Intelligent Video-Based Accident Detection and Automated Emergency Alert System Leveraging ResNet-50 Architecture

Natramizh Saravanan[1]; Vigneshwaran S.[2]; Syed Roshan Abbas Kazmi[3]; Dr. V. Saminadan[4]

[1,2,3,4]Department of Electronics and Communication Engineering, Puducherry Technological University, Puducherry, India.

Publication Date: 2026/05/05

Abstract: The rapid growth in vehicle usage due to increasing population has led to a significant rise in road accidents, making it a serious global concern. According to the World Health Organization (WHO), road accidents are among the leading causes of death worldwide, claiming millions of lives each year. Factors such as reckless driving, violation of traffic rules, increased congestion in urban areas, and distractions like mobile phone usage contribute heavily to these incidents. Additionally, delayed emergency response and lack of immediate medical assistance are major reasons for increased fatalities, highlighting the need for a faster and more efficient accident response system.

To address this issue, an automated accident detection and rescue system based on deep learning is proposed. The system operates in two phases: the first phase involves detecting accidents using image preprocessing and a Convolutional Neural Network (CNN) with the ResNet50 algorithm, trained on a custom dataset created from online video sources due to limited dataset availability. In the second phase, once an accident is detected, an alert message is automatically sent to emergency services to initiate rescue operations. This approach eliminates the need for human intervention and ensures quicker response times, thereby improving the chances of saving lives. The use of ResNet50 enhances detection accuracy, making the system more reliable compared to traditional methods.

Keywords: *Accident Detection, Deep Learning, Convolutional Neural Network (CNN), ResNet-50, Computer Vision, Traffic Surveillance, Automated Alert System, Transfer Learning, Internet of Things (IoT).*

How to Cite: Natramizh Saravanan; Vigneshwaran S.; Syed Roshan Abbas Kazmi; Dr. V. Saminadan (2026) Intelligent Video-Based Accident Detection and Automated Emergency Alert System Leveraging ResNet-50 Architecture. *International Journal of Innovative Science and Research Technology*, 11(4), 3251-3257. https://doi.org/10.38124/ijisrt/26apr1846

I. INTRODUCTION

The rapid urbanization and continuous growth in the number of vehicles have made road safety a critical concern worldwide. Urban roads are becoming increasingly congested, and with the rise in traffic density, the likelihood of accidents has also increased significantly. Various factors such as rash driving, lack of adherence to traffic rules, distracted driving, and unfavourable environmental conditions contribute to these incidents. Despite advancements in transportation infrastructure, managing and responding to accidents efficiently remains a major challenge.

One of the most crucial issues in accident scenarios is the delay in detection and emergency response, which often leads to severe consequences and loss of life. To overcome this limitation, there is a growing need for intelligent systems that can assist in real-time monitoring and rapid incident reporting. In this context, the proposed system introduces a deep learning-based approach for accident detection using CCTV footage, combined with an automated alert mechanism. This approach aims to enhance existing traffic monitoring systems by enabling quicker response times and supporting efficient rescue operations.

II. MOTIVATION

Currently, roads are monitored passively through CCTV cameras, meaning that the cameras themselves do not actively generate intelligent alerts. Furthermore, the deployment of traffic police personnel is often highly restricted at road crossings, highways, and remote locations. In many instances, accidents and injuries occur due to pedestrian negligence or individuals being stranded in isolated areas. Adding to the severity of this issue, the majority of bystanders surrounding an accident scene are often busy capturing photographs and videos, ignorant of the

ISSN No: -2456-2165 https://doi.org/10.38124/ijisrt/26apr1846

fact that their slight negligence and delay in helping could cost a life. Moreover, the camera footage is typically only available to government agencies after the incident has already occurred and the damage is done. To overcome these critical challenges, the proposed system is designed to utilize existing CCTV cameras to capture traffic 24/7 and actively process the video feeds using deep learning algorithms. The primary motivation of this project is to automatically discover accidents and immediately send signals to ambulances or firefighting services, ensuring that the suitable resources needed for saving lives are dispatched in time.

III. RELATED WORKS

Numerous studies have been conducted to automate accident detection using various hardware and software methodologies. The existing literature can be broadly categorized into sensor-based IoT systems and computer vision-based deep learning approaches.

