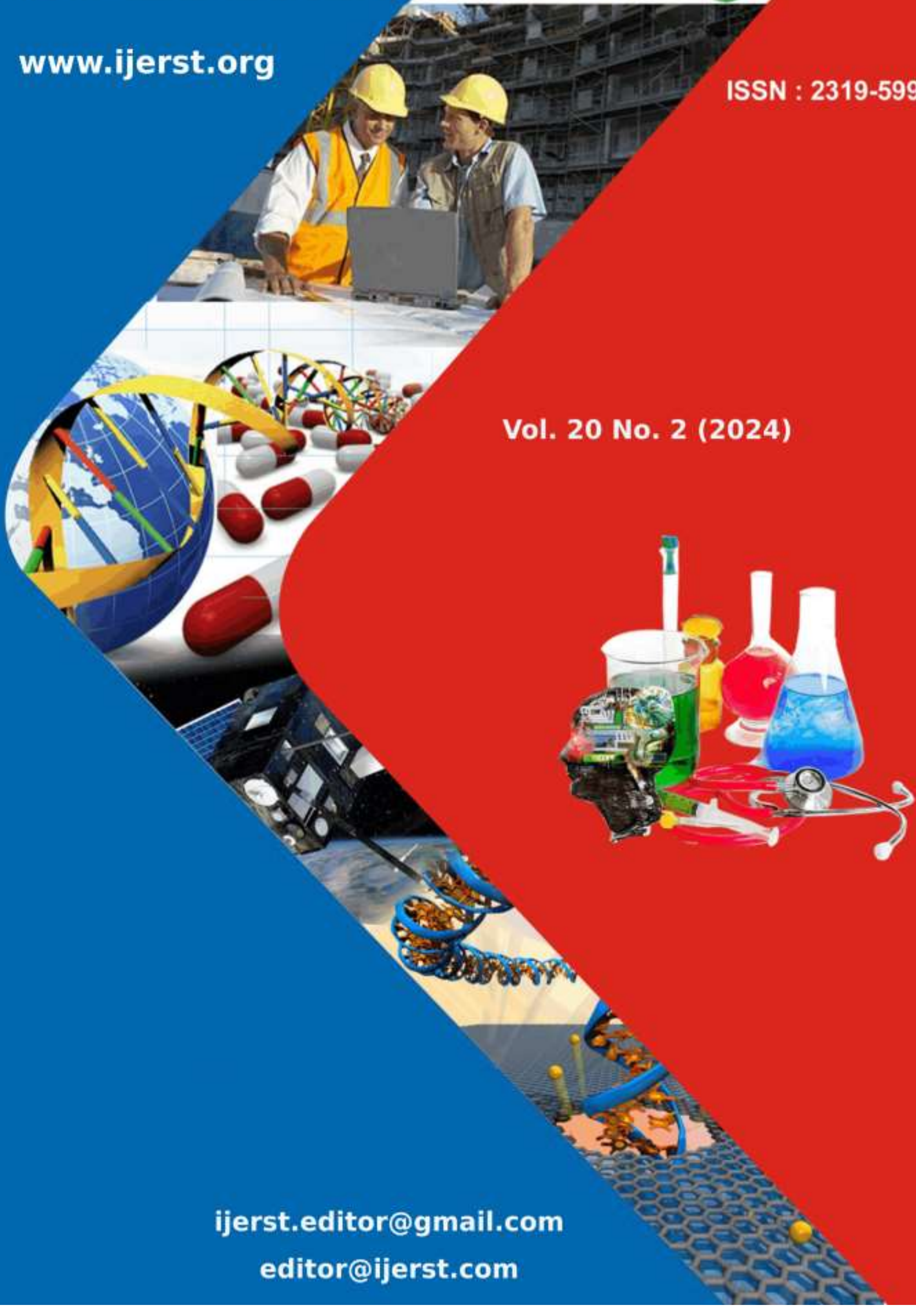
International Journal of Engineering Research and Science & Technology

