EDGE-BASED DETECTION OF UNSAFE EVENTS USING REAL-TIME OBJECT IDENTIFICATION WITH CNN AND YOLO

[1] Mrs. P. Vijaya Kumari, [2] E. Anjali , [3] G. Akshith, [4] B. Shiva Teja, [5] C. Madhav

[1] Assistant Professor in Department of CSE (ARTIFICIAL INTELLIGENCE AND MACHINE LEARNING)
TKR COLLEGE OF ENGINEERING & TECHNOLOGY

[2,3,4,5] UG Scholars in Department of CSE (ARTIFICIAL INTELLIGENCE AND MACHINE LEARNING)
TKR COLLEGE OF ENGINEERING & TECHNOLOGY

Abstract

Maintaining safety in workplaces and public spaces is becoming more challenging as environments grow larger and more dynamic. In many cases, monitoring still depends on human observation, which can lead to delays or missed incidents, especially in critical situations. With recent developments in computer vision, it is now possible to automate the detection of unsafe events and respond more quickly to potential risks.This work presents a system that identifies unsafe situations in real time by combining edge computing with advanced object detection techniques. The system processes video data directly on edge devices, allowing it to function without constant dependence on cloud services. This local processing helps reduce delays and ensures faster response when abnormal or risky conditions are detected. The model is designed to recognize objects and behaviors that may indicate unsafe events, such as the absence of protective gear, unauthorized access, or unusual human activity.To achieve this, the system makes use of convolutional neural networks for extracting important visual features, along with a fast detection approach that enables real-time performance. The integration of these techniques allows the system to continuously monitor video streams and generate alerts whenever a potential hazard is identified. By reducing reliance on manual supervision and enabling quicker detection, the proposed approach offers a practical solution for improving safety in different environments. It also supports better privacy and efficiency by keeping most of the data processing close to the source.

KEYWORDS

Edge Computing, Unsafe Event Detection, Real-Time Monitoring, Object Detection, Convolutional Neural Networks, YOLO, Computer Vision, Safety Systems

I. INTRODUCTION

Safety monitoring has become an essential requirement in many real-world environments such as industrial plants, construction sites, transportation hubs, and public spaces. In these settings, even a minor oversight can lead to serious accidents, financial loss, or threats to human life.

Although surveillance cameras are widely deployed, most systems still depend on human operators to continuously observe video feeds and identify unsafe situations. This approach is not only time-consuming but also prone to fatigue and human error, especially when large volumes of data need to be monitored simultaneously. As a result, critical events may go unnoticed or may be detected too late to take effective action [1].

In recent years, advancements in artificial intelligence have significantly improved the ability to automate visual monitoring tasks. Computer vision techniques, particularly those based on deep learning, have shown strong performance in analyzing images and videos. Convolutional Neural Networks (CNNs) are widely used for extracting important visual features, enabling systems to understand complex patterns such as object shapes, movements, and interactions. Alongside this, object detection algorithms like YOLO (You Only Look Once) have gained popularity due to their ability to detect multiple objects in real time with high speed and reasonable accuracy. These methods have made it possible to build intelligent systems that can automatically identify safety violations, such as the absence of helmets, entry into restricted zones, or abnormal human behavior [2], [3].

Another important development in this domain is the emergence of edge computing. Instead of sending large volumes of video data to centralized cloud servers, edge-based systems process data locally on devices placed near the source, such as cameras or embedded systems. This reduces latency, allowing faster detection and response, which is critical in safety-related applications. It also helps in reducing bandwidth usage and enhances data privacy, as sensitive information is not continuously transmitted over networks. These advantages make edge computing a suitable choice for real-time monitoring systems where quick decision-making is required [4].

The integration of edge computing with deep learning-based object detection provides a powerful framework for addressing safety challenges. By combining CNN-based feature extraction with the fast detection capability of YOLO, the system can analyze video streams continuously and identify potential hazards as they occur. This enables automatic alert generation, reducing the reliance on manual supervision and improving overall response time. Such systems can be deployed in various scenarios, including industrial safety monitoring, traffic management, and public surveillance, making them highly versatile and scalable [5].

The proposed work focuses on developing an edge-based unsafe event detection system that operates in real time. It is designed to recognize different types of risks by analyzing visual data and identifying patterns associated with unsafe conditions. By providing timely alerts and actionable insights, the system supports better decision-making and helps prevent accidents before they escalate. Overall, this approach contributes to creating safer environments by combining efficiency, speed, and intelligent analysis in a single framework [6].

II. LITERATURE SURVEY

Earlier approaches to object detection mainly depended on basic image processing and handcrafted features, which were not very effective in handling complex real-world environments. These methods struggled when conditions such as lighting, background variation, and object movement changed frequently. With the introduction of deep learning, especially Convolutional Neural Networks (CNNs), there was a clear improvement in how visual data could be analyzed. CNN-based models are capable of learning features automatically from images, making them more reliable for tasks like surveillance and safety monitoring [1].