➢ *Sensor and IoT-Based Systems:*

Many conventional models heavily rely on external hardware components. For instance, Lakshmy et al. proposed a system utilizing sensors to detect accidents and alcohol consumption alongside a supervised Convolutional Neural Network (CNN) [4]. Similarly, Rathod and Gadhiya developed a system using edge devices where a deep learning model (utilizing ResNet and InceptionV2) is activated only if connected physical sensors detect an anomaly [2]. The primary limitation of these approaches is their physical vulnerability; sensors can easily become faulty or get completely destroyed during the impact of an accident, leading to a higher possibility of false alarms or total system failure.

➢ *Computer Vision and Object Detection Approaches:*

To eliminate hardware dependency, several researchers have shifted towards purely vision-based detection via traffic cameras. Ghahremannezhad et al. proposed a real-time traffic surveillance framework using the state-of-the-art YOLOv4 object detection algorithm [1]. While effective, implementing this model consumes excessive computational power and requires a significant amount of time. Ahmed et al. also adopted a dual-algorithm approach, using YOLOv4 for accident detection alongside ResNet-152 to identify fire ignitions [6]. However, training such models on two separate algorithms necessitates the use of highly expensive supercomputers.

➢ *CNN and Deep Learning Variants:*

Other related works have directly applied CNNs and their variants to classify traffic anomalies. Kumeda and Fengli utilized deep CNNs to classify accidents into different categories based on traffic, time, and location, though this approach required a massive dataset and struggled to produce highly accurate results [3]. Renu et al. explored combining CNNs with Long Short-Term Memory (LSTM) networks, but the system's real-time performance was low, and it struggled to detect accidents during improper visibility conditions [7]. Furthermore, Ghosh et al. developed a CNN-based model to predict accident severity and alert relevant stakeholders [8]. The major drawbacks of their model included a reliance on pre-trained datasets without object segmentation, and an inability to train complete data due to the vanishing or exploding gradient problem.

