JETIR.ORG ISSN: 2349-5162 | ESTD Year : 2014 | Monthly Issue
JOURNAL OF EMERGING TECHNOLOGIES AND INNOVATIVE RESEARCH (JETIR)
An International Scholarly Open Access, Peer-reviewed, Refereed Journal

NIGHT-TIME ACCIDENT DETECTION USING CCTV AND REPORTS TO THE NEAREST HOSPITALS AND FIRE STATIONS

[1]Lakshmi Madhu, [2]Dr. Revathi MCA, M.Phil, Ph.D.
[1]MSc Data Science and Business Analysis, [2]School of Computer Science
Rathinam College of Arts and Science
India

Abstract

Nowadays, Road accidents have been on the rise due to the rapid growth of population and vehicles. Nighttime accidents have resulted in numerous casualties and permanent disabilities. In India, 40% of road accident fatalities occur due to being unattended and excessive blood loss and 10% are due to fire exposures. The proposed system efficiently overcomes the above problems. The proposed system consists of two phases: "Accident detection using the GoogLeNet model of CNN architecture" from CCTV and "Reporting to the nearest hospitals and Fire Stations" through alert messages. The proposed system alerts nearby hospitals and fire stations using the CCTV's IP address and the longitude and latitude of the hospital's location.

Keywords: Road Accidents, CCTV, fire stations, hospitals, GoogLeNet CNN Model

1. Introduction

The population increase is multiplying, especially in India it's become the most populated country in the world. The rate of accidents is also added as follows. Every time India loses 1.5 million lives in road accidents. According to the Ministry of Road Transport and Highways, every time, roughly 1.5 lakh people die on Indian roads, rephrasing, on average, into 1130 accidents and 422 deaths every day. Nearly 29% of these accidents occurred after dark, especially at midnight. The main reasons for the nighttime accident deaths are the excess blood loss due to being unattended and without getting the proper medical care at the right time. The fear of the people to face the nocturnal accidents is also a reason for the accident deaths.

There are some existing systems like accident detection based on a simple vibration sensor as the heart of this system and makes use of message queuing telemetry transport (MQTT) protocol [1], detecting accidents from an accelerometer, and the vibration sensor [2], vehicle crash detection system using Arduino, GPS, GSM, and accelerometer, and the alert system [3] [4], accident detection and alert system using an android application [5] and accident detection using the attached accelerometer in the vehicle and the heartbeat sensor on the user's body [6]. However, after the analysis, all these existing systems had a very common and important disadvantage i.e. time. Proper medical care at the proper time is the major factor that decides life, especially in nocturnal accidents. If the accident can be recognized in the least time, it will help to save the life of the victim. Some other types of advanced methods used in the existing system include accident-prone systems using YOLO [7], road accident detection using machine learning [8], a heuristic method [9], IOT-Based Accidental Detection System (ADS) Using Raspberry Pi [10], IOT Based Road Accident Detection and Prevention System [11] [12] [13] and Accident Detection Using Convolutional Neural Networks [14]. After the evaluation of these systems, they failed to find the new shapes of the vehicles after accidents and also failed to give an efficient accuracy according to modern technology.

This paper proposes a web app that becomes an efficient solution for the above problems. It consists of formerly installed CCTV camera footage that is used to describe accidents in real time. This is done by pre-training a DL model with a dataset containing frames labeled accident and non-accident. The model uses this knowledge to describe accidents and get the position via surveillance cameras fixed in the neighborhoods. The camera's IP address is used to get the exact position of the accident, a new point in India, and corresponding cautions are transferred to the most near hospitals, in the case of fire, cautions are transferred to the most near fire stations also. The alert includes the latitude and longitude of the accident and a request to shoot ambulance services from the nearest hospitals and fire station services to the position.

2. Literature Survey

Adith Jose et al. (2021) proposed an accident detection and warning system. The article suggests a new way to use existing CCTV cameras with night vision, without the need for any extra sensors [15]. Using real-time video processing, the proposed device can identify accidents and quickly notify emergency services such as ambulances and fire departments. This could help save lives by ensuring that help arrives quickly.

Akanksha A Pai et al. (2023) proposed a method named 'Real Time Accident Detection from CCTV and Suggestion of Nearest Medical Amenities' [16]. This article proposes a real-time accident detection and alerting system using CNN models with ReLU and Softmax activation functions. The system assumes that visual data in a temporary sequence correspond to traffic accidents. It consists of two phases: visual feature extraction and temporal pattern detection. Convolution and recurrent layers learn visual and temporal features from scratch and publicly available datasets. The model achieved an accuracy of 96% in detecting the accidents.

C. K. Gomathy (2022) described accident detection and alert systems using GPS and GSM [17]. This system aims to notify nearby medical centers of accidents so that they can provide immediate medical assistance. An accelerometer attached to the vehicle detects the vehicle's tilt, while a heartbeat sensor on the user's body detects abnormal heartbeat patterns to assess the severity of the accident. The system will make a decision based on the data from the accelerometer, GPS, and GSM modules, and send the information to the smartphone. The Android app on the smartphone will then send text messages to the nearest medical center and the user's friends. The app also shares the exact location of the accident, which can save time.

Dr. Bharat Naresh Bansal and V. Garg (2021) conducted A Review paper on "Vehicle Accident Detection, Tracking, and Notification Systems"- A Comparative Study [18]. This paper reviews existing accident detection systems for vehicles, assessing their strengths, weaknesses, and potential for further development. Upon detecting an accident, these systems typically leverage sensors like accelerometers or vibration sensors to automatically trigger notifications. Alerts, containing both an SMS/email message and the accident location (acquired through GPS), are then sent to pre-registered contacts such as family members, emergency services, or nearby hospitals. Various technologies like GSM and Wi-Fi facilitate communication and data transmission within these systems.

