International Journal of Engineering Research and Sustainable Technologies
Volume 2, No.2, June 2024, P 36-44
ISSN: 2584-1394 (Online version)

REAL TIME ACCIDENT DETECTION AND REPORTING SYSTEM USING CNN ALGORITHM

Barathwaj K[1], Ganesh Kumar H[2], Charan MJ[3], P. Dineshkumar[4], T. Kirubadevi[5]
[1,2,3,4,5] Department of CSE, Dr.M.G.R.Educational and Research Institute, India
*** Corresponding author email address**:-dineshkumar.it@drmgrdu.ac.in

https://doi.org/000000/000000/

Abstract

In order to identify patterns for decision-making, deep learning, a branch of artificial intelligence, imitates how the human brain processes data. Within machine learning, it utilizes networks that discern and categorize patterns from unstructured or untagged data. Referred to as deep neural learning, Convolutional Neural Networks (ConvNets or CNNs) shine in image recognition tasks, adept at identifying faces, objects, and traffic signs. Their prowess extends to robotics, enhancing vision for self-driving cars. Despite widespread awareness of driving regulations, a substantial number globally fall victim to vehicle crash injuries due to drivers' negligence, despite their knowledge. This paper contributes to road accident detection through the Mask R- CNN method, aiming to improve safety measures.

Keywords: *Machine Learning, Accident Detection, Feature Extraction, Image Classification, Twilio, SMSalert, CNN Algorithm.*

1. Introduction

The central ambition revolves around integrating an advanced system designed to discern accidents within video footage captured by surveillance cameras. The core aim is to swiftly identify incidents, enabling immediate alerts to relevant authorities, thereby facilitating prompt assistance to individuals involved in accidents. This cutting-edge system relies on sophisticated Deep Learning Algorithms, specifically harnessing the power of Convolutional Neural Networks (CNNs) to meticulously analyze frames extracted from the video stream sourced from these cameras. Its primary objective is to accurately pinpoint and classify accidents within mere seconds of their occurrence, maximizing the efficiency of emergency responses and aid.

The strategic focus of this system implementation primarily aligns with highways, acknowledging the inherent challenges associated with the timely arrival of aid to accident sites due to sparse traffic and the remoteness of some locations. The intention is to address the critical gaps in response time and support mechanisms, particularly in scenarios where victims urgently require help. By deploying this system along these highways, it endeavors to act as a vigilant and proactive mechanism, significantly reducing the time between an accident's incidence and the commencement of necessary assistance. For effective coverage and surveillance, the deployment plan involves strategically positioning CCTV cameras at intervals of approximately 500 meters along parkways. These cameras, acting as observation nodes, serve as the primary data source for the proposed accident recognition framework. Integrated within these cameras, the system actively processes the recorded footage, employing its robust accident detection model to swiftly identify and flag any incidents that warrant immediate attention. By focusing on this strategic positioning along highways and leveraging the capabilities of these surveillance cameras as a conduit for observation, the system seeks to revolutionize response mechanisms. Through the fusion of cutting-edge technology and proactive surveillance, it aims to significantly improve the responsiveness and efficacy of emergency services, ensuring timely aid and support for accident victims in critical situations.

2. Related Works

[1 In today's fast-paced world, accidents often occur without immediate attention. We solve this by utilizing mobile technology based on microcontrollers and coupled with LabVIEW to create an Enhanced Accident Detection System using biomedical smart sensors. The accident location and the victims' physiological parameters, such as body temperature, heart rate, and coma stage recovery status, are transmitted via this system in a timely manner, via SMS to emergency care centers. This prompt communication aims to reduce fatalities by expediting emergency response. Additionally, individuals involved in minor accidents can use a switch to indicate the incident's limited severity, preventing unnecessary alarming SMS transmissions. This

system strives to improve accident response, aiding emergency services and potentially saving lives by providing accurate and timely victim information.