IV. TECHNIQUES USED

➢ *Video Pre-Processing*

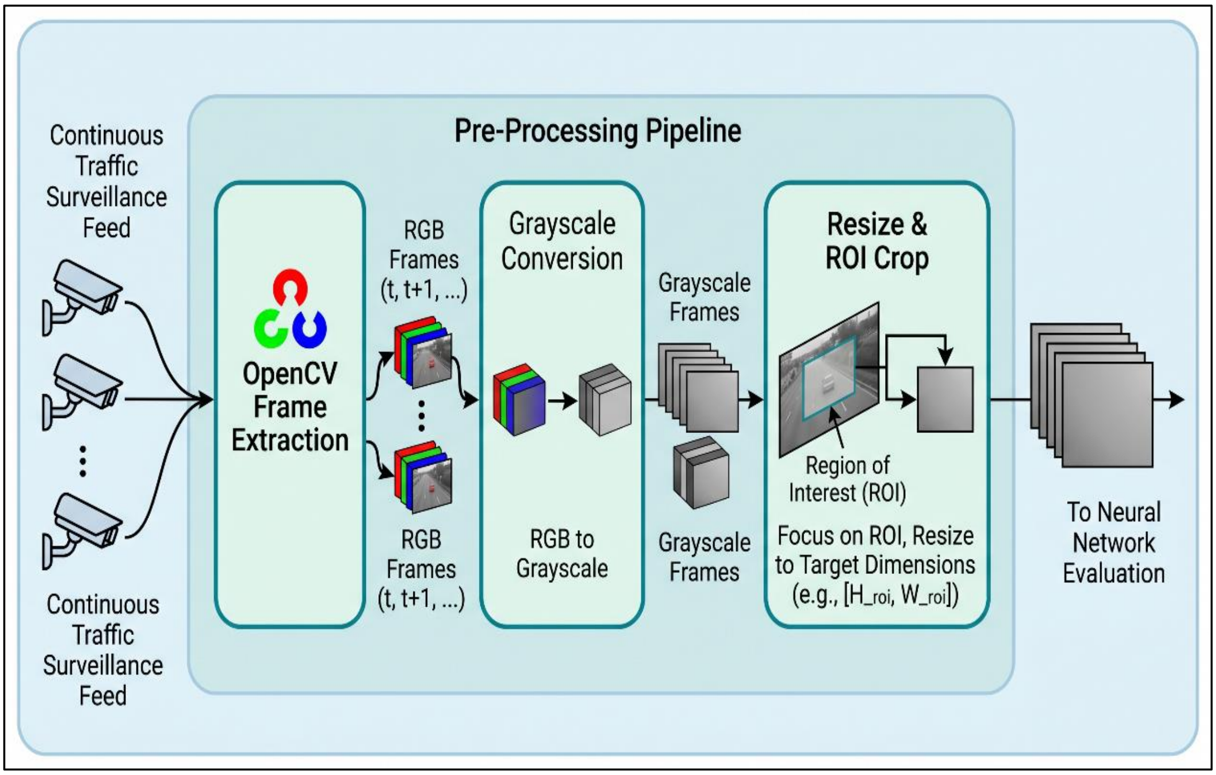

Fig 1 Preprocessing Steps for Video Ingestion

ISSN No: -2456-2165 https://doi.org/10.38124/ijisrt/26apr1846

The initial phase of the system involves handling the continuous video feed obtained from traffic surveillance cameras. Using the OpenCV library, the input video is extracted and broken down into individual image frames. These frames then undergo pre-processing, where they are converted from RGB to Grayscale formats and resized to focus on the specific region of interest before being evaluated by the neural network.

➤ *Convolutional Neural Network (CNN)*

To enable intelligent computer vision, the system employs a Convolutional Neural Network (CNN), a deep learning algorithm highly suited for image recognition and processing. The CNN automatically extracts spatial features such as shapes, edges, and textures through a series of convolutional layers. Simultaneously, pooling layers (such as max pooling) are used to down-sample the feature maps, reducing spatial dimensions and lowering the required computational load before passing the data to fully connected layers for classification.

➤ *ResNet-50 Architecture*

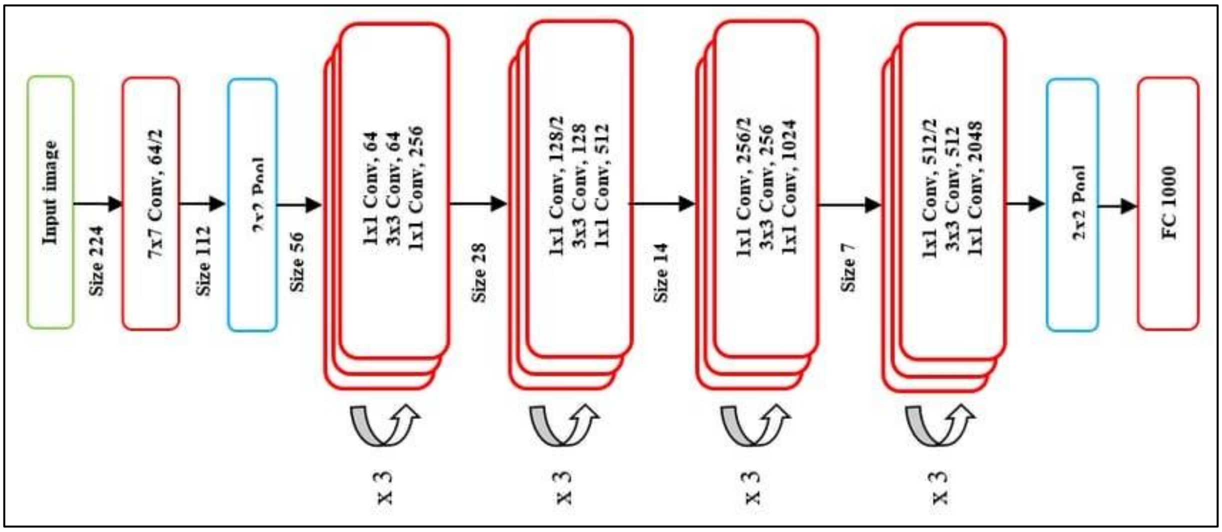

Fig 2 ResNet-50 Architectural Diagram

The core accident detection engine is built upon the ResNet-50 architecture, a highly robust CNN variant comprising 48 Convolution layers, 1 MaxPool layer, and 1 Average Pool layer. Traditional deep neural networks often suffer from the "Vanishing Gradient problem," where training accuracy degrades and weights fail to update effectively as the network gets deeper. ResNet-50 completely resolves this through residual blocks with "skip-connections" (mathematically denoted as $y = x + F(x)$). These skip-connections act as gradient super-highways, allowing data to flow without magnitude alteration. The network extensively leverages Identity and Convolutional blocks, paired with Conv2D layers, Batch Normalization to standardize data and speed up training, and Rectified Linear Unit (ReLU) activation functions to introduce non-linearity and avoid saturation.