E. Mohanraj, M. Dakshnamoorthy, and S. Karthikeyan (2022) developed "Accident Prevention using IoT" [19]. This project aims to enhance road safety by preventing accidents caused by drunk driving and detecting collisions. It utilizes an alcohol sensor to detect the presence of alcohol in the driver's breath and disable the ignition if intoxication is detected, thus deterring potential accidents. Additionally, a vibration sensor monitors for sudden jolts, indicating a possible collision. Upon detecting an accident, the sensor transmits data to a microcontroller, which analyzes it to confirm the event. All relevant vehicle information, including accident detection and possible intoxication, is then sent to a Telegram application using a dedicated bot, facilitating timely emergency response.

G. Elumalai, O. S. P. Mathanki, S. Swetha (2015) described "Vision-Based Intelligent Traffic Analysis System for Accident Detection and Reporting System" [20]. This system acts as a vigilant guardian at accident-prone intersections, equipped with the ability to (1) spot potential collisions, (2) capture key moments, and (3) alert relevant authorities. It achieves this by employing a "picture queuing" technique, where snapshots of vehicles are extracted from video footage captured by specialized cameras. By focusing on rapid changes in speed, position, direction, and angle of moving vehicles, the system's sophisticated algorithm can identify situations with high accident risk. These captured images serve as valuable substitutes for traditional crash data, offering detailed insights for further analysis and prevention efforts.

Gupta & Jain (2019) developed a method for accident detection using convolutional neural networks (CNNs) [21]. The paper proposes a method of classifying each frame of a video as an accident or non-accident using a trained CNN model. Convolutional neural networks learn visual features from small patches of an image while preserving the spatial relationships between pixels. Pooling layers reduce the number of parameters in the network when the input image is large. The fully connected layer converts the grid of features into a vector, which is then fed into a layer that resembles a standard neural network. Inception v3 is a benchmark model for image classification that uses a variety of convolution operations compared to traditional convolutional neural networks. This enables the model to examine the image more closely and identify features.

Ijjina et al. (2019) proposed a computer vision-based accident detection in traffic surveillance [22]. This paper proposes a new approach to detecting road accidents using modern techniques. The proposed framework uses Mask R-CNN to accurately detect objects in surveillance footage, followed by an effective centroid-based algorithm to track those objects. The probability of an accident is calculated by analyzing the speed and trajectory abnormalities of a vehicle after colliding with other vehicles. The framework achieves high accuracy in detecting accidents and minimizing false alarms on standard road traffic CCTV surveillance footage. With a detection rate of 71% and a false alarm rate of only 0.53%, the system is highly reliable and effective.

Mohammed Ahmed et al. (2023) created a new computer vision system that can detect and classify traffic accidents in real-time, even in challenging conditions such as sparse or dense traffic, low visibility, and varying weather [23]. The system has five parts. The first part uses the YOLOv5 algorithm and DeepSORT tracker to detect and track vehicles in real time with an accuracy of 99.2%. The second part is the most important part of the system. It uses the YOLOv5 algorithm to efficiently detect and classify the severity of accidents with an accuracy of 83.3%. The third model uses the ResNet152 transfer learning algorithm to classify fires that occur after traffic accidents, achieving an accuracy of 98.955%. Overall, this new computer vision-based system shows great potential for improving traffic incident detection and response, especially in difficult conditions.

Ojsadmin (2017) introduced "Automatic Vehicle Accident Detection Based on GSM System" [24]. This paper presents a budget-friendly approach for automatically detecting traffic accidents using the GSM network. The proposed system aims to minimize fatalities by leveraging vibration sensors and GSM communication. It integrates hardware components (circuits) with software to process data and display information on a user-friendly interface built with LabView. The system employs vibration sensors mounted on both sides of vehicles to detect unusual movements indicative of an accident. Upon detecting such an incident, an SMS alert is automatically sent to a pre-defined recipient. This notification can expedite search and rescue efforts, potentially saving lives.

Praharsha Sarma, Utkarsh Kumar, C.N.S. Vinoth Kumar, and M.Vasim Babu(2020) proposed "Accident Detection and Prevention Using IoT & Python Opencv" [25]. This project aims to solve the problem of accident detection by leveraging the Internet of Things (IoT) and computer vision technology. Using ultrasonic sensors, the system detects potential obstacles around the vehicle. Additionally, vibration and accelerometer sensors are attached to the axle monitor for unusual movements. Upon detecting an accident, the Arduino Uno microcontroller triggers a series of automated safety measures. The system transmits an emergency message with the vehicle's location (longitude and latitude) via GSM and GPRS modules, facilitating a swift response. To ensure rapid occupant egress, it automatically unlocks doors, unbuckles seatbelts, and activates hazard lights. This proactive system continuously monitors sensor values and reacts based on pre-defined thresholds programmed into the Arduino platform.

Pranali, Ulhas Patil, and A. Ingole (2017) conducted "A Survey on Accident Detection, Tracking and Recovery of Vehicles" [26]. This system serves as an embedded emergency response tool, utilizing Global Positioning System (GPS) and Global System for Mobile Communication (GSM) technologies. In case of an accident, it automatically pinpoints the exact vehicle location and sends alerts to pre-programmed emergency contacts, including police stations, ambulances, and hospitals. By leveraging GPS data, it provides the exact location on a map, facilitating swift response and rescue efforts.

Ramesh Mohanasundaram (2020) introduced a Web application for "Accident Emergency in Nearby Hospitals and Donor Locator" [27]. The proposed web app uses GPS to show users a list of all hospitals near their current location. The app also allows users to view a list of eligible blood and organ donors who are nearby. It acts as a donation community for donors and victims. This app could save people a lot of time and hassle by making it easier to find emergency medical services. It could also save lives by making it easier for blood donation centers and hospitals to coordinate donations and care.

