1280 × 720 resolution, and 7.9 seconds duration. The result shows that all the frames in the video were correctly classified as accident and non-accident frames. The original video was reduced to 5.9 seconds i.e., a 25.3% reduction. The results of this video showed that the proposed system gave promising results under rainy weather as well.

4) EXPERIMENT 4

An accident in daylight was recorded in this video. The length of this video is 7 seconds with 30 *fps* and 1280 × 720 resolution. The result showed:

- In the first two seconds 30 frames and 24 frames were classified as false accidents, respectively.
- The remaining frames in the video were classified as true accidents.

This video was not reduced as the video was reduced based on the number of accidents detected. This video classified all the frames as accidents either true or false. Therefore, when the video was summarized, the lengths of the original and summarized videos were the same.

5) EXPERIMENT 5

This footage was a recording of a crowded main road where an accident occurred. The video specifications include 420 frames, 30 *fps*, 1280 × 720 resolution, and 14 seconds duration. According to the results:

- There were numerous false accidents in the second and fifth seconds of the original video which caused these two seconds to be part of the summarized video.
- In the first, third, and fourth second all frames was classified as true non-accidents.
- In the sixth, seventh, and eleventh to thirteen seconds of the video, only a few frames were classified as false non-accidents.

If only a few frames were classified as false accidents and non-accidents, they can be ignored without compromising the video summary results as per the proposed system. This video was reduced to 21.4% which was 11 seconds in duration.

The aforementioned different examples show that the proposed system is applicable to summarize road traffic events in ITS.

E. COMPARATIVE ANALYSIS

This section compares the proposed system with existing techniques through qualitative and quantitative analysis.

1) **Qualitative Analysis**
of the proposed system with the existing techniques for accident detection and video summarization presented in Table 7 indicates a variation: certain existing methods operate on VANET, while others are centered around infrastructure. Most techniques in the literature used sensors (other than CCTV cameras) for collision/accident detection, but making sure that all the participating vehicles have sensors installed is challenging. Therefore, infrastructure-based systems are getting the attention of researchers in which CCTV cameras are used to identify events. The comparison also shows that the real-time dataset is used for training and testing purposes in the existing techniques. Whereas, the synthetic dataset for training and real-time data for testing is used in a few techniques. In our previous research, YOLO was trained using synthetic data and in the proposed research, DCNN is trained on synthetic data. Most techniques are working on deep learning or DCNN. The proposed research

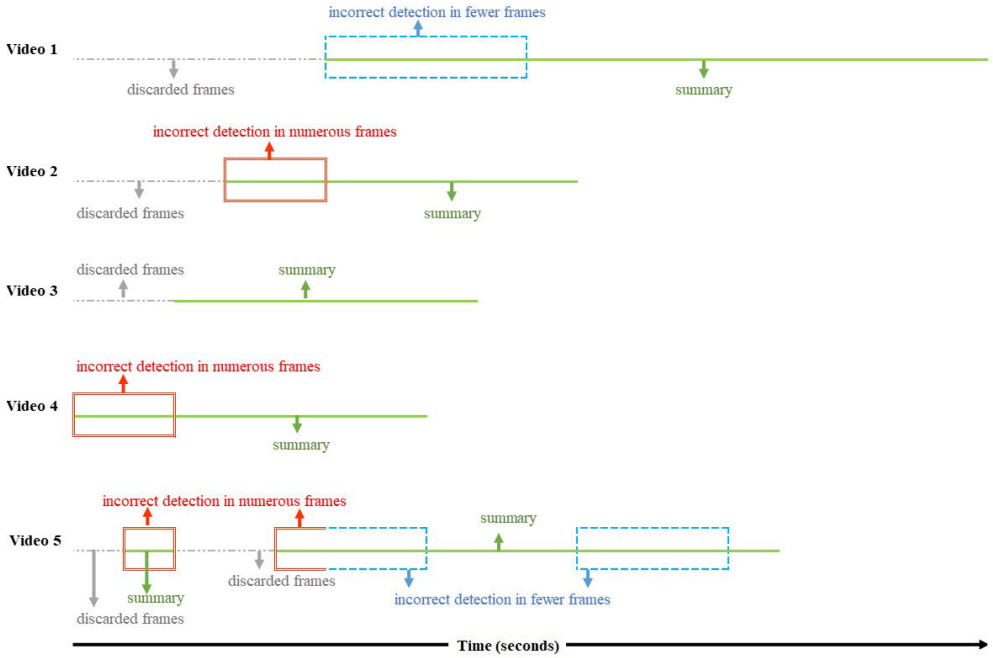

FIGURE 9. Test results timelines on real-time videos showing discarded frames, incorrect detection in numerous and fewer frames, and summary.