www.ijerst.org

ISSN : 2319-5991



Vol. 20 No. 2 (2024)

ijerst.editor@gmail.com

editor@ijerst.com

ISSN 2319-5991 www.ijerst.com
Vol. 20, No. 2, 2024

Research Paper

INTELLIGENT ACCIDENT ALERT SYSTEM USING AI FOR TRAFFIC SURVEILLANCE VIDEOS

N. Sravani[1], Varala Vikas[2], Kathi Kalyan[2], Banothu Nagaraju[2], Rangisetti Adithya[2]
[1]Assistant Professor, [2]UG Student, [1,2] Department of Information Technology
[1,2]CMR Engineering College (UGC-Autonomous), Medchal Rd, Kandlakoya, Telangana 501401

Received: 07-3-2024 Accepted: 12-4-2024 Published: 22-4-2024

ABSTRACT

Accident detection from CCTV footage is a vital advancement in computer vision, contributing significantly to public safety. Road and public area accidents can have serious consequences, making timely detection crucial for rapid emergency response and potential life-saving interventions. Traditional detection systems primarily relied on motion detection algorithms and predefined rules to identify incidents. These systems triggered alarms when unusual activity was detected, but they were often prone to false positives and lacked the ability to differentiate between different types of incidents accurately. Additionally, manual monitoring proved inefficient, as it required constant human supervision, leading to potential delays in response times.

With advancements in machine learning, particularly convolutional neural networks (CNNs), the process of accident detection has become more automated and precise. Machine learning models can recognize complex patterns and features within images and video sequences, making them highly effective for accident detection tasks. The increasing number of accidents and the demand for swift emergency responses have highlighted the necessity for automated solutions. Manual monitoring is not only resource-intensive but also prone to human error, especially when attention is divided across multiple screens and locations.

Automated accident detection using machine learning enables real-time monitoring, significantly improving response efficiency. This research presents a customized approach to detecting accidents from CCTV surveillance footage using machine learning techniques. By replacing traditional rule-based methods with intelligent, data-driven algorithms, the system enhances accuracy and reduces false alarms. The integration of machine learning for accident detection revolutionizes public safety by ensuring immediate alerts and swift emergency responses, ultimately contributing to safer roads and public spaces.

1.INTRODUCTION

Over 1.3 million deaths happen each year from road accidents, with a further of about 25 to 65 million people suffering from mild injuries because of road accidents. In a survey conducted by the World Health Organization (WHO) on road accidents based on the income status of the country, it is seen that low and middle-income or developing countries have the highest number of road accident related deaths. Developing countries have road accident death rate of about 23.5 per 100,000 population, which is much higher when compared to the 11.3 per 100,000 population for high-income or developed countries [1]. Over 90% of road traffic related deaths happen in developing countries, even though these countries have only half of the world's vehicles. In India, a reported 13 people are killed every hour as victims to road accidents across the country. However, the real case scenario could be much worse as many accident cases are left unreported. With the present data, India is on the way to the number one country in deaths from road accidents due to the poor average record of 13 deaths every hour, which is about 140,000 per year [2]. An accident usually has three phases in which a victim can be found. First phase of an accident is when the death of the accident victim occurs within a few minutes or seconds of the accident, about 10% of accident deaths happen in this phase. Second phase of an accident is the time after an hour of the accident which has the highest mortality rate (75% of all deaths). This can be avoided by timely help reaching the victims. The objective is to help accident victims in this critical hour of need. Third phase of an accident occurs days or weeks after the accident, this phase has a death rate of about 15% and takes medical care and resources to avoid. The main objective is to incorporate a system which is able to detect an accident form video footage provided to it using a camera. The system is designed as a tool to help out accident victims in need by timely detecting an accident and henceforth informing the authorities of the same. The focus is to detect an accident within seconds of it happening using advanced Deep Learning Algorithms which use Convolutional Neural Networks (CNN's or ConvNet) to analyze frames taken form the video generated by the camera.