Sharmila Gakwaid (2021) prepared a paper based on the "Systematic Literature Survey on Accident Alert & Detection" [28]. The purpose of this paper is to examine the various methods that have been used to reduce accidents, particularly by preventing and detecting them. This paper examines different proposed methods that use various techniques at different stages, as well as their advantages and disadvantages. This can help to determine and develop an effective and accurate accident alert and detection system. Based on a thorough analysis of existing solutions and literature, system specifications are proposed. A critical analysis and review of systems that have contributed to accident alerts and detection are also presented.

Shubham Deshpande (2018) introduced the "Real-time Mobility Model of AVCSS (Advanced Vehicle Control and Safety System) for Detecting Accidents with Tracking Technology" [29]. This paper proposes a system for automated accident response using connected vehicles. This network allows for real-time tracking of vehicles after accidents and transmits crucial information to facilitate rapid emergency service deployment. To illustrate the system's functionality, consider an emergency response network or SOS system. Upon detecting an accident, the vehicle automatically relays its geographical coordinates, enabling its real-time location visualization on Google Maps via an internet connection. The transmitted message additionally includes detailed vehicle information, aiding emergency personnel in their response.

Shuming Tang, Xiaoyan Gong, Feiyue Wang (2002) proposed a system called "Traffic incident detection algorithm based on non-parameter regression" [30]. Many countries worldwide struggle with traffic congestion, often leading to frustration and inefficiency. This work proposes a novel traffic incident detection algorithm based on non-parametric regression to tackle this challenge. The algorithm is then compared to existing methods, evaluating its performance in terms of detection rate, false alarms, and detection time. Simulation results demonstrate that the proposed approach boasts a higher detection rate, lower false alarms, and longer detection time, signifying its potential effectiveness.

Suman, Chiranjeev Kumar (2020) developed "An approach to detect the accident in VANETs using acoustic signal" [31]. Tragically, road accidents claim numerous lives each year, with an average of 16 fatalities per incident. This alarming statistic underscores the urgent need for solutions that prevent casualties. This paper presents a potential answer in the form of a robust, microprocessor-compatible algorithm designed to safeguard lives in vehicles. The paper delves into the algorithm's details and processes, highlighting its remarkable effectiveness. Rigorous testing demonstrates the algorithm's resilience, achieving an impressive 94.7% accuracy in collision detection and 88.7% recognition rate for human distress calls.

S. Veni, R. Anand, and B. Santosh (2020) prepared a paper on "Road Accident Detection and Severity Determination from CCTV Surveillance" [32]. This study proposes a novel method for accident detection based on analyzing the motion field dispersion of vehicles during collisions. By extracting the optical flow from video frames, the motion field of the road is generated. Subsequently, moving objects are isolated and tracked, enabling the calculation of the dispersion in their optical flow angle vectors. Significant deviations in these dispersions exceeding pre-defined thresholds flag potential accidents. Additionally, the severity of the accident can be estimated by the magnitude of the motion field dispersion range. This approach proves effective in detecting collisions between various types of moving objects.

Usman Khalil, Tariq Javid, Adnan Nasir (2017) proposed "Automatic road accident detection techniques: A brief survey" [33]. Equipping every vehicle with an automatic system for detecting and reporting accidents could significantly improve road safety. This paper surveys existing techniques for automatic accident detection, aiming to minimize casualties. Additionally, it proposes a new, cost-effective approach utilizing ultrasonic sensors for the same purpose.

Versavel, Jo (1999) introduced "Road safety through video detection" [34]. This article delves into two specific applications: the video and congestion monitoring system and the mobile road works monitoring system. Video detection, capable of both traffic data collection and automatic incident detection, shines in this role thanks to its high detection rate, rapid response time, efficient incident control, and minimal false alarms. These qualities make video invaluable for achieving incident management goals like swift intervention and preventing secondary incidents.

Yi et al. (2023) developed a new method for detecting highway crashes using distributed vibration sensors [35]. This article presents a new method for detecting highway vehicle collisions with guardrails using distributed vibration sensors. The method uses a 1DResNet and SVM model to identify unique features in the sensor data. It achieved an average accuracy of 96.2% in tests. The proposed method was tested in real-world highway accidents and achieved 100% accuracy in detecting crashes. This suggests that the method can improve highway safety and reduce accident risks.

3. Problem Definition

Road accidents are one of the major causes of disabilities and deaths as the population is increasing. The main reasons for road accident deaths are being unattended and excess blood loss, especially at night. Proper healthcare access to hospitals and rescue methods will save many lives.
There is an urgent need for an efficient model that can accurately detect accidents notify nearby hospitals for assistance and in the case of a fire explosion alert the nearby fire stations. The project aims to detect accidents using video inputs from CCTV cameras, process them using the trained GoogLeNet CNN model, and use the CCTV's IP address to determine the location of hospitals and fire stations. After finding this, send a message from the proposed system to the nearby hospitals and fire stations to ensure that the victims receive immediate help. In case of unresponsive from the hospitals in a limited period will automatically move to the next hospitals and fire stations.

4. Methodology

Our model was designed using an agile approach.

4.1 System Requirements

CCTV cameras are installed at regular intervals along a road to monitor video streams for accidents continuously. When an accident is detected, the system determines its location and alerts nearby hospitals to dispatch ambulances. When fire is detected during an accident, the system alerts the fire stations to take immediate action.
The CCTV camera should operate 24/7. The model should report accidents in less than 1-2 minutes, support at least 10 simultaneous users, and reliably report accidents.

4.2 Dataset Details

The dataset contains JPEG images of accidents captured from YouTube videos. It is split into three folders for training, testing, and validation purposes, each with Accident and Non-Accident subfolders. To determine fire explosions during accidents, each accident folder consists of fire and non-fire subfolders. This includes consecutive frames of accidents so the model can learn to distinguish between accidents and non-accidents and fire explosions.

4.3 Flow of Events

The diagram in Figure 1 shows a general overview of the system's workflow. The user logs in and the system begins monitoring traffic. If an accident is detected, the system identifies its location and the nearest hospital and then alerts the hospital and other required facilities. If a fire is found during the accident it alerts the fire stations. If no accident is detected, the system continues to monitor traffic.