[2] In today's world, traffic accident fatalities remain a significant concern. To effectively combat this issue, there's a crucial need for an automatic traffic accident detection system that rapidly alerts emergency responders. While existing methods utilize built-in vehicle systems, they pose challenges due to their cost, maintenance complexities, and limited availability across vehicles. However, recent advancements in smartphone technology, including enhanced processing power and sensor capabilities, have paved the way for smartphone-based accident detection systems. Typically, high-speed collision detection systems based on smartphones use GPS to determine the vehicle's speed and the accelerometer's G-Force readings. It's important to remember, though, that a significant percentage of traffic incidents happen at slow speeds, high lighting the necessity for systems that can detect accidents in such scenarios. This paper focuses on addressing this by incorporating low-speed car accident detection into smartphone-based systems. One significant challenge in low- speed accident detection is differentiating between a user inside a vehicle and someone walking or running slowly outside. A system that separates the speed changes typical of low-speed vehicles from those of walkers is introduced in this study to help alleviate this. The two phases of this proposed system's operation are the detection phase, which detects auto accidents at both low and high speeds, and the notification phase, which immediately notifies emergency responders of the location of the accident and provides them with detailed information, including images and videos, to enable prompt intervention and recovery.

[3] Speed is a major contributing factor in auto accidents. If emergency personnel had responded promptly after learning of the disaster, many lives might have been saved. The goal of this project is to develop an accident detection system that can identify many components and notify the rescue team when an accident happens. Preserving precious human life requires an effective automatic accident detection system that can automatically notify emergency services of the accident site. Accident detection and alerting are the focus of the proposed system. It locates the car involved in the collision with exact latitude and longitude, then sends this information to the closest emergency response provider. Real-time accident detection and rescue crew alerts are the goals of the project.

[4] Roadway pothole detection technology approaches are being developed to offer actual offline vehicle control (for automated driving or vehicular applications) or offline data gathering for road repair. Pothole-detecting technology is an important field of study because of its potential to improve highway safety and reduce the cost of road repair. Deep learning algorithms are among the most promising techniques being studied by researchers to identify potholes on roadways. Deep learning techniques use artificial neural networks to identify patterns and features in vast volumes of data. This technique has been effectively applied in a number of applications, such as object identification and picture recognition. The same techniques can be used to identify potholes in road pictures. One such technique is the use of convolutional neural networks (CNN). Due to their architecture, which allows them to process images by learning pertinent features and patterns, CNNs have been shown to perform with good accuracy in a wide range of image identification applications. Because of these characteristics, researchers worldwide have been examining various techniques for identifying potholes on roads. Roads contribute significantly to the economy and are necessary for mass transit. Potholes in the roads are a key cause of concern for the transportation networks. Automating pothole identification has been suggested by numerous research to use deep learning algorithms, which deal with various image analysis and object detection techniques. The automated pothole detecting technique must have the maximum degree of precision and consistency. Practical testing of this system in a simulated environment demonstrated highly promising performance results. By incorporating low-speed accident detection into smartphone technology and facilitating immediate communication of comprehensive accident data to emergency responders, this system stands to significantly reduce the time between accident occurrence and emergency response, potentially saving lives in critical situations.

3. Methodology

3.1 Model B+ Raspberry Pi 3

To help enthusiasts of programming learn the basics of computer science at a minimal cost, the UK-based Raspberry Pi foundation created the tiny Raspberry Pi 3 Model B+ portable computer. A 64-bit quad-core processor operating at 1.4 GHz, four USB ports, faster Ethernet, Bluetooth 4.2, and 1GB of RAM are all included with the Raspberry Pi. The Raspberry Pi can easily integrate the required components thanks to its 40 GPIO (General Purpose Input Output) ports. Though the built-in hard drive's massive storage capacity is

absent, it does include a microSD card that may be utilized for light storage and app booting.

3.2 Pi Camera

The Pi Camera, designed exclusively for Raspberry Pi single-board computers, stands out as a compact and adaptable imaging accessory. Its small form factor allows direct connection to the dedicated camera port on the Raspberry Pi, offering excellent image and video capture capabilities.