TABLE 7. Qualitative analysis of the proposed system with existing techniques.

Existing Approaches	System	Training Data	Testing Data	Model	Summarization
W.-J. Chang et al. (2019) [17]	VANET	Real-time	Real-time	Deep Learning	×
B. Kumeda et al. (2019) [19]	–	Real-time	Real-time	CNN	×
G. Rajesh et al. (2020) [20]	Infrastructure	Real-time	Real-time	CNN	×
D. Yang et al. (2021) [21]	VANET	Simulation	Simulation	Deep CNN	×
T. Kosambia et al. (2021) [37]	Infrastructure	Real-time	Real-time	ResNet-152	✓
Z. Zhou et al. (2022) [22]	VANET	Real-time	Real-time	Multi-layer NN	×
S. W. Khan et al. (2022) [27]	Infrastructure	Real-time	Real-time	CNN	×
T. K. Vijay et al. (2022) [28]	Infrastructure	Synthetic	Real-time	Deep CNN	×
M.Tahir et al. (2023) [29]	Infrastructure	Synthetic	Real-time	YOLO*v5*	✓
Proposed System	**Infrastructure**	**Synthetic**	**Real-time**	**Deep CNN**	✓

used a frame-based classification technique for an event-driven video system.

2) **Quantitative Analysis** through classification accuracy of the accident frames is given in Table 8. According to the details and visual results in Fig. 7, the proposed system is providing a promising result with an average accuracy of 82.3% (ranging from 56.7% to 100%) using synthetic data for model training. The lowest accuracy, standing at 56.7%, was observed in the test video associated with the night scene due to an insufficiency of synthetic data for night scenes used during training. The results are also compatible enough with the existing techniques to achieve higher classification accuracy.

VI. LIMITATIONS

The average model accuracy for road traffic events classification could be improved by considering the following key points:

- The synthetic dataset generation from the perspective of CCTV cameras in smart cities for the training of the model.
- The use of real-time test videos with good quality for model training.
- The use of multiple CCTV cameras to record a scene from different angles.
- The training dataset includes mostly four-wheelers only, humans and other vehicles (like bikes) were not part of our test data.
- Testing is performed on short video clips (already available), so videos are not greatly reduced in length.
- The focus of this research was only on two classes i.e., accident and non-accident road traffic events. The pre-accident and post-accident scenarios were not considered.
- This research is implemented and compared with CNN models, not 3D-CNN models.

TABLE 8. Quantitative analysis of the proposed system with existing techniques.

Existing Approaches	Classification Accuracy
W.-J. Chang et al. (2019) [17]	96%
B. Kumeda et al. (2019) [19]	91.64%
G. Rajesh et al. (2020) [20]	85%
D. Yang et al. (2021) [21]	95%
T. Kosambia et al. (2021) [37]	98%
Z. Zhou et al. (2022) [22]	65.6% - 84.5%
S. W. Khan et al. (2022) [27]	82%
T. K. Vijay et al. (2022) [28]	–
M. Tahir et al. (2023) [29]	55% - 85%
Proposed System	**56.7% - 100%**

VII. FUTURE DIRECTIONS

- The accuracy of the trained DCNN model and video summarization can be further improved by increasing the training data for the particular scenario. Furthermore, the use of multi-view synthetic data [28] could be beneficial to improve the authenticity of the proposed system, and the number of classes can be extended to detect other events related to the ITS in smart cities.
- In the future, the 3D-CNN model can undergo training with synthetic data to enable a comparison of results between CNN and 3D-CNN models.
- Moreover, intelligent vehicles and smart city infrastructure (ITS here) can also be trained to reduce traffic delays due to congestion. If an accident occurs, ITS can send an alert to emergency health services for immediate action. Meanwhile, another signal is sent to vehicles that can use an alternate way i.e., lane or route change to reduce traffic congestion.
- The proposed system detects events from visual data using CCTV videos. Additional information like the date and time of the event along with the weather conditions, person/vehicle behaviour, vehicle speed, and traffic congestion can be used to enhance the proposed system. The enhanced proposed system would be a context-aware system [8] in ITS which will give context to the detected event. The applications of the context-aware system in ITS include public safety, reduced traffic congestion, and incident detection.
- Undoubtedly, AI simplifies the lives of people in smart cities, but it poses certain complications, including data collection and protection when shared in infrastructure. Event-based privacy protection using encryption [3], [45] in the event-driven videos along with the key management schemes is another extension of the proposed system.