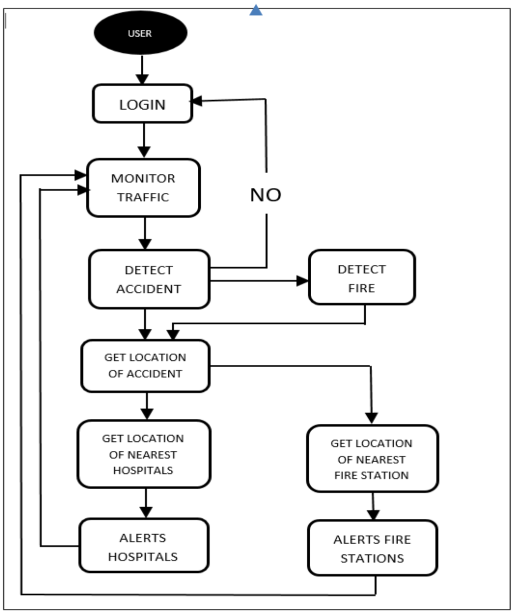


Figure 1: Flow of events of the proposed system

4.4 GoogLeNet CNN Model

GoogLeNet is a convolutional neural network (CNN) that was developed by researchers at Google and won the ImageNet

Large Scale Visual Recognition Challenge (ILSVRC) in 2014. It is a 22-layer deep network that uses a novel architecture called the inception module. This allows the network to learn a variety of different features at different scales. GoogLeNet can be used for computer vision tasks, including image classification, object detection, and image segmentation. It is also a popular choice for transfer learning, where a pre-trained GoogLeNet model is fine-tuned on a new dataset to solve a specific problem. It consists of a convolutional modules, padding modules, pooling layer, Inception modules and flattening modules. The implementation of the GoogLeNet CNN model in the Python notebook is shown in Figure 2.

```
In [10]: # Define the GoogleNet (Inception) architecture
def google_net():
    input_img = Input(shape=(250, 250, 3))

    # First Inception module
    tower_1 = Conv2D(64, (1,1), padding='same', activation='relu')(input_img)
    tower_1 = Conv2D(64, (3,3), padding='same', activation='relu')(tower_1)

    tower_2 = Conv2D(64, (1,1), padding='same', activation='relu')(input_img)
    tower_2 = Conv2D(64, (5,5), padding='same', activation='relu')(tower_2)

    tower_3 = MaxPooling2D((3,3), strides=(1,1), padding='same')(input_img)
    tower_3 = Conv2D(64, (1,1), padding='same', activation='relu')(tower_3)

    output = keras.layers.concatenate([tower_1, tower_2, tower_3], axis=3)

    # Second Inception module
    tower_4 = Conv2D(128, (1,1), padding='same', activation='relu')(output)
    tower_4 = Conv2D(128, (3,3), padding='same', activation='relu')(tower_4)

    tower_5 = Conv2D(128, (1,1), padding='same', activation='relu')(output)
    tower_5 = Conv2D(128, (5,5), padding='same', activation='relu')(tower_5)

    tower_6 = MaxPooling2D((3,3), strides=(1,1), padding='same')(output)
    tower_6 = Conv2D(128, (1,1), padding='same', activation='relu')(tower_6)

    output = keras.layers.concatenate([tower_4, tower_5, tower_6], axis=3)

    # Flatten the output and add fully connected layers
    output = Flatten()(output)
    output = Dense(256, activation='relu')(output)
    output = Dropout(0.5)(output)
    output = Dense(1, activation='sigmoid')(output)  # Binary classification (accident or not)

    model = Model(inputs=input_img, outputs=output)
    return model
```

Figure 2: GoogLeNet CNN Model

Convolutional layers are the primary blocks of CNNs. They perform feature extraction by applying convolution operations to the input data. In GoogLeNet, multiple convolutional layers are stacked to capture different levels of features. These layers typically use small-sized filters to convolve over the input data and apply non-linear activation functions.

Inception modules are designed to be computationally efficient while still achieving high accuracy. They do this by using a combination of convolution layers with different kernel sizes, as well as max-pooling layers. Each inception module consists of multiple branches, where each branch performs a different convolution operation. These branches are then concatenated to form the module's output.

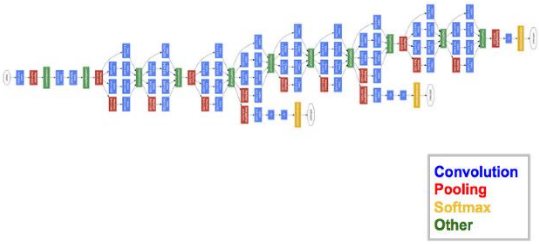


Figure 3: Detailed diagram of GoogLeNet CNN

GoogleNet, also known as Inception v1, is not defined by a single equation. It's a complex convolutional neural network with several layers as in the figure 3 and operations, each of which involves various equations and calculations. The key mathematical concepts behind different components of GoogleNet:

Convolution: This operation forms the core of CNNs, including GoogleNet. It involves sliding a filter (kernel) across an input image, multiplying corresponding elements, and summing the results. The equation for each element in the output feature map is:

$$output[i, j] = \sum(p[k, l] * f[k, l]) \quad (1)$$

where $p = input$, $f = filter$

where the summation covers all elements of the filter overlapping the input at position (i, j).