Depending on the model, it may come with either a fixed-focus lens or support for interchangeable lenses, providing users with enhanced versatility. Leveraging the computational power of the Raspberry Pi, the camera seamlessly integrates for on-board image processing and manipulation. Widely applied in diverse projects such as video streaming, surveillance, computer vision, and photography, the Pi Camera, with its user-friendly programming interfaces and compatibility with languages like Python, proves to be a valuable asset for enthusiasts and developers seeking to incorporate imaging functionalities into their Raspberry Pi endeavors.

4. Software Used

4.1 Keras

It is a basic, open-source library written in Python, Keras was created to enable broad deep neural network research. Developed and maintained as part of the ONEIROS project, it is compatible with well-known Python libraries like Theano and TensorFlow. Francois Chollet is the person behind it. Its smooth operation atop other computing backends is a notable characteristic. With its logically built high-level abstractions, Keras makes it easier to create and develop deep learning models without requiring a particular computational backend. Keras, which is positioned as an interface rather than a stand-alone machine learning framework, makes it easier to deal with text and image data by providing implementations of key components of neural networks, including as layers, optimizers, and activation functions.

4.2 CV2

A popular and adaptable open-source computer vision and machine learning software library is called OpenCV (Open Source Computer Vision Library). Since its initial development by Intel in 1999, the project has been directed by the community. With a strong foundation for image and video processing, OpenCV is intended to offer a full suite of tools for real-time computer vision applications. It is usable by a wide range of developers and researchers due to its support for numerous programming languages, such as C++, Python, and Java. Many functions are covered by OpenCV, such as the manipulation of images and videos, object detection, facial recognition, feature extraction, and machine learning integration. It is particularly popular in fields such as robotics, augmented reality, and medical image analysis. OpenCV's modular structure enables users to seamlessly integrate its components into various projects, fostering innovation in computer vision applications. Constantly evolving, OpenCV has undergone numerous updates, with the community actively contributing to its development. Its rich documentation, tutorials, and user forums make it a valuable resource for both beginners and experienced developers seeking to implement computer vision solutions across different domains.

4.3 Twilio

Developers may include text messaging, voice calls, and chat into their applications with the help of Twilio, a cloud communications platform that offers APIs. It makes communication functionality simpler so that developers may concentrate on creating their apps rather than worrying about running complicated infrastructure. offering services like voice calling, SMS, and video functionality through simple and accessible APIs. Developers can leverage Twilio's APIs to integrate features such as programmable messaging, voice over IP (VoIP), and real-time video into their web and mobile applications. This enables businesses to create customized and scalable communication solutions without the need for extensive infrastructure. One of Twilio's key strengths is its versatility; it supports multiple programming languages and has extensive documentation and tutorials. Businesses use Twilio for various purposes, including customer engagement through automated messaging, two-factor authentication, and building complex communication workflows. Twilio's innovative approach empowers developers to enhance user experiences by seamlessly integrating communication features. Its pay-as-you-go pricing model and ease of use have contributed to its widespread adoption across industries, from startups to large enterprises, making Twilio a

go-to solution for businesses looking to implement robust and flexible communication capabilities in their applications across different domains.

5. Implementation

CNNs are employed in the modeling of spatial data, such as pictures. CNNs have demonstrated efficacy in several applications such as object detection and image categorization. Sequential data is modeled and predictions are based on LSTMs. LSTMs are frequently utilized in language modeling, sequence generation, text classification, and other fields. When dealing with sequential data that has spatial input, standard LSTMs can be applied straight away. Thus, a CNN- LSTM architecture must be employed to complete jobs involving image or video sequences. The proposed model combines CNN and LSTM layers for continuous classification of recorded video from a camera. Convolution Network to take into account features that are both spatial and temporal. This is isolated from the model's convolution and recurrent sections.

The CNN is used in a CNN-LSTM network to extract features from the images, which are subsequently transmitted to the LSTM for sequence prediction. Common uses for them include activities like activity recognition, image and video description, and so forth.

5.1 Convolution Layer

The convolution layer is the first stage in extracting useful information from images. Convolution helps to retain the relationship between the pixels in an image by extracting visual qualities from small squares of input data. This mathematical technique uses two inputs: a kernel, or filter, and a piece of the image as a matrix.

