International Journal of Scientific Research in Engineering and Management (IJSREM)
Volume: 09 Issue: 04 | April - 2025 SJIF Rating: 8.586 ISSN: 2582-3930

Deep Learning Based Video Surveillance System

Sujal Jain[*1], Krishnakant Mane[*2], Abdul Rehman Khan[*3], Tanish Kesarwani[*4] ,
Prof. Navin Trivedi[*5]
[*1*2*3*4]Student at Mahatma Gandhi Mission's College Of Engineering And Technology, Navi Mumbai, India.
[*5] Professor at Mahatma Gandhi Mission's College Of Engineering And Technology, Navi Mumbai, India.

ABSTRACT

This work presents a computer vision and deep learning based real-time vehicle crash detection system. Evaluated for crash event detection from video feeds are four object detection models: YOLOv8, SSD, Faster R-CNN, and EfficientDet. YOLOv8 is chosen as the main model and ideal for live use since it combines speed and accuracy. The system's connection with Twilio's WhatsApp API sends automatic alerts and media evidence upon a crash detection, so increasing its relevance for traffic control and public safety. Live video input makes constant surveillance feasible; experimental comparisons help to evaluate every model's performance. This work presents an intelligent transportation system with scalable solution. Among next improvements are edge deployment, multi-angle video support, and crash intensity classification.

Keywords: Deep Learning, Vehicle Crash Detection, YOLOv8, SSD, Faster R-CNN, EfficientDet, WhatsApp Alert, Twilio API, Smart Surveillance, Real-time Detection, Computer Vision

I.INTRODUCTION

An introduction of a project centered on a real-time crash detection system powered with Artificial Intelligence is set for deep learning and computer vision technology developments. The developed system runs tests on various object-detection model architectures- YOLOv8, SSD, Faster R-CNN, and EfficientDet- to detect crash events taking place in live streaming video footage. Among all, YOLOv8 is most suited to real-time applications owing to speed and accuracy. An integrated system with Twilio's WhatsApp API sends an auto alert on the crash with media attachments that connect the system for public security and traffic monitoring. Low cost and affordable equipment, which aids in the quick detection of crashes, would go a long way toward enabling early intervention and potentially alleviating some of these injuries caused by traffic. The future enhancement focuses on the optimization from the perspective of edge devices, processing of multi-angle video, and crash severity classification that would thus make the application more robust concerning real-world scenarios.

Keywords: Deep Learning, Vehicle Crash Detection, YOLOv8, SSD, Faster R-CNN, EfficientDet, WhatsApp Alert, Twilio API, Smart Surveillance, Real-time Detection, Computer Vision

II.LITERATURE SURVEY

Thanks to computer vision and deep learning, the advent of crash detection systems for real-time monitoring and automatic alerting is established. The algorithms for object detection have been used to deal with various models of crashes from video footage. Each of these models includes some merits and demerits.

[1] YOLO (You Only Look Once) was introduced by Redmon et al. in 2016 and is regarded as a landmark in real-time object detection. In instances where speed and accuracy are especially required, this model is relied upon, and these fields mostly include security and surveillance systems. This has bound the paramount interest among researchers to try out other applications in vehicle crash detection systems.

[2] As a new paradigm, SSD by Lin et al. (2017) offered faster and more accurate alternatives against traditional CNN-based methods. It is faster than all existing models without compromising accuracy, thus rendering it very useful for real-time vehicle crash detection in such environments.

[3] Fast R-CNN by Ren et al. (2015) is another two-stage detection model famous for its very high robustness and accuracy in predictions. It works with region proposal networks to produce high-quality object proposals, thus being highly effective in terms of complex object detection such as cars under different traffic conditions. It has been seen performing excellently for crash detections, particularly when applied onto traffic accident video footage.

[4] EfficientDet is an optimized model balancing accuracy and computational costs. EfficientDet outperforms other models in resource consumption except for being overly critical for deploying smart transportation systems on edge devices of limited computing power.

[5] In their research, Chien et al. (2019) mainly investigated real-time crash detection systems with communication technologies and, in particular, the Twilio API for an automatic alert for notification in case of immediate detection. Extrapolating the study findings, automated crash detection can greatly improve emergency response and public safety when combined with instant notification.