We have focused on setting up this system on highways where the traffic is less dense and timely help reaching the accident victims is rare. On highways we can setup CCTV camera's placed at distance of about 500 meters which act as a medium for surveillance, on this camera we can set up the proposed system which takes the footage from the CCTV camera's and runs it on the propsed accident detection model to detect accidents. In this system, we have a Raspberry Pi 3 B+ Model which acts as a portable and remote computer to be set up on a CCTV camera. For demonstration purposes, we will be using a Pi Camera which can be directly set up on a Raspberry Pi. We have pre-trained an Inception v3 model to be able to detect accidents by training it on two different sets of images and sequence of video frames. The images and video frames are 10,000 severe accident frames and 10,000 non-accident frames. The Inception v3 algorithm can now detect an image or frames of a video to be an accident frame by up to 98.5% accuracy. This model was then implemented on a Raspberry Pi using TensorFlow, OpenCV and Keras. When a video is shown to the Raspberry Pi through the Pi camera, it runs each frame of the video through the model created and then predicts whether the given frame is an accident frame or not. If the prediction exceeds a threshold of 60% or 0.6 the Raspberry Pi then initiates the GSM module setup with it to send a message to the nearest hospital and police station, informing them about the accident which has been detected with the timestamp of when it occurred, the location of where it occurred, and the frame at which the accident was detected for further analyses. Also, an emergency light lights up. The system we have made can detect accidents to an accuracy of about 95.0%. It can be done on a Raspberry Pi which is a card-sized computer, which makes it easily portable and remote. The system developed can act as a reliable source of information in detecting accidents which can be done automatically. This project would help us in reducing the ginormous number of road accident related deaths that occur in our country.

Keywords: Accident Detection, Computer Vision, Machine Learning, Convolutional Neural Networks (CNNs), Real-time Monitoring, Public Safety.

2. LITERATURE SURVEY

2.1 An enhanced accident detection and victim status indicating system: Prototype

Authors: Prabakar, S., et al.

Abstract: In the speedy moving world, nobody is ready to look what's happening around them. Even when there occurs an accident nobody cares about it. This is an intention to implement an innovative solution for this problem by developing an Enhanced Accident detection System for Indicating Victim Status from the accident zone. This system has been developed and implemented using the biomedical smart sensors and microcontroller based mobile technology integrated with the evolving LabVIEW platform. The system will automatically identify the accident, then immediately transmit the location of the accident and the status of the physiological parameters of the victims to the emergency care center phone number through Short Message Service (SMS). The victim's physiological parameters such as body temperature, Heartbeat, Coma stage recovery status have been transmitted in the SMS. So the proposed system ensures that to reduce the human death ratio by accidents. When the accident occurs and realizes that there is no severe collision, then the person involved in accident has to press the switch provision which has been made to indicate that the accident is diminutive, and no communication will be established i.e. no further alarming SMS has been transmitted.

2.2 Car Accident Detection and Notification System Using Smartphone

Authors: Hamid M. Ali,Zainab S. Alwan

Abstract: Every day around the world, a large percentage of people die from traffic accident injuries. An effective approach for reducing traffic fatalities is: first building automatic traffic accident detection system, second, reducing the time between when an accident occurs and when first emergency responders are dispatched to the scene of the accident. Recent approaches are using built-in vehicle automatic accident detection and notification system. While these approaches work fine, they are expensive, maintenance complex task, and are not available in all cars. On the other hand, the ability to detect traffic accidents using smartphones has only recently become possible because of the advances in the processing power and sensors deployed on smartphones. Most of the smartphone-based accident detection systems rely on the high speed of the vehicle (extracted from the smartphone GPS receiver) and the G-Force value (extracted from smartphone accelerometer sensor) to detect an accident. As many references assure that 90% of road-traffic accidents occur at low speed of the vehicle. Hence, in addition to the high-speed accident detection, this paper concentrated on low speed car accident detection. The main obstacle that encounters the low-speed accident is how to differentiate whether the user is inside the vehicle or outside the vehicle, walking or slowly running. The effect of this obstacle is minimized, in this work, by a proposed mechanism that distinguishes between the speed variation of low-speed vehicle and walking or slowly running person. The proposed system consists of two phases; the detection phase which is used to detect car accident in low and high speeds. The notification phase, and immediately after an accident is indicated, is used to send detailed information such as images, video, accident location, etc. to the emergency responder for fast recovery. The system was practically tested in real simulated environment and achieved quite very good performance results.