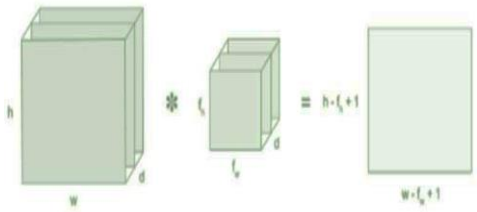

Fig. 1. Image matrix times filter or kernel matrix

Figure 1 displays a dimensioned picture matrix. ($h \times \mathrm{w} \times d$), A filter that outputs the dimension ($h - fh +$ 1) is ($fh \times f\mathrm{w} \times d$). × (w − fw + 1) × 1. It is a picture of a 3 × 3 filter matrix convolving a 5 × 5 image matrix. The resulting matrix, known as the feature map, is a 3× 3 matrix. When multiple filters are applied to an identical image, distinct information can be extracted from it. It can be applied to the extraction of spatial data, including blur and edge detection.

5.2 Pooling Layer

The main purpose of pooling layers is to reduce the number of parameters when the input image is too large. Alternatively called subsampling or down sampling, spatial pooling keeps the most relevant data while reducing each feature map's dimensionality. Three primary categories of pooling layers exist generally:

Sum Pooling , Maximum Pooling , Average Pooling

5.3 Completely Networked Layer

We transform our grid into a vector and apply a layer akin to a basic neural network in this layer.

5.4 Long Short Term Memory

Units of an RNN are called LSTM units. A cell, an input gate, an output gate, and a forget gate comprise an LSTM unit. Over time, the cell retains its values, and the three gates facilitate the control of information entering and leaving the cell.

6. Evaluation of Inception V3

Inception v3 has been used as the industry standard for picture classification for a number of years. Inception v3 receives the lowest error rate of all the benchmark CNN models for picture categorization, according to a quick comparison as displayed in Table 1.

Table 1. Comparisons of different CNN models' errors using the Image net dataset

Model	**Rate of Errors**
AlexNet	15.3%
Inception (Google Net)	6.67%
Inception v2	4.9%
Inception v3	3.46%

Inception v3 uses a diverse set of convolutions, unlike traditional neural networks. As a result, the model is able to extract more features and explore the image more thoroughly. The notion "Why not do it all?" led to the development of Inception models. The decision of which layers in a convolutional neural network to leave out is usually unclear. Occasionally, other filter sizes also appear to function very well. Our goal with the Inception architecture is to place multiple convolutional filters with varying dimensions and pooling layers in one network layer, letting the model select the optimal one. This greatly increases the classification margin but also makes the model substantially more complex than a basic CNN. Figure 3's architecture serves two purposes.