Activation functions: GoogleNet uses ReLU (Rectified Linear Unit) as its activation function. The equation for ReLU is:

$$ReLU(x) = max(0, x) \quad (2)$$

Pooling: This operation reduces the dimensionality of feature maps. GoogleNet uses average pooling, where the average value within a specified window is used as the output. The equation for each element in the output is:

$$output[i, j] = 1 / (window\_size) * \sum(p[k, l]) \quad (3)$$

where the summation covers all elements in the window centered at (i, j).

type	patch size/ stride	output size	depth	#1×1	#3×3 reduce	#3×3	#5×5 reduce	#5×5	pool proj	params	ops
convolution	7×7/2	112×112×64	1							2.7K	34M
max pool	3×3/2	56×56×64	0								
convolution	3×3/1	56×56×192	2		64	192				112K	360M
max pool	3×3/2	28×28×192	0								
inception (3a)		28×28×256	2	64	96	128	16	32	32	159K	128M
inception (3b)		28×28×480	2	128	128	192	32	96	64	380K	304M
max pool	3×3/2	14×14×480	0								
inception (4a)		14×14×512	2	192	96	208	16	48	64	364K	73M
inception (4b)		14×14×512	2	160	112	224	24	64	64	437K	88M
inception (4c)		14×14×512	2	128	128	256	24	64	64	463K	100M
inception (4d)		14×14×528	2	112	144	288	32	64	64	580K	119M
inception (4e)		14×14×832	2	256	160	320	32	128	128	840K	170M
max pool	3×3/2	7×7×832	0								
inception (5a)		7×7×832	2	256	160	320	32	128	128	1072K	54M
inception (5b)		7×7×1024	2	384	192	384	48	128	128	1388K	71M
avg pool	7×7/1	1×1×1024	0								
dropout (40%)		1×1×1024	0								
linear		1×1×1000	1							1000K	1M
softmax		1×1×1000	0								

Figure 4: Layer by Layer architectural details of GoogLeNet CNN model

Inception modules: These are the key innovations of GoogleNet. They combine multiple parallel convolution layers with different filter sizes and a 1x1 convolution for dimensionality reduction. While there's no single equation for the entire module, you can find equations for each layer within it.

5. Implementation

The implementation process is as follows:

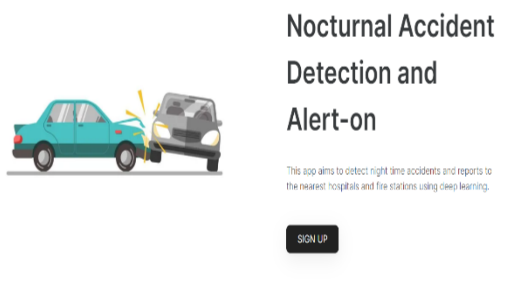


Figure 5: Admin login page

Figure 5 visually represents the application's landing page, while Figure 4 provides a screenshot of the login interface specifically designed for authorized personnel managing the CCTV surveillance system.

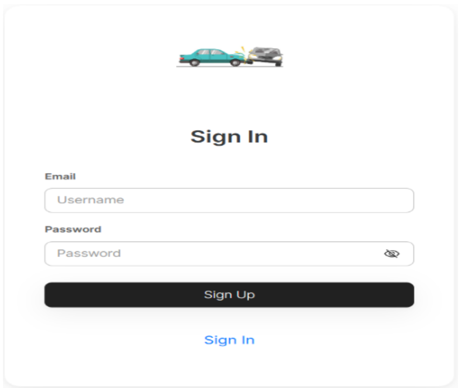
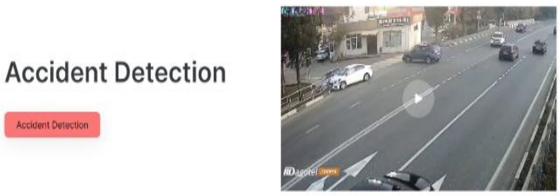


Figure 6: Admin sign in

Figure 7: Accident detection page when there is no accident

In the absence of any accident, Figure 7 highlights the "No accident" option as the selected choice.

Figure 8: Change in accident detection page when an accident occurs



In the event of an incident, Figure 8 showcases the readily available "Accident Details" option, allowing for immediate reporting and response.

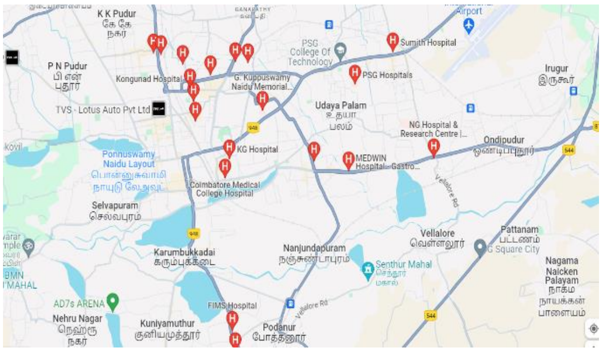


Figure 9: Details of hospitals

Imagine a vigilant network of CCTV cameras safeguarding your city. Each camera, identified by its unique IP address, feeds real-time footage into a central system. But this system isn't just passively monitoring; it's actively prepared to respond in emergencies. A comprehensive database stores details of all nearby hospitals, including city, state, and crucial location coordinates (latitude and longitude). This data is meticulously formatted in a common language the computer understands, like a .csv file (like a neatly organized spreadsheet).

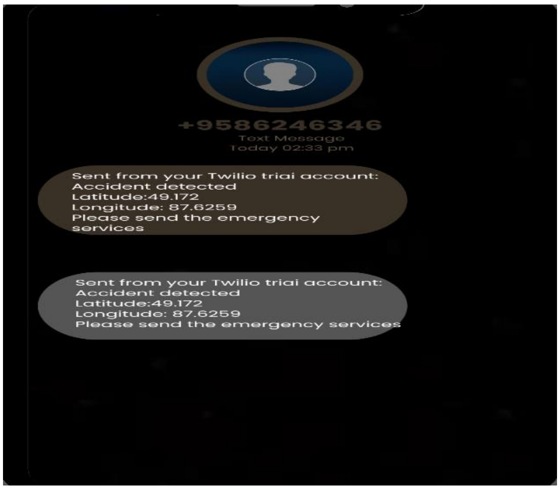


Figure 10: Alert Messages

6. Results and Analysis

A comprehensive evaluation of the system involved testing it on a range of video streams. Beyond the 97% accuracy in accident detection, we also monitored training and validation metrics like loss and accuracy, which are visualized in the following figures.