VIII. CONCLUSION

This paper proposes an effective and pioneer solution for large-scale visual data storage challenges in Intelligent Transport Systems (ITS) using event classification and event-based video summarization approaches. The proposed real-time road traffic monitoring system is based on five fundamental building blocks. The security and privacy of individuals/objects are considered due to the CCTV camera-oriented system. Synthetic data is widely used for the training of autonomous vehicles and robots. In consideration of EU-GDPR requirements for data protection in smart cities, a customised supervised model (DCNN) was trained to achieve the targeted results. The proposed system produced promising results while tested on videos captured in varying environments and lighting conditions such as night, rainy, etc. The videos were fairly reduced in duration i.e., 23.1% after event-based video summarization. The comparative evaluation with existing approaches shows that the adopted scheme is providing an average of 82.3% (ranging from 56.7% to 100%) accuracy on test data. The accuracy for the test video associated with the nighttime scene was the lowest, reaching 56.7%, primarily due to the insufficient availability of synthetic data specifically tailored for nighttime scenarios. With developed functionality, the proposed system will be invaluable for future context-aware road traffic monitoring systems within modern ITS smart infrastructures.

REFERENCES

[1] World Bank. *Overview*. Accessed: Sep. 24, 2022. [Online]. Available: https://www.worldbank.org/en/topic/urbandevelopment/overview

[2] M. N. Asghar, M. S. Ansari, N. Kanwal, B. Lee, M. Herbst, and Y. Qiao, "Deep learning based effective identification of EU-GDPR compliant privacy safeguards in surveillance videos," in *Proc. IEEE Int. Conf. Dependable, Autonomic Secure Comput., Int. Conf. Pervasive Intell. Comput., Int. Conf. Cloud Big Data Comput., Int. Conf. Cyber Sci. Technol. Congr. (DASC/PiCom/CBDCom/CyberSciTech)*, Oct. 2021, pp. 819–824.

[3] M. Tahir, M. N. Asghar, N. Kanwal, B. Lee, and Y. Qiao, "Joint crypto-blockchain scheme for trust-enabled CCTV videos sharing," in *Proc. IEEE Int. Conf. Blockchain (Blockchain)*, Dec. 2021, pp. 1–6.

[4] E. Alberti, A. Tavera, C. Masone, and B. Caputo, "IDDA: A large-scale multi-domain dataset for autonomous driving," *IEEE Robot. Autom. Lett.*, vol. 5, no. 4, pp. 5526–5533, Oct. 2020.

[5] Y. Lin, C. Tang, F.-J. Chu, and P. A. Vela, "Using synthetic data and deep networks to recognize primitive shapes for object grasping," in *Proc. IEEE Int. Conf. Robot. Autom. (ICRA)*, May 2020, pp. 10494–10501.

[6] Waymo. *Home*. Accessed: Sep. 26, 2022. [Online]. Available: https://waymo.com/

[7] E. Alajrami, H. Tabash, Y. Singer, and M.-T. E. Astal, "On using AI-based human identification in improving surveillance system efficiency," in *Proc. Int. Conf. Promising Electron. Technol. (ICPET)*, Oct. 2019, pp. 91–95.

[8] G.-L. Huang, A. Zaslavsky, S. W. Loke, A. Abkenar, A. Medvedev, and A. Hassani, "Context-aware machine learning for intelligent transportation systems: A survey," *IEEE Trans. Intell. Transp. Syst.*, vol. 24, no. 1, pp. 17–36, Jan. 2023.

[9] Z.-Q. Zhao, P. Zheng, S.-T. Xu, and X. Wu, "Object detection with deep learning: A review," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 30, no. 11, pp. 3212–3232, Nov. 2019.