➤ *Transfer Learning*

Training deep convolutional networks from scratch demands massive amounts of data and exorbitant computational resources. To overcome this limitation, the system implements Transfer Learning. The model reuses pre-trained weights from standard benchmark computer vision datasets (such as ImageNet). The top layers of the pre-trained model are omitted and replaced with custom dense layers specifically designed to classify the processed frames into binary categories: ‘Accident’ or ‘No Accident’.

➤ *Automated Alert Mechanism*

Once the deep learning model detects an accident with a probability score of 90% or higher, it triggers the alert module. This communication phase utilizes the Vonage API, a REST-based interface that programmatically transmits SMS alerts with low latency to nearby emergency responders.

Additionally, the system incorporates Internet of Things (IoT) techniques using an Arduino UNO microcontroller connected to an ESP8266 (NodeMCU) Wi-Fi module. Through serial communication at a 9600 baud rate, the Python detection script triggers the Arduino. The Arduino then utilizes the ESP8266 to upload critical accident logs—including the Log ID, date, time, and location—to a dedicated web server page for centralized tracking and future reference.

ISSN No: -2456-2165 https://doi.org/10.38124/ijisrt/26apr1846

➢ *Proposed Work*

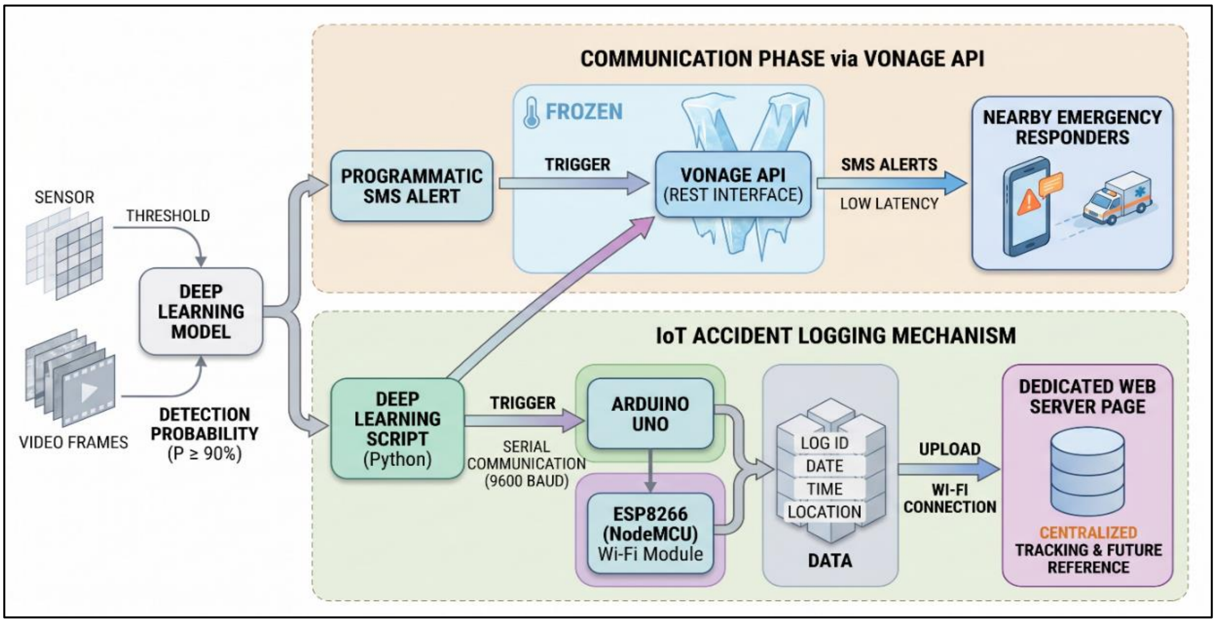

Fig 3 Process of Automated Alert Mechanism

The proposed work is an intelligent accident detection and rescue system that utilizes deep learning algorithms to monitor traffic via CCTV cameras and automatically alerts nearby emergency response services using an API. To ensure the system operates continuously without human interference, the framework is divided into two primary components: the Detection (Deep Learning) Module and the Alert Module.