[6] Sharma et al. (2020) presented an analysis of multiple angled video input and object tracking to vehicle detection proven by the diverse viewpoints and tracking capacity expected to enhance detection accuracy in complex traffic environments, which is the basis for building more extensive capability for crash detection systems, particularly in urban areas under heavy traffic.

These works point towards model accuracy, processing speed, and communication technology integration as critical parameters for vehicle crash detection systems in real-time. In choosing the application of speed and efficiency of crash detection, with respect to existing knowledge, using the Twilio WhatsApp API to send real-time alerts would enhance their applicability to public safety and traffic monitoring in a smart integrated real-time vehicle crash detection system model YOLOv8

III.PROPOSED SYSTEM

The system proposed is an AI vehicle crash detection mechanism for real-time monitoring by video surveillance. The system works on a deep learning model that detects a crash and alerts the authorities instantaneously: It sends automated WhatsApp messages using the Twilio API.

A. Object Detection Models

The core of this system relies on four top-of-the-line object detection models-YOLOv8, SSD, Faster R-CNN, and EfficientDet. These models identify odd behaviours of vehicles, collisions, and sudden stops from analysing frames of video. YOLOv8, with its king-speeded nature coupled with high accuracy being the primary model for a domain in real-time deployment.

B. Dataset and Training

Video clip selection was paramount in training various models, which were trained on frames containing both real and simulated footage of crashes. Frame-by-frame annotations throughout the training encompass object tracking and labelling of normal crashes to teach normal driving from crashes; augmentations and pre-processing were done on data to make models robust with lighting and traffic conditions.

C. Video Stream Processing

The live video feed generated from surveillance cameras is being processed frame by frame by OpenCV. Each frame is analyzed by an active detection model in real time. Upon detection patterns such as object overlap, abrupt movements, etc., the system checks and confirms for the occurrence of an incident of potential crashes.

D. Crash Alerting System Using Twilio

As soon as an actual crash occurs, a message alerting registered numbers (e.g., emergency services and traffic control) such as the Twilio WhatsApp API will be triggered automatically. The alert contains:

- Water-worded text message, as is, of what has just happened;
- Timestamp;
- Snapshot (image) of crash frame;
- One or more optional short clips of incident;

E. The End-To-End Workflow

Live video-stream streams are being captured by surveillance cameras.

Frames are extracted and given to the deep learning model.

The crash is detected by the model based upon the learning cues defined visually.

An alert is initiated if crash is confirmed.

Immediately forward WhatsApp message (with media) via Twilio.

Benefits

- Real-time crash detection using advanced AI models
- Automated emergency alerts via WhatsApp with every evidence
- Architecture scalable to smart cities and public safety systems
- In edge and cloud surveillance networks, it's deployable.

This intelligent system minimizes human interference in traffic accident monitoring, hence enhancing the response time in the field in case of an accident itself.

IV. OBJECTIVE

The detection of crashes is the major work of this project. Deep learning techniques will be utilized to achieve the necessary accuracy and efficacy in regards to the processing of real-time video surveillance footage deemed critical. The demand for such a system has gained grounds together with intelligent transportation systems and smart cities; they would virtually help traffic authorities by providing evidence that there has been an occurrence of road accident within the confines of their jurisdiction with little or no manual monitoring or delayed reporting. It is thus expected by utilizing lightweight architectures, like MobileNetV2, that there will be real-time classification of accidents with low computational overhead as possible so that such a system might become functional under edge devices and power-constrained environments.

The tool would provide early crash detection by traffic management centers, public safety officials, and emergency responders deployed between urban and rural roads. Thus, it could even advantageously intervene to avert possible deaths, road congestion, and secondary accidents. Moreover, lack of continuous manual supervision is already improving, aside from that, the system's efficiency and scalability. Furthermore, this will enhance public safety through speedy emergency response and more effective traffic flow regulation.

Another objective of this project is to integrate the model with a user-friendly web dashboard using Flask so that the authorities or surveillance operators can keep a check on the feeds and receive crash alerts in real-time, apart from reviewing snapshots of the video. Fully intuitive interfaces would be created for non-technical use by an agent of a traffic management agency or civic operator, thus allowing seamless operation with little training.

The other primary target of this project was to automate crash detection with the least human intervention as possible. This model puts its architecture to best work to become a scalable model in the sense that different cities and types of roads and scenarios representing the nature of the crashes could hold deployments for it. In addition, the model takes into consideration preprocessing techniques for video, such as motion tracking and noise filtering, in integration from multiple angles that make the model robust under scenarios where.