2.3 Smart Car: An IoT Based Accident Detection System

Authors: Arif Shaik,Natalie Bowen,Jennifer Bole,Gary Kunzi

Abstract: The Internet of Things (IoT) offers limitless possibilities to both the public and private sectors. Automobile manufacturers are interested in IoT applications to increase the safety of their vehicles, to meet customers' demands and ultimately to offer cutting-edge products which maximize profit. The healthcare industry is concerned with how the IoT can improve the speed and accuracy of communication. This paper describes the feasibility of equipping a vehicle with technology that can detect accident and immediately alert emergency personnel. When there is a car accident someone has to actively seek help such as calling 911 for emergency services. There is no automatic notification to the police, ambulance, friends, or family. The Internet of Things (IoT) can be used to produce an automatic notification and response to the scene. A signal from an accelerometer and a GPS sensor are automatically sent to the cloud and from there, an alert message will be received by whoever is subscribed to that car. The signal will indicate the severity of the accident and the GPS location. The ambulance will use the GPS coordinates to get to the scene quickly.

2.4 Vehicle to Vehicle communication

Authors: Shagufta Ali

Abstract: Vehicle to Vehicle communication A robust wireless network of connected vehicles is needed to enable future telematics and infotainment applications in the vehicular domain. We need to focus on terms reliable and continuous system performances as vehicle to vehicle (V2V) faces a highly dynamic time-varying channel conditions and varying vehicle network topology. This paper mainly focuses on Wireless communication technologies used in v2v communication. Keeping in mind the term's reliability, scalability, latency and throughput of the system, the technologies we have focused on are Cellular vehicle to everything communication (C-V2X), 4rth Generation Long term evaluation (4G-LTE) and dedicated short range communication (DSRC). Also, we have discussed the technical challenges v2v is facing. Further, we propose three methods related to high-speed vehicles mobility and complex channel environment along with vehicle network topological variations Loss differentiation Rate Adaption (LORA) is a scheme proposed to approximate the average packet loss rate (PLR) for every sender and select a data rate based on PLR. Exponential Effective SNR Mapping (EESM) and Mutual Information Effective SNR Mapping (MIESM) are two Adaptive modulation and coding mechanisms used for data rate selection according to current channel conditions. In order to adjusts and modify the data rate between vehicles according to the number of connectable vehicles nearby, a physical topology-triggered adaptive transmission scheme is proposed. Reliability and more continuous data transmission for V2V communications can be ensured by these methods and it can be proved by Extensive evaluation results. As a result, the effective improvement in performance on network throughput can be observed.

2.5 SOFTWARE ENVIRONMENT

Python is a high-level, interpreted scripting language developed in the late 1980s by Guido van Rossum at the National Research Institute for Mathematics and Computer Science in the Netherlands. The initial version was published at the alt. Sources newsgroup in 1991, and version 1.0 was released in 1994.

Python 2.0 was released in 2000, and the 2.x versions were the prevalent releases until December 2008. At that time, the development team made the decision to release version 3.0, which contained a few relatively small but significant changes that were not backward compatible with the 2.x versions. Pythons 2 and 3 are very similar, and some features of Python 3 have been back ported to Python 2. But in general, they remain not quite compatible.

Both Python 2 and 3 have continued to be maintained and developed, with periodic release updates for both. As of this writing, the most recent versions available are 2.7.15 and 3.6.5. However, an official End of Life date of January 1, 2020, has been established for Python 2, after which time it will no longer be maintained. If you are a newcomer to Python, it is recommended that you focus on Python 3, as this tutorial will do.