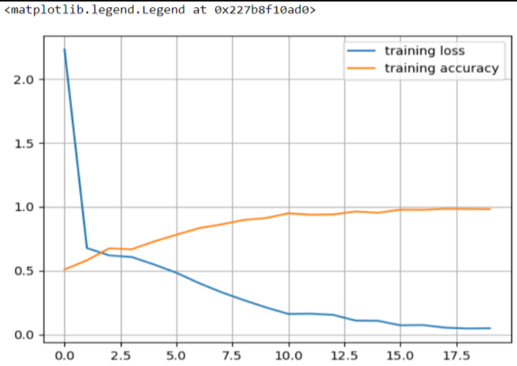


Figure 11: Training loss vs accuracy

The formula for calculating training accuracy is:

$$Training\ Accuracy = ((TP + TN)/\ T) * 100 \quad (4)$$

TP - True Positives
TN - True Negatives
T - Total Number of Predictions

Here, correct predictions represent instances where the model's output matches the desired outcome for each input in the training data. The total number of predictions simply refers to the entire training set size. Typically, when visualizing training loss and training accuracy over time, the ideal scenario shows a decreasing loss alongside an increasing accuracy. However, it's essential to recognize that these metrics can sometimes behave independently. A decreasing loss doesn't always guarantee improved accuracy, and vice versa. This often occurs when a model overfits the training data, meaning it memorizes specific patterns instead of learning generalizable concepts. For optimal performance, a good GoogLeNet convolutional neural network should strive for both low training loss and high training accuracy. This indicates the model's ability to make accurate predictions on the training data while maintaining the flexibility to adapt to unseen data.

As depicted in Figure 11, the training loss line exhibits a consistent decline, while the training accuracy line demonstrates a steady and gradual upward trend, supporting this desired behavior.

The graph of validation loss vs accuracy is a common way to visualize the performance of a GoogLeNet CNN model during the training phase. The validation loss is a measure of how well the model is predicting the correct output on a separate validation set of data, while the validation accuracy is a measure of how often the model's predictions match the actual output on the validation set.

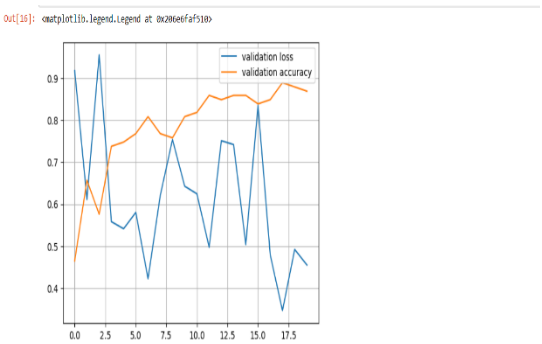


Figure 12: Validation loss vs accuracy

Validation Accuracy = (Number of Correct Predictions / Total Number of Predictions) * 100 (5)

The formula for validation loss:

$$L\,(y, \hat{y}) = -\sum y\_i * log\,(\hat{y}\_i) \quad (6)$$

where y is the actual output on the validation dataset, ŷ is the predicted output on the validation dataset, and i is the index of the output vector.

During the training phase, the model is trained on a set of training data, and its weights are adjusted based on the difference between the predicted output and the actual output. However, it is important to evaluate the performance of the model on a separate validation set to ensure that it can generalize well to new data. The validation set is usually a subset of the overall dataset that is not used for training the model.

Figure 12 represents a graph of validation loss vs accuracy, wherein it shows the validation loss decreasing over time, while the validation accuracy increases. This indicates that the model is becoming better at making accurate predictions on the validation set as it trains. However, it is important to note that the validation loss and accuracy may not always follow the same trend as the training loss and accuracy. If the validation loss begins to increase while the validation accuracy decreases, it could indicate that the model is overfitting to the training data and is unable to generalize well to new data. This means that the model is becoming too complex and is fitting to the noise in the training data, rather than the underlying patterns. To prevent overfitting, techniques such as regularization and dropout can be used to reduce the complexity of the model and prevent it from fitting to the noise. In general, a good GoogLeNet CNN model should have both low validation loss and high validation accuracy, indicating that it can make accurate predictions on new data that it has not seen before. Monitoring the validation loss and accuracy can help in determining if the model is learning effectively and if there are any issues with overfitting.

The GoogLeNet CNN model utilizes the cross-entropy loss function to evaluate the discrepancy between its predicted labels and the ground truth labels within the training data. This function, mathematically expressed as:

*cross_entropy_loss = -sum(y * log (y_pred))* (7)

where y represents the one-hot encoded actual label and y_pred signifies the predicted label probability distribution, aiming to minimize the difference. The negative sign flips the problem into a maximization one.

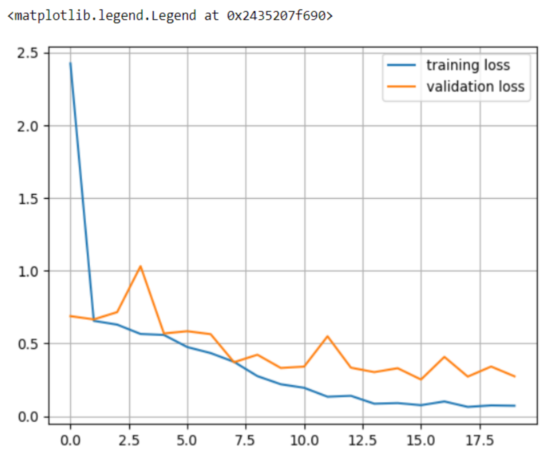


Figure 13: Cross-entropy loss

Figure 13 illustrates the training and validation loss curves during GoogLeNet's training for accident detection. As the number of training iterations (epochs) increases, the training loss consistently decreases, suggesting the model actively learns from the provided data. The validation loss exhibits an initial decline followed by a plateau after a few epochs. Notably, the gap between the training and validation loss remains minimal. This overall trend indicates effective learning and promising performance in accident detection by the GoogLeNet CNN model.