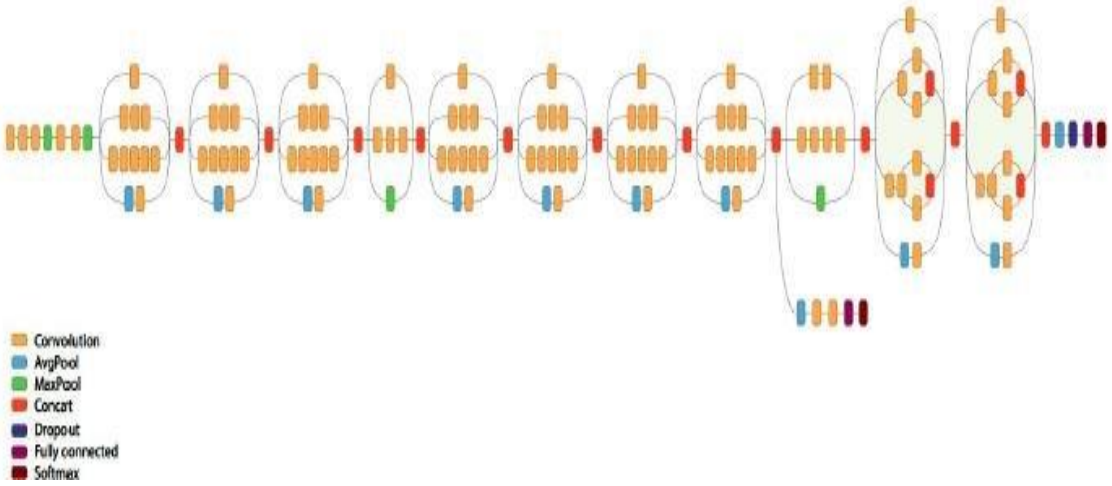

Fig. 2. Schematic Representation of Inception v3

Initially, the smaller convolutions are employed to effectively retrieve the fundamental local information. Second, it makes use of the larger convolutions to get the more intricate abstracted characteristics. Only 1 × 1, 3 × 3, and 5 × 5 filter sizes were allowed in the model. Convolutions can be somewhat expensive, thus dimensionality reduction and memory conservation can be accomplished by running a 1 × 1 convolution prior to the 3 × 3 or 5 × 5 convolutions themselves.

The vanishing/exploding gradients are a common problem for deep CNNs. Consequently, From Figure 2, Inception v3's auxiliary classifiers are employed as a regularizer. Computed ultimate loss is the mean of all

losses from each softmax layer in the model. To ensure that the output dimensions for filter concatenation after each inception phase remain the same, "same convolutions" have been used continuously throughout the model.

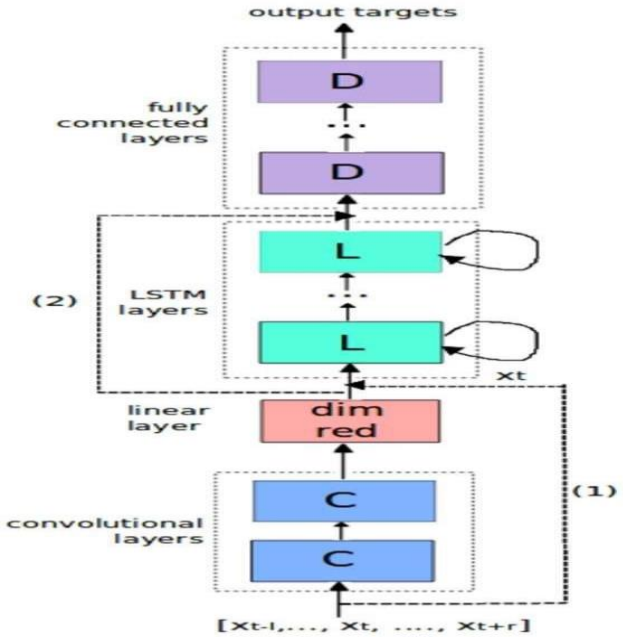

Fig. 3. Architecture of proposed model

The 256 hidden LSTM units in each of the two layers that make up the LSTM portion of the model are followed by dense layers, as illustrated in Figure 3. Every deeper network was outperformed by this shallow network. A dropout layer was inserted between each of these layers to avoid overfitting. In addition, the data was converted into a 4D tensor prior to being fed into the LSTM.As mentioned earlier, our primary instrument for creating the proposed model has been the Inception v3.

The final bottleneck layers were removed after the convolution and pooling layers, making room at the end of the network for further layers like the LSTM and dense layers. Rather of feeding our softmax predictions into the LSTM layers, our technique uses the output of the final pool layer, which consistently gives us a greater accuracy when we train it. Our last pool layer gives us an image feature representation rather than actual probability. Every frame of our movie is passed through Inception v3 and then we save the output from the network's final pool layer. We basically take out the top classification part of the network to obtain a 2,048-d vector of features from the final pool layer that we can input into our RNN.

We then convert those recovered features into sequences of extracted features. Rather than feeding our images through the CNN every time we wish to train a new network architecture or get a new sample, we aggregate the sampled frames from our video, store it on disk, and utilize this to train several RNN models. We add each frame to a queue of size N in order to achieve that, loop through each frame in chronological sequence, and then pop off the first frame that we added. For this experiment, we used five time-steps, or a queue of size five, for our LSTM layers. 5 was used so that we could get a prediction every second because the Pi Camera records videos at a frame rate of 5 frames per second. This has been demonstrated in action in the Figure 3. The Figure 4 illustrates the feeding sequences of LSTM layer frames. Also Figure 5 and Figure 6 illustrates the losses in training and validation as well as the accuracy.

