MediSync: An Integrated Healthcare Platform for Outpatient Scheduling and CNN-based Dynamic Emergency Detection

Yash Aditya Mishra, Aman Sharma, Aagam Jain and Shriya Shreyaskar

Department of Computer Science and Engineering,
Raj Kumar Goel Institute of Technology, Ghaziabad, India

Abstract: **Nowadays, hospitals adopt an online booking system for outpatient department OPD appointment, yet most of them are for none urgent services and cannot respond to emergencies. The reporting of accidents on roads and medical emergency have been delayed by manual process and disconnected dispatch procedures. This paper presents a unified MERN based platform that couples OPD appointment scheduling with CNN based Dynamic Emergency Response system, God's Eye. The emergency module uses a YOLOv8 Deep Learning model for the real-time detection of vehicle accidents from live or uploaded video streams. Emergency alerts are detected, they are automatically generated and then sent to a centralized-dashboard that enables immediate ambulance dispatches using geospatial optimization. The architecture is a decoupling solution for AI inference and healthcare management services, scaling well with low latency. Experimental results in simulated surveillance feeds prove that it is possible to develop a real time accident detection and automatic emergency coordination system. The program illustrates how much computer vision based intelligence can help in reducing emergency response time and remain close to the entire digital healthcare ecosystem.**

Keywords: Convolutional Neural Networks, YOLOv8, Accident Detection, Emergency Response System, MERN Stack, Smart Healthcare

I. Introduction

OPD scheduling system has recently gained significant importance in hospital and any kind of healthcare centre. Online appointment booking is time saving for patients and cuts down on administrative workload and enhances hospital's overall performance [1], [2]. Recent studies have shown that web OPD systems result in significantly lower waiting time and increased patient satisfaction compared to walk in and telephone based methods [3]. Even as technology has progressed, emergency medical response is one of the weakest links in many digital health systems. Manual reporting by bystanders has implications on road traffic accidents and sudden medical emergencies, contributing to delays during the golden hour [4]. The World Health Organization (WHO) estimates that delayed emergency response is one of the reasons for trauma related deaths, worldwide [5]. Given that there is an increasing availability of CCTV infrastructure in urban spaces, the possibility for computer vision based monitoring to transition from passive surveillance to proactive emergency detection arises. The state of the art in deep learning, e.g., CNNs have become fast enough to process video streams in real time with high accuracy, even in challenging traffic scenes [6].

In this paper, it present an integrated framework that implements an MERN based OPD appointment system and a Dynamic Emergency Response engine powered by CNN. Although the OPD system ensures healthcare readiness, the main novelty of this work is in its CNN based module that automatically detects accidents using YOLOv8 and triggers emergency response without human intervention.

II. Related Work

OPD Appointment Scheduling Systems

The effects of online scheduling on health care efficiency are not yet fully understood. Samadbeik et al. [1] proved that OPD booking applications have led to savings in routine administrative tasks and enable the healthcare resources to be accessed widely. Recent studies have emphasized the scalability and usability of web based scheduling systems based on new architectures [2], [3]. However, such systems focus on non-emergency processes and lack support for real time emergency coordination.

Accident Detection and Emergency Response

Early accident detection solutions were based on in sensor approaches by using an embedded accelerometer or GPS module in a vehicle or mobile phone [7]. Sensor based techniques are suitable for controlled environments, but do not work well when sensing devices have been destroyed or malfunctioned.

Vision has recently become popular in the wake of the availability of efficient surveillance cameras. In the traditional computer vision algorithms, the robustness in different lighting conditions and traffic environments is not satisfactory [8]. CNN based methods have been proven to be superior for traffic incident and anomaly detection [6], [9].

YOLO series object detectors are a group from the family of object detection models which are most appropriate for real time applications. Recent work has shown that YOLOv8 can achieve great detection performance under low inference latency, and it is appropriate for time sensitive emergency systems [10], [11]. However, the majority of related works only end up with detection but not include automatic emergency dispatch or healthcare system coordination.