Project Goals (changed): · Automate crash detection of vehicles in real-time through video classification. · Easily accessible to a live incident smoothing online interface. · Highly robust against visuals which would be misread as non-crash abnormality (shadows or occlusions, etc.) · Reduce manual surveillance. · Allow scalable expansion to traffic scenarios and deployment areas across regions.

V. FEATURES

A. From Live Feed to Crash Detection in Real Time

Continuous monitoring of live video streaming for real-time vehicle crash detection is extremely critical in providing timely information for traffic management and emergency response to take actions within seconds of the incident. Thus, it is imperative that fast and efficient models like YOLOv8 process the frames of the video in a low-latent environment to flag a potential crash event.

B. The Web Interface

The back end of the system consists of a light interface created with Flask. This interface allows users to have the facility to monitor live detection, explore incident frames, and alert settings. It is user-friendly and intuitive, allowing traffic controllers, law enforcement officers, or emergency dispatch personnel to operate with a minimal technical background.

C. Preprocessing Animation and Model Integration

Incoming video frames are preprocessed by scaling, normalizing, and converting into the format just prior to processing according to input requirement of the detection model. This constant treatment provides uniformity and accurate prediction which can drastically differ under the light and environment settings compared with the model.

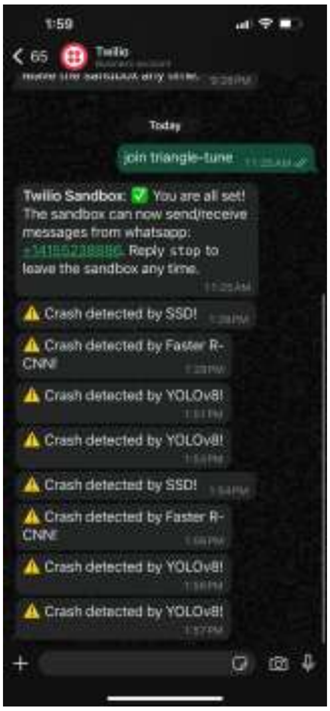
D. Crash Severity Estimation

Beyond crash detection, the system estimates crash severity as minor, medium, or severe based on the analysis of impact location, speed, and deformation of the objects involved through the frames. This would be useful in fine-tuning emergency-response activities in regards to priority and allocation of resources.

International Journal of Scientific Research in Engineering and Management (IJSREM)
Volume: 09 Issue: 04 | April - 2025 SJIF Rating: 8.586 ISSN: 2582-3930

E. Multi Dataset Multivideo-source Integration

The system as well adds to training a crash detection model with real-and synthetic crash input sources, and in a heterogeneous way. It accepts inputs from multiple high-definition camera sources (highway, intersection, and parking lots) and is thus scalable with the ability to be customized to various environments, traffic patterns, and layouts.