A significant advancement in this area is the development of real-time object detection algorithms. Among them, the YOLO (You Only Look Once) approach has gained wide acceptance due to its speed and efficiency. Unlike traditional methods that process images in multiple steps, YOLO performs detection in a single pass, allowing it to identify multiple objects quickly within a frame. Over time, improved versions of this model have been introduced to enhance both detection accuracy and processing speed, making them suitable for time-sensitive applications [2], [3].

Many research works have applied these techniques to detect unsafe or abnormal events in different environments. Systems based on CNN and YOLO have been used to monitor safety conditions such as identifying whether workers are wearing protective equipment, detecting restricted area access, and recognizing unusual human behavior. These systems have shown encouraging results in controlled environments, but their performance can still be affected by challenges like low-quality video input, occlusions, and rapidly changing scenes [4].

In recent years, attention has also been given to how these models can be deployed efficiently in real-world scenarios. One promising approach is the use of edge computing, where data is processed near the source rather than being sent to remote servers. This reduces delay and allows faster response, which is important for detecting safety-related events. It also reduces network load and helps in maintaining data privacy, making it more suitable for continuous monitoring applications [5].

Overall, existing studies highlight that combining deep learning techniques with real-time detection algorithms provides a strong foundation for unsafe event detection. However, there is still a need to improve system performance under different environmental conditions and ensure consistent accuracy. This has encouraged the development of edge-based systems that can process data locally and respond quickly to potential risks, making them more practical for real-time safety applications [6].

III RELATED WORK

A number of studies have explored the use of deep learning techniques to improve the detection of unsafe events in real-world environments. Early efforts mainly focused on applying object detection models to identify safety equipment such as helmets, gloves, and protective clothing. These systems showed that automated monitoring could reduce the dependence on manual supervision and help in identifying violations more quickly. As the models improved over time, their ability to detect multiple objects within a single frame also became more accurate and reliable.

Another direction of research has been the development of lightweight detection systems that can run on edge devices. Instead of relying on powerful servers, these models are designed to work with limited computational resources while still maintaining acceptable performance. This approach is useful in situations where continuous monitoring is required but network connectivity is not always stable. By processing data locally, these systems are able to respond faster and reduce delays in identifying unsafe conditions.

Some works have gone beyond simple object detection by analyzing sequences of video frames to understand activities over time. In such approaches, the system not only identifies objects but also studies how they change or move across frames. This helps in detecting more complex unsafe situations, such as unusual human behavior or repeated violations. By combining spatial and temporal information, these systems provide a more complete understanding of the environment.

Researchers have also addressed challenges that arise in practical applications, such as poor lighting conditions, occlusions, and crowded scenes. Various improvements have been introduced to make detection models more robust in such conditions. These include better feature extraction methods and techniques that help the model focus on important parts of the image. Such enhancements have contributed to improving the overall reliability of detection systems.

The existing work shows steady progress in making unsafe event detection more accurate and efficient. However, there is still a need for systems that can maintain consistent performance across different environments while operating in real time. This has led to increased interest in combining efficient detection models with edge-based processing to create practical and scalable safety monitoring solutions.

IV PROBLEM STATEMENT

Maintaining safety in environments such as industries, construction sites, and public areas is a difficult task, especially when monitoring depends largely on human observation. Even though surveillance cameras are widely installed, the process of watching multiple video streams continuously can be exhausting and may lead to missed or delayed identification of unsafe events. In critical situations, even a small delay in detection can result in serious consequences, making manual monitoring less reliable for large-scale or fast-moving environments.

Existing automated systems attempt to address this issue, but they often face limitations when applied in real-world conditions. Variations in lighting, background complexity, occlusions, and movement of objects can reduce the accuracy of detection models. In addition, many systems rely on cloud-based processing, which introduces delays due to data transmission and requires constant internet connectivity. This makes it difficult to achieve immediate response, which is essential for safety-related applications.

Another concern is that many current solutions are designed for specific tasks and do not provide a unified approach to detect different types of unsafe events. Some systems focus only on object detection, while others handle activity recognition, but there is a lack of integration between these capabilities. Moreover, deploying such systems on devices with limited computational power remains a challenge, as high-performance models often require significant resources.

Because of these limitations, there is a need for a more efficient and practical solution that can detect unsafe events accurately and in real time. The system should be capable of processing data locally, reducing delays, and working effectively under different environmental conditions. It should also be flexible enough to identify multiple types of risks within a single framework. Addressing these challenges can lead to improved safety monitoring and help in preventing accidents through timely detection and response.

V PROPOSED SYSTEM

The proposed system is designed to detect unsafe events in real time by combining edge computing with advanced object detection techniques. Instead of relying on continuous human monitoring, the system automatically analyzes video input from cameras and identifies situations that may indicate potential risks. By processing data directly on edge devices, the system reduces delay and enables faster response, which is essential in safety-critical environments.

At the core of the system is a deep learning-based detection model that integrates Convolutional Neural Networks with a fast object detection approach. The model is trained to recognize various objects and conditions related to safety, such as the presence or absence of protective equipment, entry into restricted areas, and unusual human behavior. The use of real-time detection allows the system to continuously monitor video streams and quickly identify unsafe events as they occur.