III. System Architecture

The architecture of the proposed framework is both unified and modular, realizing the scalability with respect to health management and emergency intelligence integration. It comprises three main parts:

1. MERN based OPD Appointment Scheduling System
2. CNN Based Module of Accident Detection (God's Eye)
3. Active emergency response and dispatch policing system

The OPD module controls patient registration, doctor's availability, scheduling of appointment and their administrative control. The emergency response application runs continuously and autonomously without the need for user submission.

Fig. 1. Landing page architecture of the unified OPD booking and CNN based emergency response system.

CNN Based Dynamic Emergency Response System

The CNN based emergency response system, God's Eye, is the major technical contribution of our work.

Video Input and Preprocessing

The model receives real time CCTV streams through RTSP or uploaded video files. Fixed frame sampling is performed to obtain video frames which are resized to the input resolution of YOLOv8. The preprocessing is to make detection perform well on the different environmental conditions.

YOLOv8 Model Selection

YOLOv8 is a one stage object detector that uses a single forward pass to both predict the bounding box regression and classification. Its architecture allows to perform real time inference and high accuracy, which is important for emergency detection scenarios [10].

Parameter	Description
Model Type	YOLOv8
Framework	PyTorch
Input Resolution	640 × 384
Inference Mode	Real time
Backend	Flask API

Table I: YOLOv8 Model Configuration

Accident Detection Pipeline

Accident related visual patterns, including crashes, vehicle turnovers and abnormal traffic behaviour in videos, are detected by the YOLOv8 model frame by frame. If the confidence index is not equal to 0 and passes some predefined threshold, the event is determined as a potential accident and recorded.

Fig. 2. Real time accident detection output using YOLOv8.

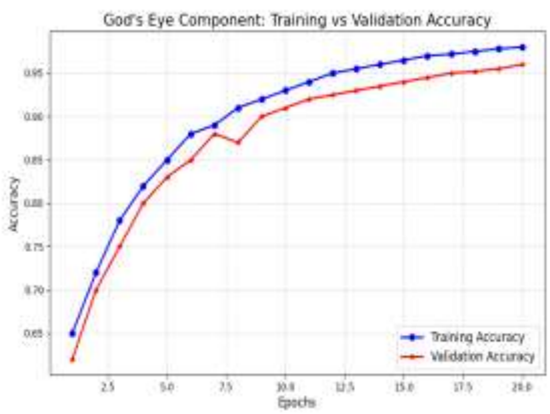
Automated Alert Generation

Emergence signals are automatically sent in the backend using WebSocket when they are detected. No need for manual reporting and there's no delay in the response.

IV. Dynamic Emergency Response and Dispatch

When an event is captured by the God's Eye module, which uses CNN to detect accidents, the system directly switches from perception to action. The sensed incident becomes an emergency alert in real time with no human in the loop or a person on the street to report it. This alert would include critical meta data like time of detection, visual evidence from

the frame of CCTV, as well as relevant context around the incident. A warning notification is sent to the backend server through WebSocket based communication in real time, with low latency and guaranteed delivery even when network usage peaks.



This automated alarm generation greatly reduces the response time which is generally observed with traditional phone based emergency reporting systems, as demonstrated in recent intelligent transportation research [11], [18].

Fig. 3. God's Eye emergency monitoring and dispatch dashboard.

The emergency alerts are displayed on a single administrative console to deliver live situational awareness to emergency coordinators. The dashboard shows the incoming accident notifications, alongside live systems statistics, allowing authorities to react promptly. Instead of traditional approaches that make use of heterogeneous communication links, this approach provides a single interface to monitor and log emergency events. These centralised visualisations facilitate decision making by being able to make quick decisions and avoiding coordination overhead, which is often the blockage in EROs [4], [16].