- *Detection (DL) Module:*

The detection module serves as the core computer vision engine, utilizing the ResNet-50 architecture to classify continuous traffic frames into two distinct classes: ‘Accident’ and ‘No Accident’.

✓ *Data Preparation and Pre-Processing:*

Due to the general unavailability of a standardized dataset, the model is trained using a Kaggle dataset consisting of 1,500 CCTV footage frames extracted from various YouTube videos. The data is processed using the Keras ImageDataGenerator and image_dataset_from_directory() functions to handle batching and augmentation. During real-time surveillance, the video feed is processed using OpenCV to extract individual frames, convert them from RGB to Grayscale, and resize them to the specific region of interest.

✓ *Model Implementation:*

Training a deep Convolutional Neural Network from scratch requires massive computational resources; thus, the proposed model utilizes Transfer Learning. The ResNet-50 model reuses pre-trained weights from the standard ImageNet benchmark dataset, replacing the top layers with custom dense layers tailored to this specific binary classification task.

✓ *Detection Process:*

The live video feed runs continuously through the trained model, which uses NumPy to calculate the probability of an accident occurring in the frame. If the calculated probability score reaches 90% or above, the system immediately triggers the Alert Module.

- *Alert Module:*

Once the accident probability threshold of 90% is met, the system initiates a warning beep and activates both the software API and the hardware IoT components to ensure rapid response.

✓ *Vonage API for SMS Alerts:*

To facilitate immediate communication, the system utilizes the Vonage API, a REST-based interface. By leveraging a private key and API secret, the system automatically sends a low-latency SMS alert containing the incident details to the appropriate emergency responders globally.

✓ *Hardware and IoT Cloud Logging:*

Alongside the API, the system integrates an Arduino UNO microcontroller coupled with an ESP8266 (NodeMCU) Wi-Fi module. When an accident is detected, the Python detection script transmits the character 'A' to the Arduino via serial communication at a 9600 baud rate, prompting the Arduino to display an "Accident detected" message on an attached LCD. Simultaneously, the ESP8266 module connects to the internet and uploads a comprehensive log of the event—including a unique Log ID, the exact date, time, and location—to a dedicated centralized web server (www.iotclouddata.com) for future reference and administrative tracking.

V. DATASET

➢ *Dataset Origin and Description*

International Journal of Innovative Science and Research Technology
ISSN No: -2456-2165
https://doi.org/10.38124/ijisrt/26apr1846