Python is still maintained by a core development team at the Institute, and Guido is still in charge, having been given the title of BDFL (Benevolent Dictator For Life) by the Python community. The name Python, by the way, derives not from the snake, but from the British comedy troupe Monty Python's Flying Circus, of which Guido was, and presumably still is, a fan. It is common to find references to Monty Python sketches and movies scattered throughout the Python documentation.

2.6 WHY CHOOSE PYTHON

If you're going to write programs, there are literally dozens of commonly used languages to choose from. Why chooses Python? Here are some of the features that make Python an appealing choice.

Python is Popular

Python has been growing in popularity over the last few years. The 2018 Stack Overflow Developer Survey ranked Python as the 7th most popular and the number one most wanted technology of the year. World-class software development countries around the globe use Python every single day.

According to research by Dice Python is also one of the hottest skills to have and the most popular programming language in the world based on the Popularity of Programming Language Index.

Due to the popularity and widespread use of Python as a programming language, Python developers are sought after and paid well. If you'd like to dig deeper into Python salary statistics and job opportunities, you can do so here.

Python is interpreted

Many languages are compiled, meaning the source code you create needs to be translated into machine code, the language of your computer's processor, before it can be run. Programs written in an interpreted language are passed straight to an interpreter that runs them directly.

This makes for a quicker development cycle because you just type in your code and run it, without the intermediate compilation step.

One potential downside to interpreted languages is execution speed. Programs that are compiled into the native language of the computer processor tend to run more quickly than interpreted programs. For some applications that are particularly computationally intensive, like graphics processing or intense number crunching, this can be limiting.

In practice, however, for most programs, the difference in execution speed is measured in milliseconds, or seconds at most, and not appreciably noticeable to a human user. The expediency of coding in an interpreted language is typically worth it for most applications.

Python is Free

The Python interpreter is developed under an OSI-approved open-source license, making it free to install, use, and distribute, even for commercial purposes.

A version of the interpreter is available for virtually any platform there is, including all flavors of Unix, Windows, macOS, smart phones and tablets, and probably anything else you ever heard of. A version even exists for the half dozen people remaining who use OS/2.

Python is Portable

Because Python code is interpreted and not compiled into native machine instructions, code written for one platform will work on any other platform that has the Python interpreter installed. (This is true of any interpreted language, not just Python.)

Python is Simple

As programming languages go, Python is relatively uncluttered, and the developers have deliberately kept it that way.

A rough estimate of the complexity of a language can be gleaned from the number of keywords or reserved words in the language. These are words that are reserved for special meaning by the compiler or interpreter because they designate specific built-in functionality of the language.

Python 3 has 33 keywords, and Python 2 has 31. By contrast, C++ has 62, Java has 53, and Visual Basic has more than 120, though these latter examples probably vary somewhat by implementation or dialect.

Python code has a simple and clean structure that is easy to learn and easy to read. In fact, as you will see, the language definition enforces code structure that is easy to read.

But It's Not That Simple For all its syntactical simplicity, Python supports most constructs that would be expected in a very high-level language, including complex dynamic data types, structured and functional programming, and object-oriented programming.

Additionally, a very extensive library of classes and functions is available that provides capability well beyond what is built into the language, such as database manipulation or GUI programming.

Python accomplishes what many programming languages don't: the language itself is simply designed, but it is very versatile in terms of what you can accomplish with it.

Conclusion

This section gave an overview of the **Python** programming language, including:

- A brief history of the development of Python
- Some reasons why you might select Python as your language of choice

Python is a great option, whether you are a beginning programmer looking to learn the basics, an experienced programmer designing a large application, or anywhere in between. The basics of Python are easily grasped, and yet its capabilities are vast. Proceed to the next section to learn how to acquire and install Python on your computer.

Python is an open-source programming language that was made to be easy-to-read and powerful. A Dutch programmer named Guido van Rossum made Python in 1991. He named it after the television show Monty Python's Flying Circus. Many Python examples and tutorials include jokes from the show.