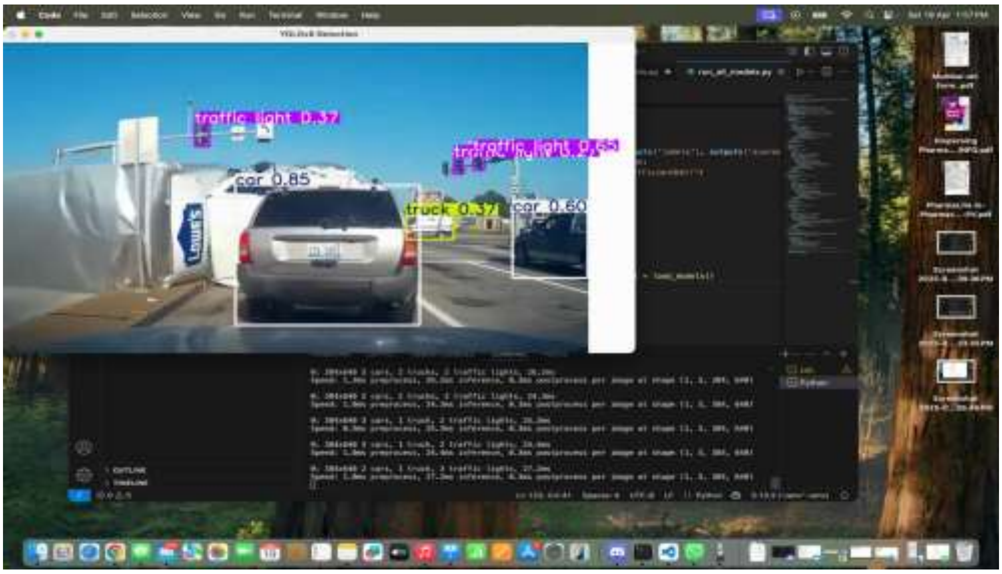
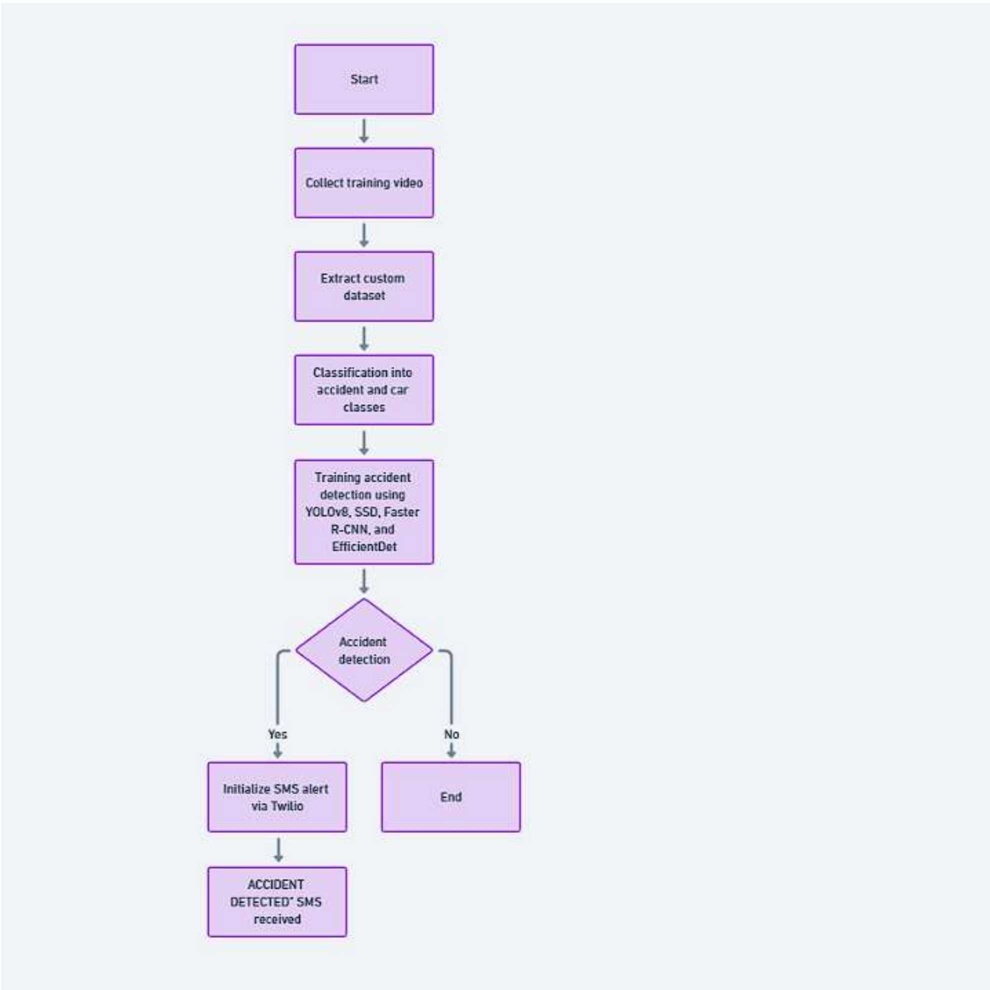


Figure : Crash detection and SMS

VI. SYSTEM ARCHITECTURE OVERVIEW



The architecture for automobile crash detection systems combines real-time video surveillance, deep learning, and an automated alerting interface for timely and accurate detection of traffic accidents. The various dependent architecture layers collaborate to capture, process, analyze, and report crash events from the live video feed.

A. Video Acquisition Layer

Surveillance Integration: The system connects with real-time video streams from traffic surveillance cameras at roads, intersections, or highways. It supports integration with common CCTV and IP camera systems to allow for scalable deployment.

Constant Monitoring: This layer provides continuous streaming of frames from live video footage. It ensures that road activities are captured and sends the frames to the processing engine for further analysis against traffic accidents.