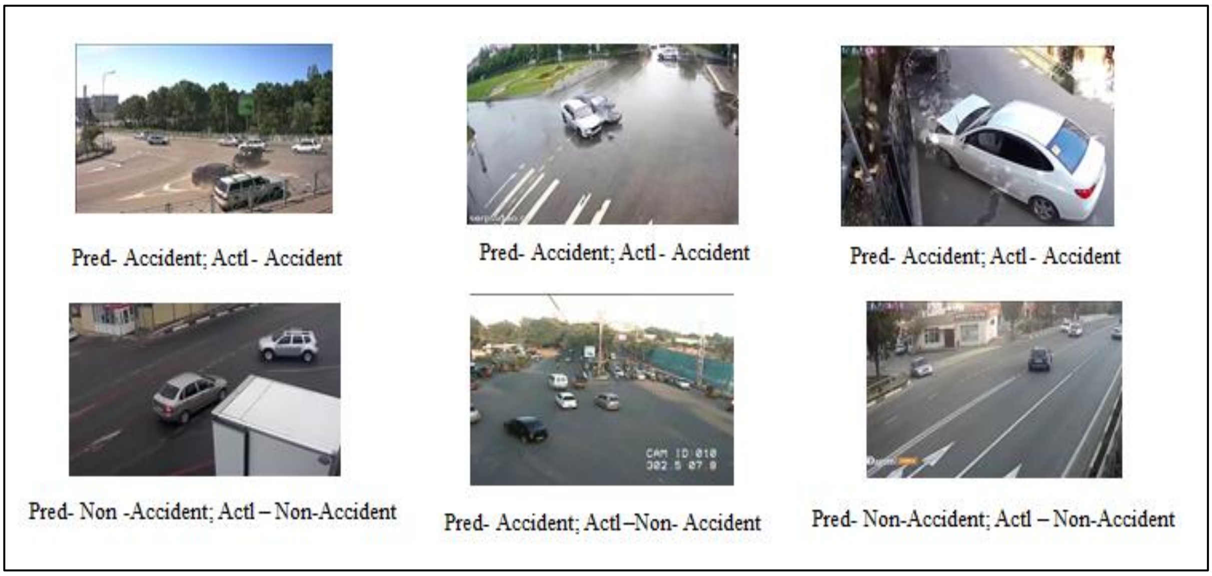

Fig 4 Kaggle Dataset for Accident Classification

A significant challenge in developing computer vision-based accident detection models is the general unavailability of standardized, real-time accident datasets. To address this, the proposed system utilizes a personalized dataset sourced from Kaggle, specifically focused on accident detection from CCTV footage. This dataset is composed of various image frames extracted from YouTube videos that capture real-world traffic scenarios, including both accident and non-accident events.

➢ *Data Composition and Organization*

Deep learning models heavily rely on the quantity and quality of data for accurate training. The selected Kaggle dataset consists of a total of 1,500 extracted frames. To facilitate a robust training and evaluation process, this data is systematically partitioned into three primary directories: train, test, and val (validation). Within each of these directories, the frames are further categorized into two distinct sub-folders corresponding to the classification outputs: 'Accident' and 'Non-Accident'. Notably, the dataset includes consecutive frames of collision events, which is crucial for helping the model learn the progressive visual differences between an unfolding accident and regular traffic movement.

➢ *Data Preparation and Augmentation*

Loading large image datasets directly into system memory can cause severe computational bottlenecks or memory failures. To efficiently handle the dataset during the training phase, the system employs Keras's built-in image_dataset_from_directory() function, which automatically generates a TensorFlow dataset containing the images and maps them to their respective folder classes. Additionally, the system utilizes the Keras ImageDataGenerator class. This class enables progressive loading of the images in batches and allows for real-time image data preparation and augmentation, ensuring that the model can handle the dataset smoothly without exceeding hardware memory limits.