Python is an interpreted language. Interpreted languages do not need to be compiled to run. A program called an interpreter runs Python code on almost any kind of computer. This means that a programmer can change the code and quickly see the results. This also means Python is slower than a compiled language like C, because it is not running machine code directly.

Python is a good programming language for beginners. It is a high-level language, which means a programmer can focus on what to do instead of how to do it. Writing programs in Python takes less time than in some other languages.

Python drew inspiration from other programming languages like C, C++, Java, Perl, and Lisp.

Python has a very easy-to-read syntax. Some of Python's syntax comes from C, because that is the language that Python was written in. But Python uses whitespace to delimit code: spaces or tabs are used to organize code into groups. This is different from C. In C, there is a semicolon at the end of each line and curly braces ({}) are used to group code. Using whitespace to delimit code makes Python a very easy-to-read language.

Python use [change / change source]

Python is used by hundreds of thousands of programmers and is used in many places. Sometimes only Python code is used for a program, but most of the time it is used to do simple jobs while another programming language is used to do more complicated tasks.

Its standard library is made up of many functions that come with Python when it is installed. On the Internet there are many other libraries available that make it possible for the Python language to do more things. These libraries make it a powerful language; it can do many different things.

Some things that Python is often used for are:

- Web development
- Scientific programming
- Desktop GUIs
- Network programming
- Game programming

3. PROPOSED SYSTEM

The proposed model is a fusion of CNN and LSTM layers for continuous video classification taken from a camera. The CNN part of the proposed model was mainly inspired by the Inception v3, but with certain tweaks it has fitted well to our training images. The LSTM layers were added to the existing Convolution Network to consider temporal features along with spatial features. This is further divided into the convolution and recurrent parts of the model. In a CNN-LSTM network, the CNN is primarily used for feature extraction from the images, which is passed on to the LSTM for sequence prediction. They are widely used in tasks like Activity Recognition, Image Description, Video Description etc.

Advantages Of Proposed System

1. Initially, the model had difficulties in predicting the right class since the only difference between cars and broken cars were dents, broken headlights etc.
2. Thus, random noise was being accounted during the prediction since not all the accident images were of high clarity.

4. RESULTS AND DISCUSSION

4.1 Implementation Description

The project implements an AI-powered approach for accident occurrence alerting from traffic surveillance videos using a Convolutional Neural Network (CNN). Here's a description of the implementation:

— **Graphical User Interface (GUI):** The project begins by creating a GUI using the Tkinter library in Python. The GUI serves as the interface for users to interact with the accident detection system. It includes buttons for loading and generating the CNN model, browsing system videos for testing, starting the accident detector, and exiting the application. Additionally, there are text boxes to display messages and the path of the loaded video.

— **Loading and Generating CNN Model:** The application allows users to load and generate the CNN model using a pre-trained model stored in the 'frozen_inference_graph.pb' file. The model is loaded using TensorFlow, and the GUI displays a message confirming the successful loading of the model.

— **Uploading Video for Testing:** Users can browse and select a video file from their system for testing the accident detection system. The selected video's path is displayed in the GUI, and the application is ready to start detecting accidents in the video.

— **Accident Detection Algorithm:** The core of the application is the accident detection algorithm, which runs in real-time on the selected video. It utilizes the loaded CNN model to detect potential accidents in the video frames. The algorithm extracts frames from the video, processes them using the CNN model, and analyzes the detected objects in each frame.

— **Collision Detection and Alerting:** The algorithm specifically looks for objects related to potential accidents, such as vehicles. It calculates the intersection area between bounding boxes of detected vehicles to determine if a collision is imminent. If the intersection area exceeds a predefined threshold, indicating a potential collision, the algorithm triggers an alert by displaying a message on the GUI and producing a beeping sound to alert the user.

— **Real-Time Video Display:** Throughout the accident detection process, the GUI displays the video stream in real-time, overlaid with any detected accidents or potential collisions. Users can observe the video feed and monitor the system's detection performance.