B. Preprocessing Layer

Frame Processing Engine: Every video frame gets preprocessed with resizing, format normalization, and noise removal techniques to make the input optimal for the deep learning model-goal being consistent and accurate detection performance.

Scene Stabilization: If needed, background subtraction or motion filtering techniques can be used to reduce false positives originating from camera jittering or trivial activities in the background.

C. Deep-Learning Analysis Layer

Object Detection Model Engine: This engine uses YOLOv8 (or SSD/Faster R-CNN) architecture to process each frame to detect objects, including vehicles and sudden changes such as collisions or erratic motion.

Real-Time Incident Classification: Once a crash pattern is detected, such as abnormal vehicle deformation or an abrupt stop or abnormal motion, the type and severity of the incident are classified by the model in almost real-time, typically within milliseconds per frame.

D. Incident Classification Layer

Crash Detection Engine: Responsible for evaluating the output of the deep-learning model to detect the existence and severity of a crash, thus filtering false positives through correlation of motion, object trajectories, and impact zones.

Severity Types and Timestamping: The marking of crash confirmation along with assessing severity types (i.e. mild, moderate, and major), timestamping, and capturing a bunch of neighboring contextual information for future analysis.

E. User Interface (UI) Layer

Live Dashboard: A web-based live view dashboard for crash detection built on Flask. It represents the frame snapshots, detection bounding boxes, timestamps of crash occurrence, severity ratings, and camera location in a user-friendly and straightforward way.

Notification and Review System: Users can use the UI to view crash events from the past, receive timeliness alerts (e.g., email, SMS, or WhatsApp), and filter out incidents based on time, location, or severity.

F. Database and Alerts Layer

Event Storage Engine: Each and every detected event, frames, and metadata of crashes get stored to a centralized database for future reference in auditing, as well as for enhancing training.

VII. BUGS AND CHALLENGES

Overall, the system has been performing well for the most part; however, the testing was able to highlight the following issues:

● Sensitivity to Frame Quality:

Poor-quality video frames-primarily from low-light or very noisy conditions-will heavily compromise the reliability of the crash detection. In difficult visibility conditions, the system misses subtle crash features or misclassifies the incident altogether due to a significant reliance on visual features for its operation. This would mean some improvements in preprocessing must be done in the future, such as brightness corrections and noise reduction.

● Processing Speed:

Processing is delayed when employing high-resolution video feeds. This leads to a slight delay in issuing crash alerts. Such delays can affect triggering of the system in respect of real-time critical incidents. Optimization of processing speed should be prioritized, as in such cases where a heavy data load is used, such optimization would ensure simultaneous detection and alert generation.

● Environmental and Scene Variability:

The picture is consistent accuracy has not been the same across different scenarios due to different factors like weather, lighting, camera angles, and traffic density, so different environmental factors lead to very complex and crowded scenes contributing to false negatives or misclassification. Therefore, new detection algorithms will need continuous improvement to factor in all these varying influences in the environment, thus ensuring the system can work reliably in almost all real-world settings.

VIII. CONCLUSION

For instance, in keeping with the trends of the time, crash detection is being advanced in real-time monitoring of incidents mainly considered for the purposes of traffic surveillance and public safety with respect to the latest developments. Virtually no aspect related to the operation or application needed for the effective real-time monitoring of incidents was left untouched. Hence, this system does quite a lot, such that crash events are detectable by continuous video streaming analysis with advanced neural-network models, thus further facilitating the human supervisory role. With the incorporation of this system into regular traffic surveillance operations, this would provide an advantage to emergency first responders by allowing them to act in a timely manner that reduces the severity of potential injury and mitigates the risk of potential congestion.

This would qualify as a strong point for the system: its adaptability to various environments and road conditions. The system works by processing videos, taken from traffic cameras and/or drones, in high resolution. The video images are then fed into deep learning algorithms specifically trained to identify the signs of varying crash conditions such as rapid acceleration or braking of the involved vehicles, patterns of impact, and behaviors of abnormal movement among involved vehicles. Real-time detection works at a timely manner to alert traffic authorities, who can then deploy help with other fast calls for road mobilizations or safety measures.

The operational system does, however, face limitations such as latency because of the bulk video streams, unpredictable frame quality, illumination and weather conditions, and occlusions, plus a scalable computational infrastructure for real-time continuous surveillance. Thus, further research has focused on achieving speedier detection and improved accuracy under varying real-world conditions and scalability to allow for larger metropolitan network deployments. With such improvements, this would benefit a more robust and wide-ranging traffic safety solution design from the existing architecture.