Figure 14 shows the images in the output of the proposed system and the actuals concerning accidents. Below each sample image, the actual and predicted are noted.

Figure 14: The output of the proposed system

Figure 15 shows the image of the output of the accident detection from the predicted image and the actual image. Here the actual image is of an accident and the model prediction is the same as the actual image that the image is accidental.

Figure15: Output of the accident image

Similarly, in figure 16 the actual image is of non-accident and the model predict is same as the actual image that the image is a non-accidental.

Figure 16: Output of the non-accidental image

Comparison of the proposed system will help to analyse and find the efficiency of the system. To evaluate the proposed system with an existing system, taking the existing system named "Real Time Accident Detection from CCTV and Suggestion of Nearest Medical Amenities" [10]. Table 1 shows the comparison of the proposed system with the existing system.

Comparison	Existing system	Proposed system
Algorithm used	Convolutional Neural Network(CNN)	GoogLeNet Convolutional Neural Network
Dataset	Limited data	A large amount of data
Computational speed	5-6 minutes	2-3 minutes
Accuracy	96%	97%
Efficiency	Efficiency in detecting image	Efficient in the detection, and classification of images including object detection.

Table 1: Comparing the existing system with the proposed system

7. Conclusion

The implementation of an automated traffic accident detection system employing the GoogLeNet model holds immense promise for enhancing road safety and minimizing the likelihood of fatal accidents. Leveraging the ubiquity of video surveillance and advanced traffic management systems, this approach demonstrates significant potential for real-time detection of various accident types. The model's remarkable performance in terms of accuracy and recall during training and validation phases underscores its efficiency in traffic accident detection. Further research activities could investigate strategies for augmenting the model's performance by integrating additional data sources, such as audio and sensor data, to refine the system's accuracy. Additionally, training the model on a broader range of datasets could enhance its generalization capabilities and optimize its performance across diverse settings. The proposed system's effectiveness could be further validated through extensive field testing to assess its accuracy and reliability in real-world scenarios. In conclusion, the proposed approach for traffic accident detection utilizing the GoogLeNet model exhibits exceptional potential for road safety and mitigating the risk of fatal accidents. With continued research and development, this system could become an invaluable tool for accident prevention and lifesaving interventions.

8. Future Scope

While the proposed system using GoogLeNet and CCTV footage addresses immediate needs, the future lies in harnessing interconnected technologies for even greater impact. While the current system focuses on night-time accidents, future iterations could incorporate additional sensors and algorithms to detect accidents in all lighting conditions. This will help a 24*7 detection. Secondly, leveraging AI and machine learning to analyze accident severity and predict medical needs can expedite emergency response. This might involve extracting information from CCTV footage about vehicle damage, the number of occupants, and potential fire hazards. Thirdly, developing collaboration between emergency services through a central platform can optimize resource allocation and response times. This platform could facilitate communication between hospitals, fire stations, and even automated vehicles for faster assistance. Finally, expand the system beyond immediate response to offer post-accident support. This could include facilitating communication with emergency contacts, providing real-time traffic updates to avoid congested areas, and assisting with accident reporting and insurance claims. By combining these approaches, we can move beyond merely detecting accidents to preventing them, providing swift and tailored medical care, and ultimately saving lives.

9. References

[1] Kodali, Ravi Kishore, and Shubhi Sahu. "MQTT-based vehicle accident detection and alert system." 2017 *3rd International Conference on Applied and Theoretical Computing and Communication Technology (iCATccT)* (2017): 186-189.

[2] A. S. Bari, M. A. Falalu, M. A. Umar, Y. Y. Sulaiman, A. M. Gamble, & M. A. Baballe. (2022). Accident Detection and Alerting Systems: A Review. Global Journal of Research in Engineering & Computer Sciences, 2(4), 24–29. https://doi.org/10.5281/zenodo.7063008

[3] Vijayaraja, L., Dhanasekar, R., Magesh Krishna, R., Mahidhar, M., Prakash, D., & Shashikumar, P. (2021). A low-cost and user-friendly vehicle crash alert system using Arduino. *IOP Conference Series: Materials Science and Engineering*, *1055*.

[4] Maliye, M., Batheja, M.,. (2018). ACCIDENT DETECTION AND ALERTING SYSTEM. *Journal of emerging technologies and innovative research*.

[5] Sadaphal, Ishan. "Accident Detection and Alert System using Android Application." *International Journal for Research in Applied Science and Engineering Technology* (2019): n. pag.

[6] Kattukkaran, N., George, A., & Haridas, T. (2017). Intelligent accident detection and alert system for emergency medical assistance. *2017 International Conference on Computer Communication and Informatics (ICCCI)*, 1-6.

[7] Vyas, Harsh & Sharma, Samarth & Senghani, Harshil & Rathod, Dr and Vasant, Dr. (2023). Accident Prone System using YOLO. *International Journal of Scientific Research in Science, Engineering and Technology*. 09-16. 10.32628/IJSRSET23102120.

[8] B. K. M, A. Basit, K. MB, G. R, and K. SM, "Road Accident Detection Using Machine Learning," *2021 International Conference on System, Computation, Automation and Networking (ICSCAN)*, Puducherry, India, 2021, pp. 1-5, doi: 10.1109/ICSCAN53069.2021.9526546.

[9] Maaloul, Boutheina et al. "Adaptive video-based algorithm for accident detection on highways." *2017 12th IEEE International Symposium on Industrial Embedded Systems (SIES)* (2017): 1-6.