— **User Interaction and Control:** Users have control over the application through the GUI buttons. They can start and stop the accident detection process, load different video files for testing, and exit the application when finished.

4.2 Results and Description

Figure 1: Load & Generate CNN Model in the GUI This image depicts the graphical user interface (GUI) of a software application designed for loading and generating a Convolutional Neural Network (CNN) model. The GUI include options for users to upload pre-trained CNN models or generate new models based on their specific requirements. Additionally, there might be buttons or menus for users to specify model parameters, such as

architecture, hyperparameters, and optimization algorithms. Figure 2: Training CNN model with train data In this image, the GUI is displaying the process of training a CNN model using training data. The GUI include visualizations or progress bars indicating the status of the training process, such as the current epoch, training loss, and validation accuracy. Users have options to customize training settings, monitor performance metrics, and analyze model training results.

Figure 1: Load & Generate CNN Model in the GUI.

Figure 2: Training CNN model with train data.

Figure 3: Uploading video for Model Testing.

Figure 4: Model Detection of Accident Occurance.

Figure 3: Uploading video for Model Testing This image depicts a feature in the GUI where users can upload a video file for testing the trained CNN model. The GUI provide options for users to select a video file from their local storage or input a video URL for testing purposes. Additionally, there might be controls for users to play, pause, or analyze the video during testing.

Figure 4: Model Detection of Accident Occurrence In this image, the GUI is showing the detection results of the CNN model for accident occurrence in a video stream. The GUI display visual indicators, such as bounding boxes or overlays, highlighting regions in the video where accidents are detected. Additionally, there might be text notifications or alerts indicating the presence of accidents detected by the model. Users have options to adjust detection thresholds, review detection results, and take appropriate actions based on the model's

predictions.

5. CONCLUSION

Accidents are one of the most common problems that humanity faces daily, leading to loss of both life as well as property. The proposed system provides a very viable and effective solution to this problem. The proposed vehicle accident detection system can track an accident at its moment of occurrence and sends an instantaneous alert SMS regarding the accident to the nearby hospitals and police stations which includes details like timestamp and the geographical location. Unlike other systems in use, which consists of expensive sensors and unwanted hardware, the proposed system is much more cost effective and foolproof with a much-improved accuracy rate than its counterparts mainly due to a model-based approach. The experimentation, testing and validation has been carried out using images and the results show that higher sensitivity and accuracy is indeed achieved using this method, henceforth, making it a viable option for implementing this system in most of the state and national highways of the country.

REFERENCES

[1] "Global status report on road safety 2015", World Health Organization, 2019. [Online]. Available: http://www.who.int/violence_injury_prevention/road_safety_status/20 15/en/. [Accessed: 07- Mar- 2019].

[2] Prabakar, S., et al. "An enhanced accident detection and victim status indicating system: Prototype." India Conference (INDICON), 2012 Annual IEEE. IEEE, 2012.

[3] "Lexus Enform", Lexus, 2019. [Online]. Available: https://www.lexus.com/enform. [Accessed: 07- Mar- 2019].

[4] "OnStar Safety and Security Services", Onstar.com, 2019. [Online]. Available: https://www.onstar.com/us/en/services/safety-security/. [Accessed: 07- Mar- 2019].

[5] "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].

[6] C. Kockan, "Communication between vehicles" PhD thesis, Istanbul Technical University, 2008

[7] Zeng, Yuanyuan, Deshi Li, and Athanasios V. Vasilakos. "Opportunistic fleets for road event detection in vehicular sensor networks." Wireless Networks 22.2 (2016): 503-521.

[8] Szegedy, Christian, et al. "Rethinking the inception architecture for computer vision." Proceedings of the IEEE conference on computer vision and pattern recognition. 2016.

[9] Szegedy, Christian, et al. "Going deeper with convolutions." Proceedings of the IEEE conference on computer vision and pattern recognition. 2015.