[10] S. Wu, L. Zhou, Z. Hu, and J. Liu, "Hierarchical context-based emotion recognition with scene graphs," *IEEE Trans. Neural Netw. Learn. Syst.*, early access, Aug. 26, 2022, doi: 10.1109/TNNLS.2022.3196831.

[11] Z. Zhang, B. Chen, and Y. Luo, "A deep ensemble dynamic learning network for corona virus disease 2019 diagnosis," *IEEE Trans. Neural Netw. Learn. Syst.*, early access, Sep. 2, 2022, doi: 10.1109/TNNLS.2022.3201198.

[12] *Road Traffic Injuries*. Accessed: Sep. 11, 2022. [Online]. Available: https://www.who.int/news-room/fact-sheets/detail/road-traffic-injuries

[13] A. Eskandarian, *Handbook of Intelligent Vehicles: A Springer Live Reference*. Berlin, Germany: Springer-Verlag, 2011.

[14] G. Mehr and A. Eskandarian, "Traffic-aware lane change advance warning system for delay reduction at congested freeway diverge areas," *J. Transp. Eng., A, Syst.*, vol. 147, no. 9, Sep. 2021, Art. no. 04021052.

[15] M. M. Rathore, A. Paul, S. Rho, M. Khan, S. Vimal, and S. A. Shah, "Smart traffic control: Identifying driving-violations using fog devices with vehicular cameras in smart cities," *Sustain. Cities Soc.*, vol. 71, Aug. 2021, Art. no. 102986.

[16] *Synthetic Dataset for Accident Detection*. Accessed: Sep. 22, 2022. [Online]. Available: https://www.kaggle.com/datasets/mehwishtahir722/synthetic-dataset-for-accident-detection

[17] W.-J. Chang, L.-B. Chen, and K.-Y. Su, "DeepCrash: A deep learning-based Internet of Vehicles system for head-on and single-vehicle accident detection with emergency notification," *IEEE Access*, vol. 7, pp. 148163–148175, 2019.

[18] Y. Yao, M. Xu, Y. Wang, D. J. Crandall, and E. M. Atkins, "Unsupervised traffic accident detection in first-person videos," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst. (IROS)*, Nov. 2019, pp. 273–280.

[19] B. Kumeda, Z. Fengli, A. Oluwasanmi, F. Owusu, M. Assefa, and T. Amenu, "Vehicle accident and traffic classification using deep convolutional neural networks," in *Proc. 16th Int. Comput. Conf. Wavelet Act. Media Technol. Inf. Process.*, 2019, pp. 323–328. Accessed: Nov. 10, 2023. [Online]. Available: https://ieeexplore.ieee.org/abstract/document/9067530/

[20] G. Rajesh, A. R. Benny, A. Harikrishnan, J. J. Abraham, and N. P. John, "A deep learning based accident detection system," in *Proc. Int. Conf. Commun. Signal Process. (ICCSP)*, Jul. 2020, pp. 1322–1325.

[21] D. Yang, Y. Wu, F. Sun, J. Chen, D. Zhai, and C. Fu, "Freeway accident detection and classification based on the multi-vehicle trajectory data and deep learning model," *Transp. Res. C, Emerg. Technol.*, vol. 130, Sep. 2021, Art. no. 103303.

[22] Z. Zhou, X. Dong, Z. Li, K. Yu, C. Ding, and Y. Yang, "Spatio-temporal feature encoding for traffic accident detection in VANET environment," *IEEE Trans. Intell. Transp. Syst.*, vol. 23, no. 10, pp. 19772–19781, Oct. 2022.

[23] R. E. van Ruitenbeek and S. Bhulai, "Convolutional neural networks for vehicle damage detection," *Mach. Learn. With Appl.*, vol. 9, Sep. 2022, Art. no. 100332.

[24] A. Aggarwal, M. Mittal, and G. Battineni, "Generative adversarial network: An overview of theory and applications," *Int. J. Inf. Manage. Data Insights*, vol. 1, no. 1, Apr. 2021, Art. no. 100004.

[25] L. Li, Y. Lin, B. Du, F. Yang, and B. Ran, "Real-time traffic incident detection based on a hybrid deep learning model," *Transportmetrica A, Transp. Sci.*, vol. 18, no. 1, pp. 78–98, Mar. 2022.

[26] L. Deng, D. Lian, Z. Huang, and E. Chen, "Graph convolutional adversarial networks for spatiotemporal anomaly detection," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 33, no. 6, pp. 2416–2428, Jun. 2022.