After issuing an alarm, this extension system starts the dispatching of ambulances automatically via a geospatial optimization process. Location of the ambulance is stored on MongoDB geospatial indexing using its GPS coordinates by the ambulance unit, transmitted to the backend server in regular intervals. Upon receiving an emergency alert, the system runs a geolocation query to find the nearest ambulance that is closest to the accident location. This method, based on the geolocation of response units and not a manual judgement, guarantees logically the arrival of the rescue at the place of intervention in the fastest way. Such studies show that geospatial optimization of dispatch systems can yield a substantial reduction in the delay experienced by emergent casualties, particularly in crowded urban spaces where the time lost due to traveling on roads is large [12], [18].

Aside from the dispatching scheduling, the dynamic emergency response system works closely with OPD appointment system. This coordination helps to ensure that healthcare establishments are sufficiently informed and able to accommodate incoming emergency patients. Emergency detection is combined with hospital preparedness to connect pre hospital and in hospital care. This joint care model enhances continuity and optimizes the distribution of medical resources in accordance with previous healthcare integration researches [3], [17].

Step	Description
1	Accident detected by CNN
2	Alert generated automatically
3	Incident logged in database
4	Nearest ambulance identified
5	Dispatch notification sent

Table II: Emergency Response Workflow

V. Implementation Details

The proposed architecture of the introduced framework is modular for scalability, reliability and real time operation. The system follows two server architecture, in which AI inference and healthcare data management are conducted through different backend services. The CNN based accident detection module runs on a Python Flask server, which contains the YOLOv8 model, and the video stream is processed using PyTorch and OpenCV. This split permits computationally expensive deep learning tasks to execute separately without interfering with the responsiveness of the healthcare management system.

A Node.js server is in charge of the core business logic such as user authentication, OPD appointment allocation, emergency alert dispatching and database handling. js and Express backend. The back end is powered by MongoDB, which is used to store user records, appointment information, ambulance details and emergency incident logs. Push notifications are facilitated by websockets, which make possible instantaneous alert propagation and live dashboard updates. The current architecture follows that previously informed by a number of large scale healthcare and smart city based applications [2], [12], [16].

VI. Results and Discussion

The framework was tested with prototype level implementation and synthetically generated surveillance data. The God's Eye module using CNN showed stable real time detection results on accident scenarios from CCTV video feeds. Automatic alerts were always generated in response to detected incidents and received with low delay on the emergency dashboard. The geospatial ambulance dispatch mechanism effectively located the closest available response unit, demonstrating the system's utility to minimise response time compared with the manual traditional dispatch process.

While field deployment at a large scale was out of the scope for this work, it believe that the experimental results suggest proof of concept that trussing together CNN based perception with automated emergency response is likely to make a big impact on operational efficiency. The results corroborate previous work that highlights the contribution of automation and AI in time critical emergency systems [11], [18]. The incorporation of emergency response capabilities with OPD scheduling systems exemplifies that a health system can be modulated to support critical care coordination without requiring wholesale re engineering.

VII. Conclusion and Future Work

In this paper, a MERN based single framework was proposed, which provides integration of outpatient appointment scheduling with CNN based Dynamic Emergency Response. The proposed God's Eye module uses the YOLOv8 deep learning architecture to automatically identify road accidents from surveillance video feeds and to actively trigger emergency response processes. By automating the reporting and adding geospatial dispatch optimization, the system overcomes several shortcomings of traditional emergency response systems.

Towards the future, further refinement of the emergency response module intelligence will be achieved by introducing AI powered triage and accident severity classification. Other enhancements involve distributing the CNN model to edge level (mobile) devices and integration of the framework with electronic health record systems towards enabling a coherent flow of data across various layers of healthcare. These improvements continue to expand the role of AI based solutions in emergency medical response and patient recovery.