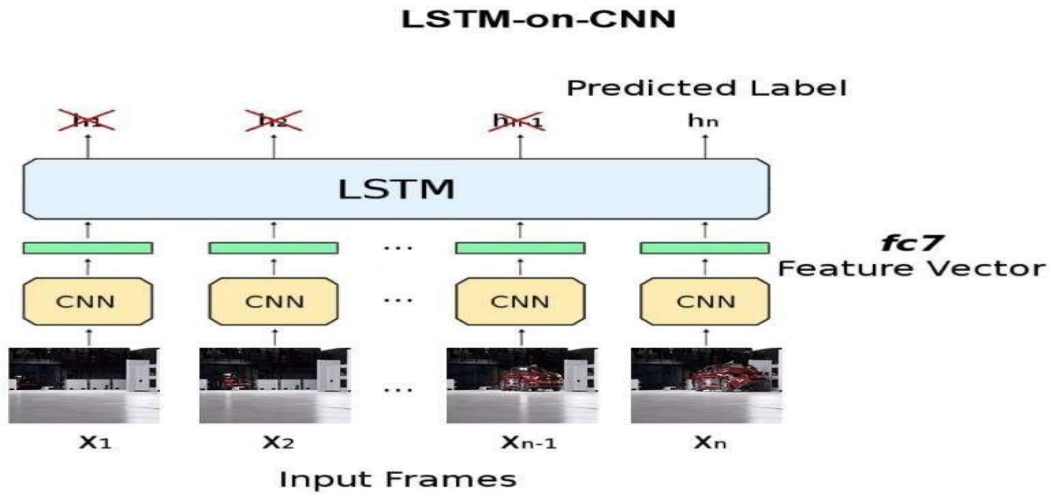

Fig. 4. Feeding Sequences of Frames to the LSTM Layer

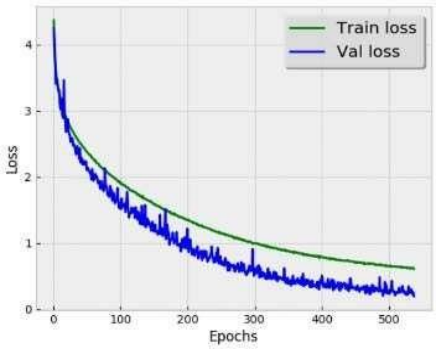

Fig. 5. Losses in training and validation

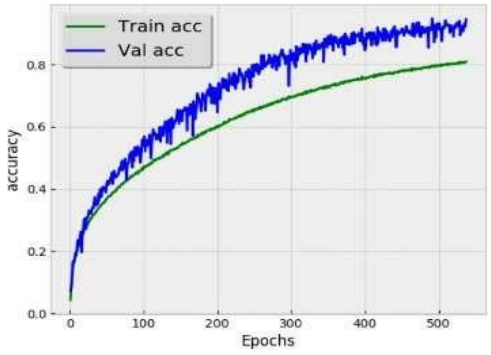

Fig.6.Both training and validation accuracy

Just the LSTM and fully connected layers of the Inception v3 were trained using our images; the convolution layers' weights were frozen before training. As mentioned earlier, these layers were followed by a sigmoid layer for the final classification. The picture augmentation feature of Keras, the picture Data Generator, was used to train each of the layers listed in Table 2. The method of using a set of images to artificially enlarge the dataset is called image augmentation. The most often used features in image augmentation are shear, zoom, and pre-processing functions. These settings essentially provide random shear, rotation, and zoom on the current images, providing the model with additional information to learn from. The following values were employed in the suggested model:
Rescale=1/255, shear_range=0.2 , zoom_range=0.2, flip_horizontal=true. Every training image underwent 500 epochs of training. The following were the outcomes attained Accuracy of Training: 0.95, Validation Precision: 0.85 , Exercise Loss: -0.2568 , Loss of Validation: -0.2894.