VI. RESULTS

➢ *Training and Validation Performance*

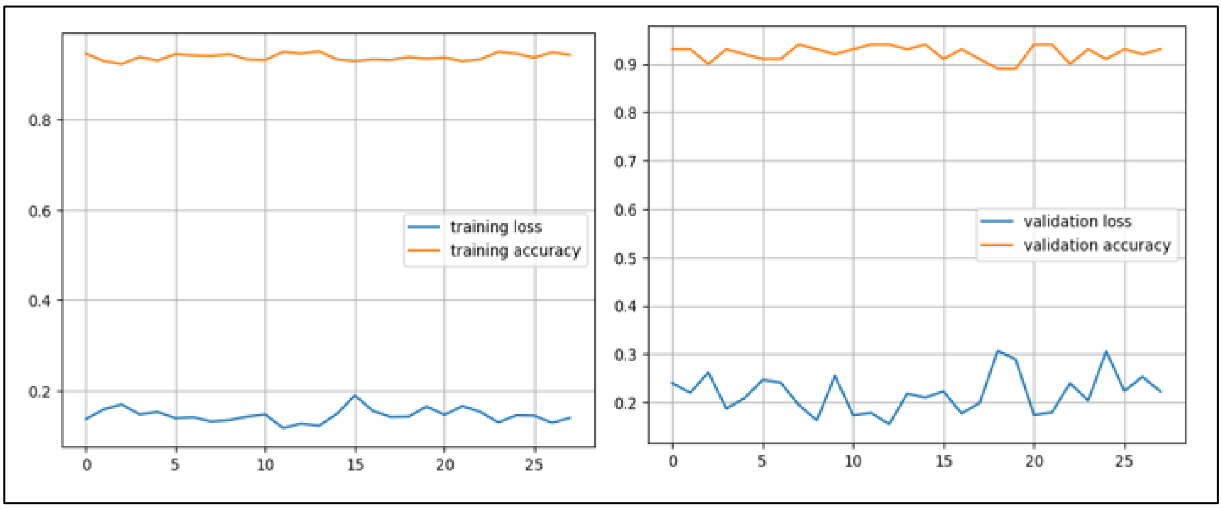

Fig 5 Epoch vs Training Characteristics and Epoch vs Validation Characteristics

ISSN No: -2456-2165 https://doi.org/10.38124/ijisrt/26apr1846

The proposed ResNet-50 deep learning model was trained and evaluated using the customized Kaggle dataset, which consisted of 1,500 CCTV footage frames properly partitioned into 'Accident' and 'Non-Accident' classes. Following the model compilation and training phases, the network successfully learned the complex visual features of vehicular collisions. The model demonstrated robust performance, achieving a highly reliable training accuracy of 95% and a validation accuracy of 92%.

➢ *Real-Time Detection Outcomes*

Fig 6 Output of the Detection Module

During the real-time continuous surveillance phase, the system effectively processed input video feeds by extracting individual frames, converting them from RGB to grayscale, and resizing them to the designated region of interest. The model, loaded via its saved JSON and weights files, continuously evaluated these frames to determine an accident probability score. The output of the detection module proved capable of accurately identifying an unfolding accident and immediately triggering the rescue module once the calculated probability score crossed the predefined 90% threshold.

➢ *Alert and Hardware Module Implementation*

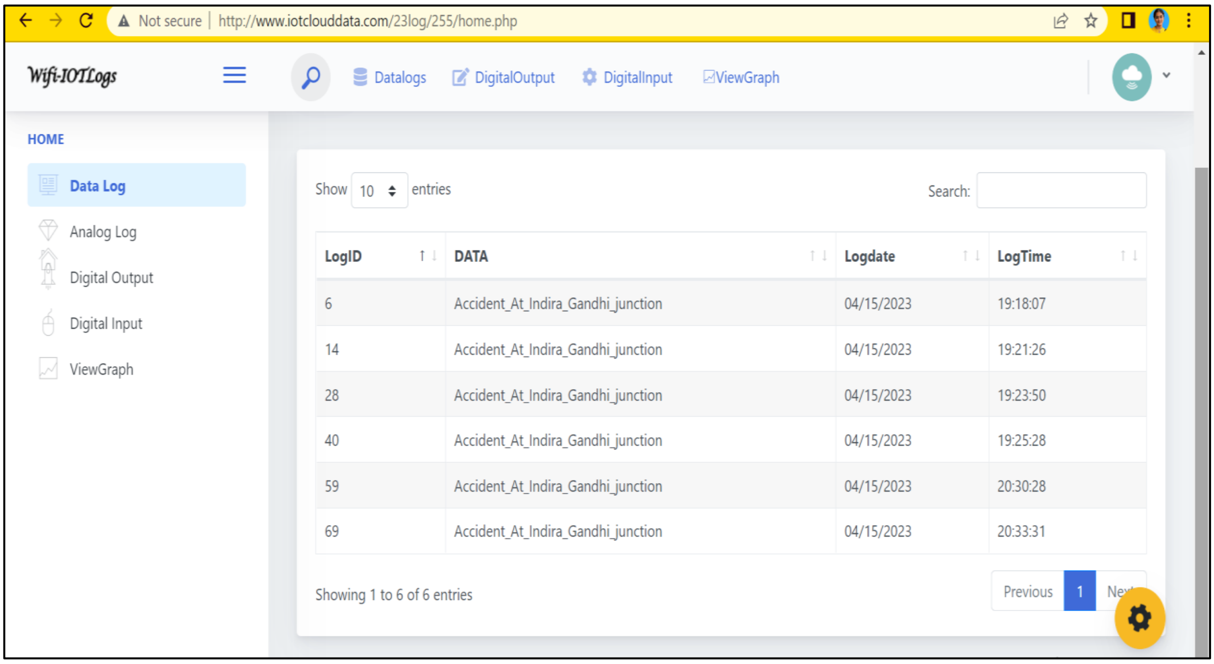

Fig 7 Server Log Data of Accident Events

ISSN No: -2456-2165 https://doi.org/10.38124/ijisrt/26apr1846

Upon successfully detecting an accident, the system immediately initiated an automated warning beep and simultaneously activated both software API and hardware communication channels. The Vonage API successfully delivered low-latency SMS alerts containing the incident logs to the designated emergency contacts.