[10] Reddy, A. Vamshidhar et al. "IOT-Based Accidental Detection System (ADS) Using Raspberry Pi." *2023 1st International Conference on Innovations in High Speed Communication and Signal Processing (IHCSP)* (2023): 392-397.

[11] Balfaqih, Mohammed & Alharbi, Soltan & Alzain, Moutaz & Alqurashi, Faisal & Almilad, Saif. (2021). An Accident Detection and Classification System Using Internet of Things and Machine Learning towards Smart City. Sustainability. 14. 210. 10.3390/su14010210.

[12] Senthil Murugan, K R et al. "IOT Based Road Accident Detection and Prevention System." *2023 International Conference on Quantum Technologies, Communications, Computing, Hardware and Embedded Systems Security (iQ-CCHESS)* (2023): 1-5.

[13] Kinage, Vivek and Piyush S. Patil. "IoT Based Intelligent System For Vehicle Accident Prevention And Detection At Real Time." *2019 Third International Conference on I-SMAC (IoT in Social, Mobile, Analytics and Cloud) (I-SMAC)* (2019): 409-413.

[14] S. Ghosh, S. J. Sunny and R. Roney, "Accident Detection Using Convolutional Neural Networks," *2019 International Conference on Data Science and Communication (IconDSC)*, Bangalore, India, 2019, pp. 1-6, doi: 10.1109/IconDSC.2019.8816881.

[15] Abishek, A. T. S., Jose, A., Davis, A., Thomas, A., & Davies, J. (2021). Accident Detection and Warning System. *International Journal of Innovative Science and Research Technology*, 6(6), 328-332. ISSN: 2456-2165.

[16] Akanksha A. Pai, Harini " Real Time Accident Detection from CCTV and Suggestion to Nearest Medical Services", *International Journal of Computer Science and Information Technology*, Vol.15, No.6, pp.15-28, 2023. DOI:10.5815/ijitcs.2023.06.02.

[17] C. K. Gomathy, K. Rohan, B. M. Kiran Reddy, and V. Geetha, "Detection of Accident and Alerting Mechanism," Journal of Engineering, Computing & Architecture, ISSN: 1934-7197

[18] Gaikwad, S.S., Khanna, M., Kothari, S., & Kudale, A. (2021). Systematic Literature Survey on Accident Alert & Detection System.

[19] Mohanraj, E., Dakshnamoorthy, M., & Karthikeyan, S. (2022). Accident prevention using IoT. *International journal of health sciences.*

[20] Elumalai, G., Mathanki, O.S., & Swetha, S. (2015). Vision-Based Intelligent Traffic Analysis System for Accident Detection and Reporting System.

[21] Gupta, S., & Jain, S. (2019). Accident Detection using Convolutional Networks Neural. *2019 International Conference on Data Science and Communication (IconDSC),* 1-6. https://doi.org/10.1109/IconDSC47508.2019.9036947

[22] Ijjina, E. P., Chand, D., Gupta, S., & Goutham, K. (2019). Computer vision-based accident detection in traffic surveillance. *2019 10th International Conference on Computing, Communication and Networking Technologies (ICCCNT)*, 1-6.https://doi.org/10.1109/ICCCNT45670.2019.8946176

[23] Ahmed, B., Imran, M., Zaghdoud, R., Ahmed, M. S., Sendi, R., Alsharif, S., Abdulkarim, J., Albin Saad, B. A., Alsabt, R., Rahman, A., & Krishnasamy, G. (2023). Detection and classification of traffic incidents in real-time based on a computer vision approach. Big data and cognitive computing, 7(1), 22. https://doi.org/10.3390/bdcc7010022 .

[24] ojsadmin, ojsadmin. "Automatic Vehicle Accident Detection Based on GSM System." (2017).

[25] Sarma, Praharsha and Utkarsh Kumar. "Accident Detection And Prevention Using Iot & Python Opencv." (2020).

[26] More, Pranali et al. "A Survey on Accident Detection, Tracking and Recovery of Vehicles." (2017).

[27] Mohanasundaram, Ramesh & Arun, R & Jothi, K & Malhotra, Ankit & Gopinath, Masila Pandiasankar & Ramasamy, Lokeshkumar. (2020). Web app for Accident Emergency to Nearby Hospitals and Donor Locator. *Bioscience Biotechnology Research Communications.*

[28] Bansal, Dr. Bharat Naresh, and Vivek Garg. "A Review paper on "Vehicle Accident Detection, Tracking, and Notification Systems"- A comparative study." (2021).

[29] Deshpande, Shubham. "Realistic Mobility Model of AVCSS (Advanced Vehicle Control and Safety System) for Accident Detection with Tracking Technology." *International Journal for Research in Applied Science and Engineering Technology* 6 (2018): 2099-2101.

[30] Tang, Shuming et al. "Traffic incident detection algorithm based on non-parameter regression." *Proceedings. The IEEE 5th International Conference on Intelligent Transportation Systems* (2002): 714-719.

[31] Suman, Amrit and Chiranjeev Kumar. "An approach to detect the accident in VANETs using acoustic signal." *Applied Acoustics* (2020): n. pag.

[32] Veni, S., Anand, R., & Santosh, B. (2020). Road Accident Detection and Severity Determination from CCTV Surveillance. *Lecture Notes in Networks and Systems*.

[33] Khalil, Usman et al. "Automatic road accident detection techniques: A brief survey." *2017 International Symposium on Wireless Systems and Networks (ISWSN)* (2017): 1-6.

[34] Versavel, Jo. "Road safety through video detection." *Proceedings 199 IEEE/IEEJ/JSAI International Conference on Intelligent Transportation Systems (Cat. No.99TH8383)* (1999): 753-757.

[35] Yi, Jichao, Shang, Y., Wang, C., Du, Y., Yang, J. (2023). An effective crash recognition method based on 1DResNet-SVM with distributed vibration sensors. Optics Communications, 529, https://doi.org/10.1016/j.optcom.2021.129263