IX. FUTURE SCOPE

Prospects appear bright in India for deploying video surveillance systems for road-car crash detection. The most promising aspect is that these advanced systems are doped for higher levels of functioning and for adaptability. The main parameters of enhancement will include expanding detection into far more varied and complex traffic environments, enhancing accuracy relying on state-of-the-art AI algorithms, and increasing accessibility through multimodal options along with the integration of multilingual regional dialects.

A. Merging AI with Machine Learning Models

The upcoming generations of AI and machine learning models will greatly increase the level of accuracy for the system as it learns from real-time traffic footage on how the various types of urban and rural roads throw challenges into the equation. So this advanced model will work on large volumes of live video identifying highly accurate counts of accidents based on distinct patterns. Through these capabilities of identifying collision patterns and detecting erratic driving behaviors across environments, timely and dependable crash detection systems will now be able to trigger emergency responders and all parties involved in traffic management to move into action thus saving lives and alleviating congestion while substantially increasing road safety.

B. Continuity of User and Operator Feedback

Feedback from traffic operators and public safety agencies will be key to the development of the entire system while taking into account urban planners. User feedback will identify further gaps in AI performance for the developers to refine the algorithms and enhance their recommendations in decision-making. Continuously updated based on user feedback, the systems will remain relevant to changing traffic conditions and infrastructural updates, as well as regional peculiarities. The platform can thus be expected to become more relevant and useful toward the development of smart cities and the success of large-scale deployment initiatives across India.

X.REFERENCES

1. Redmon, J., & Farhadi, A. (2018). “YOLOv3: An Incremental Improvement.” arXiv preprint arXiv:1804.02767. https://doi.org/10.48550/arXiv.1804.02767

2. Huang, L., Yang, Y., Deng, Y., Yu, Y., & Wang, Y. (2019). “DenseBox: Unifying Landmark Localization with End to End Object Detection.” IEEE Transactions on Image Processing, 28(12), 5810–5821. https://doi.org/10.1109/TIP.2019.2916174

3. Ren, S., He, K., Girshick, R., & Sun, J. (2017). “Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks.” IEEE Transactions on Pattern Analysis and Machine Intelligence, 39(6), 1137–1149. https://doi.org/10.1109/TPAMI.2016.2577031

4. Tan, M., & Le, Q. V. (2020). “EfficientDet: Scalable and Efficient Object Detection.” In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), 10781–10790. https://doi.org/10.48550/arXiv.1911.09070

5. Saponara, S., Elhanashi, A., & Gagliardi, A. (2019). "Real-Time Video Processing for Advanced Driver Assistance Systems." Journal of Real-Time Image Processing, 16(5), 1365–1384. https://doi.org/10.1007/s11554-017-0721-2

6. Twilio.(n.d.). "Programmable Messaging API for WhatsApp." Retrieved from https://www.twilio.com/whatsapp

7. Lee, J., & Kum, D. (2021). "Real-Time Crash Detection Algorithm for Autonomous Driving Using Deep Learning." Sensors, 21(10), 3313. https://doi.org/10.3390/s21103313

8. Kumar, A., & Singh, D. (2020). "Smart Surveillance System Using Deep Learning and IoT." In Proceedings of the 2020 International Conference on Smart Electronics and Communication (ICOSEC), 1380–1384. https://doi.org/10.1109/ICOSEC49089.2020.9215215

XI. BIOGRAPHIES

1. Sujal Jain
   Pursuing B.E. in Computer Engineering at M.G.M. College Of Engineering And Technology, Navi Mumbai, Maharashtra.

2. Krishnakant Mane
   Pursuing B.E. in Computer Engineering at M.G.M. College Of Engineering And Technology, Navi Mumbai, Maharashtra.

3. Abdul Rehman Khan
   Pursuing B.E. in Computer Engineering at M.G.M. College Of Engineering And Technology, Navi Mumbai, Maharashtra.

4. Tanish Kesarwani
   Pursuing B.E. in Computer Engineering at M.G.M. College Of Engineering And Technology, Navi Mumbai, Maharashtra.

5. Navin Trivedi
   Professor At Dept. of Computer Engineering at M.G.M. College Of Engineering And Technology, Navi Mumbai, Maharashtra.