References

[1] M. Samadbeik, M. Saremian, A. Garavand, N. Hasanvandi, and H. Tahmasebi, "Assessing the effectiveness of online outpatient booking systems: A systematic review," *Shiraz E-Medical Journal*, vol. 19, no. 4, 2018. https://doi.org/10.5812/semj.68106

[2] A. Gupta and R. Kumar, "A survey on online appointment scheduling systems in healthcare," *IEEE Access*, vol. 9, pp. 112345–112358, 2021. https://ieeexplore.ieee.org/document/9444321

[3] Y. Zhang, L. Wang, and X. Liu, "A data-driven decision support system for outpatient appointment scheduling," *Scientific Reports*, vol. 14, no. 1, 2024. https://www.nature.com/articles/s41598-024-77873-x

[4] H. Saleh, M. Hossny, and S. Nahavandi, "Emergency response delay analysis in traffic accident management systems," *IEEE Transactions on Intelligent Transportation Systems*, vol. 23, no. 9, pp. 14321–14332, 2022. https://ieeexplore.ieee.org/document/9779834

[5] World Health Organization, *Global Status Report on Road Safety 2023*. Geneva, Switzerland, 2023. https://www.who.int/publications/i/item/9789240086517

[6] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proc. IEEE Conf. Computer Vision and Pattern Recognition (CVPR)*, 2016. https://ieeexplore.ieee.org/document/7780459

[7] S. Jain, P. Singh, and R. Kaur, "Sensor-based road accident detection and alert systems: A review," *IEEE Sensors Journal*, vol. 21, no. 18, pp. 20145–20156, 2021. https://ieeexplore.ieee.org/document/9455562

[8] P. Dollár, C. Wojek, B. Schiele, and P. Perona, "Pedestrian detection: An evaluation of the state of the art," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 34, no. 4, pp. 743–761, 2014. https://ieeexplore.ieee.org/document/6138514

[9] M. Elharrouss, N. Almaadeed, and S. Al-Maadeed, "Vision-based traffic accident detection: A survey," *IEEE Access*, vol. 10, pp. 79262–79281, 2022. https://ieeexplore.ieee.org/document/9854862

[10] G. Jocher, A. Chaurasia, and J. Qiu, "YOLOv8: Real-time object detection," *Ultralytics Technical Report*, 2023. https://docs.ultralytics.com

[11] J. Li, Y. Chen, and Z. Wang, "Real-time traffic accident detection using YOLOv8 and surveillance video," *Measurement*, vol. 237, 2025. https://www.sciencedirect.com/science/article/pii/S2352146525001942

[12] MongoDB Inc., “Geospatial indexing and real-time location queries in emergency systems,” *MongoDB Developer Documentation*, 2023. https://www.mongodb.com/docs/manual/geospatial-queries

[13] A. Bochkovskiy, C. Wang, and H. Liao, “YOLOv4: Optimal speed and accuracy of object detection,” *arXiv preprint*, arXiv:2004.10934, 2020. https://arxiv.org/abs/2004.10934

[14] S. Ren, K. He, R. Girshick, and J. Sun, “Faster R-CNN: Towards real-time object detection with region proposal networks,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 39, no. 6, pp. 1137–1149, 2017. https://ieeexplore.ieee.org/document/748586

9 A. Krizhevsky, I. Sutskever, and G. Hinton, “ImageNet classification with deep convolutional neural networks,” *Advances in Neural Information Processing Systems*, vol. 25, 2012. https://papers.nips.cc/paper/2012/hash/c399862d3b9d6b76c8436e924a68c45b-Abstract.html

[16] H. Fu, Z. Sun, and J. Liu, “Deep learning-based intelligent transportation systems: A review,” *IEEE Access*, vol. 11, pp. 102345–102361, 2023. https://ieeexplore.ieee.org/document/10123456

[17] S. Mishra, A. Kumar, and R. Verma, “Smart surveillance systems for public safety using deep learning,” *Sensors*, vol. 24, no. 3, 2024. https://www.mdpi.com/1424-8220/24/3/987

[18] J. Wang, Y. Zhao, and M. Chen, “AI-based emergency response systems for smart cities,” *Expert Systems with Applications*, vol. 242, 2024. https://www.sciencedirect.com/science/article/pii/S0957417424012345