[27] S. W. Khan, Q. Hafeez, M. I. Khalid, R. Alroobaea, S. Hussain, J. Iqbal, J. Almotiri, and S. S. Ullah, "Anomaly detection in traffic surveillance videos using deep learning," *Sensors*, vol. 22, no. 17, p. 6563, Aug. 2022.

[28] T. K. Vijay, D. P. Dogra, H. Choi, G. Nam, and I.-J. Kim, "Detection of road accidents using synthetically generated multi-perspective accident videos," *IEEE Trans. Intell. Transp. Syst.*, vol. 24, no. 2, pp. 1926–1935, Feb. 2023.

[29] M. Tahir, Y. Qiao, N. Kanwal, B. Lee, and M. N. Asghar, "Privacy preserved video summarization of road traffic events for IoT smart cities," *Cryptography*, vol. 7, no. 1, p. 7, Feb. 2023.

[30] F. Jiansheng, "Vision-based real-time traffic accident detection," in *Proc. 11th World Congr. Intell. Control Autom.*, 2014, pp. 1035–1038.

[31] H. T. Nguyen, S.-W. Jung, and C. S. Won, "Order-preserving condensation of moving objects in surveillance videos," *IEEE Trans. Intell. Transp. Syst.*, vol. 17, no. 9, pp. 2408–2418, Sep. 2016.

[32] S. S. Thomas, S. Gupta, and V. K. Subramanian, "Event detection on roads using perceptual video summarization," *IEEE Trans. Intell. Transp. Syst.*, vol. 19, no. 9, pp. 2944–2954, Sep. 2018.

[33] B. Maaloul, A. Taleb-Ahmed, S. Niar, N. Harb, and C. Valderrama, "Adaptive video-based algorithm for accident detection on highways," in *Proc. 12th IEEE Int. Symp. Ind. Embedded Syst. (SIES)*, Jun. 2017, pp. 1–6.

[34] V. Mayya and A. Nayak, "Traffic surveillance video summarization for detecting traffic rules violators using R-CNN," in *Advances in Computer Communication and Computational Sciences: Proceedings of IC4S 2017*, vol. 1. Singapore: Springer, 2019, pp. 117–126.

[35] S. Wan, X. Xu, T. Wang, and Z. Gu, "An intelligent video analysis method for abnormal event detection in intelligent transportation systems," *IEEE Trans. Intell. Transp. Syst.*, vol. 22, no. 7, pp. 4487–4495, Jul. 2021.

[36] H. Shingrakhia and H. Patel, "SGRNN-AM and HRF-DBN: A hybrid machine learning model for cricket video summarization," *Vis. Comput.*, vol. 38, no. 7, pp. 2285–2301, Jul. 2022.

[37] T. Kosambia and J. Gheewala, "Video synopsis for accident detection using deep learning technique," in *Proc. Int. Conf. Smart Data Intell. (ICSMDI)*, 2021. Accessed: Oct. 16, 2023. [Online]. Available: https://papers.ssrn.com/sol3/papers.cfm?abstract_id=3851250

[38] H. B. U. Haq, M. Asif, M. B. Ahmad, R. Ashraf, and T. Mahmood, "An effective video summarization framework based on the object of interest using deep learning," *Math. Problems Eng.*, vol. 2022, pp. 1–25, May 2022.

[39] A. Pramanik, S. K. Pal, J. Maiti, and P. Mitra, "Traffic anomaly detection and video summarization using spatio-temporal rough fuzzy granulation with Z-numbers," *IEEE Trans. Intell. Transp. Syst.*, vol. 23, no. 12, pp. 24116–24125, Dec. 2022.

[40] N. Saxena and M. N. Asghar, "YOLOv5 for road events based video summarization," in *Intelligent Computing* (Lecture Notes in Networks and Systems), vol. 739, K. Arai, Ed. Cham, Switzerland: Springer, 2023, pp. 996–1010, doi: 10.1007/978-3-031-37963-5_69.

[41] H. Shahwani, S. A. Shah, M. Ashraf, M. Akram, J. P. Jeong, and J. Shin, "A comprehensive survey on data dissemination in vehicular ad hoc networks," *Veh. Commun.*, vol. 34, Apr. 2022, Art. no. 100420.