Concurrently, via serial communication at a 9600 baud rate, the Python script transmitted a signal to the Arduino UNO board, which successfully processed the command to display an "Accident detected" alert. The integrated ESP8266 (NodeMCU) Wi-Fi module also established a stable internet connection and successfully uploaded the critical accident event details—including the unique Log ID, date, time, and specific location—directly to the centralized web server page (www.iotclouddata.com). The administrative server correctly updated and maintained a visual dashboard of these logged accident events for further tracking and future reference.

VII. CONCLUSION

Accident detection in real-time environments remains a complex challenge, which has limited the large-scale adoption of such systems. Existing in-vehicle solutions, although effective in providing timely alerts, are often constrained by high costs, hardware dependency, and lack of portability. The proposed intelligent accident detection system addresses these limitations by leveraging CCTV video feeds and a deep learning-based computer vision approach. By eliminating the need for expensive in-vehicle sensors and relying purely on model-based detection, the system proves to be more economical, scalable, and accurate. It enables immediate identification of accidents at the moment they occur, thereby significantly improving response time and overall reliability in emergency situations.

The system can be further enhanced by integrating advanced capabilities such as accident prediction, detection of reckless or intoxicated driving behavior, and automatic number plate recognition for better incident analysis. Connecting the system to a centralized database would enable automatic notification of victims' emergency contacts and insurance providers using vehicle registration details. Additionally, developing a dedicated mobile or desktop application could provide real-time alerts, live CCTV footage access, and navigation support for emergency responders. Future improvements may also include intelligent traffic management features, such as predicting vehicle movement near accident zones and proactively rerouting traffic to prevent congestion and secondary accidents, thereby making the system more comprehensive and impactful.

REFERENCES

[1]. Hadi Ghahremannezhad, Hang Shi, Chengjun Liu, "Real-Time Accident Detection in Traffic Surveillance Using Deep Learning", *IEEE International Conference on Imaging Systems and Techniques, ICIST* 2022.

[2]. M. P. Rathod, D. P. Gadhiya, "Real-Time Accident Detection and Alert System using Deep Learning on Edge Devices",*IEEE International Conference on Inventive Systems and Control(ICISC)*, 2020.

[3]. Bulbula Kumeda, Zhang Fengli, Ariyo Oluwasanmi,"Vehicle Accident And Traffic Classification Using Deep Convolutional Neural Networks",*16th IEEE International Computer Conference on Wavelet Active Media Technology and Information Processing,ICCWAMTIP* ,2020.

[4]. Lakshmy S, Renjith Gopan,et al, "Vehicle Accident Detection and Prevention using IoT and Deep Learning",*IEEE International Conference on Signal Processing, Informatics, Communication and Energy Systems (SPICES),*2022.

[5]. Nikhlesh Pathik, Rajeev Kumar Gupta, Yatendra Sahu,"AI Enabled Accident Detection and Alert System Using IoT and Deep Learning for Smart Cities", *MDPI journal on sustainability*,2022.

[6]. Mohammed Imran Basheer Ahmed, Rim Zaghdoud, Mohammed Salih Ahmed ,"A Real-Time Computer Vision Based Approach to Detection and Classification of Traffic Incidents", MDPI Journal on Big data and Cognitive Computing, 2023.

[7]. Renu, Durgesh Kumar Yadav , Iftisham Anjum ,Ankita, "Accident Detection using Deep Learning: A Brief Survey", *International Journal of Electronics Communication and Computer Engineering* ,Vol 11, Issue 3, ISSN : 2249–071X, 2020.

[8]. Ghosh, S., Sunny, S.J. and Roney, R "Accident detection using Convolutional Neural Networks", IEEE International Conference on Data Science and Communication (Icon DSC) pp. 1-6, 2019.