7. Results and Discussion

As soon as it identifies an accident, the system sends a message via the Twilio module. Once it's running, it applies the recommended model to each frame of video it receives from the Pi-camera after analyzing each one. It also transfers the frame at which an accident was detected together with the size of the accident. The time stamp of the accident's detection is also shown. The accident frame and associated data are shown in Figure 7, and upon detection, an alert message informing the registered number of the occurrence is sent. In Figure 8, it is mentioned.

Table 2: The Proposed Model's Layers

Type	Depth	#x1	#3X3 reduce	#3X3	#5X5 reduce	#5X5
Convolution	1					
Max pool	0					
Convolution	2		64	192		
Max pool	0					
Inception 3a	2	64	96	128	16	32
Inception 3b	2	128	128	192	32	96
Max pool	0					
Inception 4a	2	192	96	208	16	48
Inception 4b	2	160	112	224	24	64
Inception 4c	2	128	128	256	24	64
Inception 4d	2	112	144	288	32	64
Inception 4e	2	256	160	320	32	128
Max pool	0					
Inception 5a	2	256	160	320	32	128
Inception 5b	2	384	192	384	48	128
Avg pool	0					
Dropout 40 %	0					
Dense Layer (linear)	1					
Bidirectional LSTM	256					
Dropout 20 %	0					
Bidirectional LSTM	256					
Dropout 20 %	0					
Dense Layer (linear)	64					
Dense Layer (linear)	32					

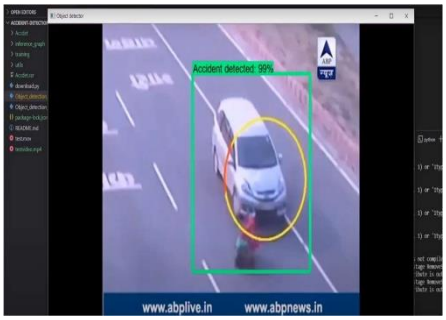

Fig. 7. Monitoring an Event in Real Time with a Camera

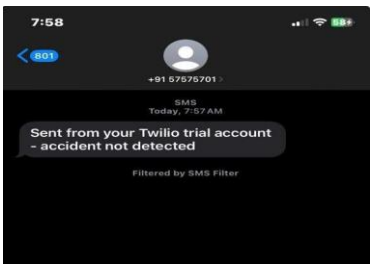

Fig. 8. Reporting after Accident Detection

The following "Accuracy" statistic was used to assess the model's performance:

Accuracy = (Number of Correct Prediction) / (Total of all cases to be predicted)

The accuracy of the running model ranged from 82% to a maximum of 98.76%. The model's accuracy was higher than 92% on average. The Mask R-CNN model emerges as a promising approach for Accident Detection, showcasing remarkable accuracy in identification. Its Intersection over Union concept proves invaluable in spotting anomalies on the road, aiding bystanders around the accident site. When integrated with an efficient Response system, this model holds potential in mitigating traffic congestions, thereby saving crucial time. Additionally, prompt detection not only facilitates quicker alerts to nearby hospitals but also enables swift communication with the families of those impacted, underscoring its significance in timely emergency response.

References

1. Prabakar, S., et al. "An enhanced accident detection and victim status indicating system: Prototype." India Conference (INDICON), 2012 Annual IEEE. IEEE, 2012.
2. "SOSmart automatic car crash detection and notification app", SOSmart automatic car crash detection app, 2019. [Online]. Available: http://www.sosmartapp.com. [Accessed: 07- Mar- 2019]. 2019
3. K.Padma Vasavi, "Real-Time Accident Detection and Intimation System using Deep Neural Networks, pp197-205,2022
4. Rohan Chorada, Hitesh Kriplani,Biswaranjan Acharya,' CNN-based Real-time Pothole Detection for Avoidance Road Accident', ICICCS,IEEE Xplore 2023.