[42] *BeamNG—YouTube*. Accessed: May 29, 2022. [Online]. Available: https://www.youtube.com/user/beamng

[43] Car Crashes Time. (Jan. 8, 2018). *CCTV Car Crashes Compilation 2018 #EP. 20*. Accessed: Aug. 23, 2022. [Online]. Available: https://www.youtube.com/watch?v=gQkoujWBxqg

[44] M. N. Asghar, N. Kanwal, B. Lee, M. Fleury, M. Herbst, and Y. Qiao, "Visual surveillance within the EU general data protection regulation: A technology perspective," *IEEE Access*, vol. 7, pp. 111709–111726, 2019.

[45] I. Aribilola, M. N. Asghar, N. Kanwal, M. Fleury, and B. Lee, "SecureCam: Selective detection and encryption enabled application for dynamic camera surveillance videos," *IEEE Trans. Consum. Electron.*, vol. 69, no. 2, pp. 156–169, May 2023.

MEHWISH TAHIR (Graduate Student Member, IEEE) received the bachelor's and master's degrees in computer science from Lahore College for Women University, Lahore, Pakistan, in 2013 and 2017, respectively. She is currently pursuing the Ph.D. degree in computer science with the Software Research Institute (SRI), Technological University of the Shannon: Midlands Midwest, Athlone, Ireland. She is also working on event-based video summarization and encryption techniques to secure the visual privacy of large-scale CCTV systems. Her research interests include deep learning, computer vision, encryption, visual privacy, video summarization, smart cities, and blockchain.

YUANSONG QIAO (Member, IEEE) received the Ph.D. degree in computer applied technology from the Institute of Software, Chinese Academy of Sciences, Beijing, China, in 2008. He is currently a Senior Research Fellow with the Software Research Institute (SRI), Technological University of the Shannon: Midlands Midwest, Ireland. He is also a Science Foundation Ireland (SFI) Funded Investigator with the SFI CONFIRM Smart Manufacturing Centre. His research interests include future internet architecture, blockchain systems, robotic control and coordination, and edge intelligence and computing. He is a member of IEEE (Communications, Computer and Robotics and Automation Societies and Blockchain Community) and ACM (SIGCOMM and SIGMM).

NADIA KANWAL (Senior Member, IEEE) received the master's and Ph.D. degrees in computer science from the University of Essex, U.K., in 2009 and 2013, respectively. She also received a prestigious Marie Skodowska-Curie Research fellowship (for three years), in 2019, to do an industry-led project related to secure and privacy-protected CCTV video storage and retrieval as a Principal Investigator. Her research interests include computer vision and machine learning. She is actively working on applying machine learning to develop cutting-edge solutions for health, security, and vision applications. Her publications on multimedia data security, visual privacy, low-level image features, virtual reality, EEG/ECG signal analysis, and pupillometry have been very well supported by the research community. She is an active reviewer of good-standing journals and conferences.

BRIAN LEE received the Ph.D. degree from the Trinity College Dublin, Dublin, Ireland, in the application of programmable networking for network management. He has over 25 years research and development experience in telecommunications network monitoring, their systems, and software design and development for large telecommunications products with very high-impact research publications. He was the Director of Research with LM Ericsson, Ireland, with responsibility for overseeing all research activities, including external collaborations and relationship management. He was the Engineering Manager of Duolog Ltd., where he was responsible for the strategic and operational management of all research and development activities. He is currently the Director of the Software Research Institute, Technological University of the Shannon: Midlands Midwest, Ireland.

MAMOONA N. ASGHAR (Senior Member, IEEE) received the Ph.D. degree from the School of Computer Science and Electronic Engineering, University of Essex, Colchester, U.K., in 2013. In 2017, she was awarded an industry-led Marie Sklodowska-Curie (MSC) Career-Fit Postdoc Research Fellowship (three years with TUS-Ireland) funded by Marie-Curie Actions and Enterprise Ireland. She has published several ISI-indexed journal articles along with numerous international conference papers. Her research interests include security aspects of multimedia (image, audio, and video), compression, visual privacy, encryption, steganography, secure transmission in future networks, video quality metrics, key management schemes, computer vision algorithms, deep learning models, and general data protection regulation (EU-GDPR). She is actively involved in reviewing research articles for renowned journals/conferences and has participated as the session chair.