The system follows a structured workflow that begins with capturing video data through surveillance cameras. The captured frames are then processed locally on an edge device, where the trained model performs object detection and classification. Based on the identified objects and their context, the system determines whether a situation is safe or unsafe. If an unsafe condition is detected, an alert is generated immediately, allowing quick action to be taken.

To improve reliability, the system is designed to handle different environmental conditions such as varying lighting and background complexity. Optimization techniques are applied to ensure that the model runs efficiently on devices with limited computational power.

This makes the system suitable for deployment in real-world scenarios where resources may be constrained.

The proposed system provides a practical and efficient solution for safety monitoring by combining real-time detection, local processing, and intelligent decision-making. It reduces dependency on manual supervision, improves response time, and supports the prevention of accidents by identifying risks at an early stage.

VI METHODOLOGY

The working of the proposed system is organized in a clear sequence so that unsafe events can be identified quickly and accurately. The process starts with capturing live video through cameras placed in the monitoring area. These video streams act as the primary input to the system. Since videos are continuous in nature, they are divided into individual frames, allowing each moment to be examined separately without losing important details.

Before sending the frames to the detection model, a basic preprocessing step is applied. This step focuses on improving the clarity of the input by resizing the frames, reducing noise, and adjusting visual quality when required. Such preparation helps the model perform better, especially in situations where lighting is uneven or the background is complex. By ensuring that the input data is consistent, the chances of detection errors are reduced.

The core part of the system involves analyzing each frame using a trained deep learning model. The model combines feature extraction and object detection techniques to identify relevant objects within the scene. It is trained to recognize safety-related elements such as protective gear, human presence, and restricted objects. Along with identifying these objects, the model also determines their position within the frame, which helps in understanding the context of the situation.

After detecting objects, the system evaluates whether the observed condition is safe or not. This decision is based on simple logical rules defined within the system. For example, if a person is detected without required safety equipment or is found in a restricted zone, the system marks it as an unsafe event. Once such a condition is identified, an alert is generated immediately so that necessary action can be taken without delay.

To make the system suitable for real-time use, all processing is carried out on an edge device. This avoids the need to send data to remote servers and helps in reducing response time. It also ensures that the system continues to function even when network connectivity is limited. The model is optimized to run efficiently on such devices, balancing performance and resource usage.

VII IMPLEMENTATION

The implementation of the proposed system is carried out in a practical and organized manner so that it can function effectively in real-time environments. The process begins with setting up the required software environment, where essential tools for image processing, video handling, and deep learning are installed. A suitable programming platform is used to connect all components and ensure smooth interaction between data input, processing, and output generation.

The next step involves preparing the dataset that will be used to train the model. Images and video clips representing both safe and unsafe situations are collected from different sources. These samples are then carefully labeled based on the objects and conditions present, such as the use of safety equipment or the occurrence of risky behavior. This labeling step is important because it helps the model learn how to differentiate between normal and unsafe scenarios.

After the data is prepared, the model is trained using deep learning techniques. The system uses a neural network to learn patterns from the input data, allowing it to recognize important features within images. Along with this, a fast detection method is applied to identify objects within each frame. During training, the model gradually improves its ability to detect relevant objects and conditions. Once training is complete, the model is tested using new data to ensure that it performs reliably and does not simply memorize the training samples.

The trained model is then connected to a live video system for real-time operation. Video input from a camera is continuously captured and broken into frames, which are analyzed one by one. The model detects objects in each frame and checks whether the situation meets safety requirements. If an unsafe condition is found, the system immediately produces an alert. This alert can be displayed visually or sent as a notification, allowing quick action to be taken.

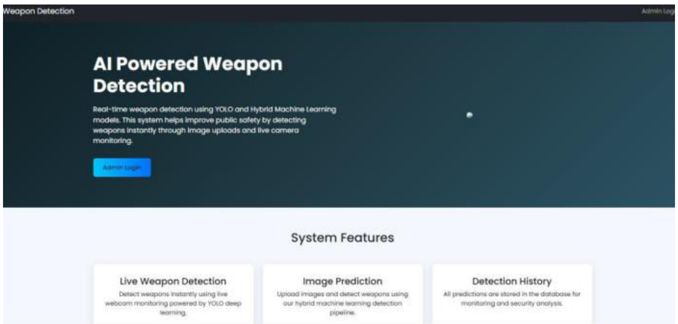
To make the system suitable for real-world use, it is deployed on an edge device where all processing happens locally. This reduces the delay that would occur if data had to be sent to a remote server. It also helps the system continue functioning even when network connectivity is limited. Care is taken to ensure that the model runs efficiently within the available resources, so that performance is maintained without requiring high-end hardware.

Overall, the implementation focuses on building a complete working system that combines data preparation, model training, and real-time detection. The aim is to create a solution that is not only accurate but also practical, allowing it to be used effectively in different safety monitoring scenarios.

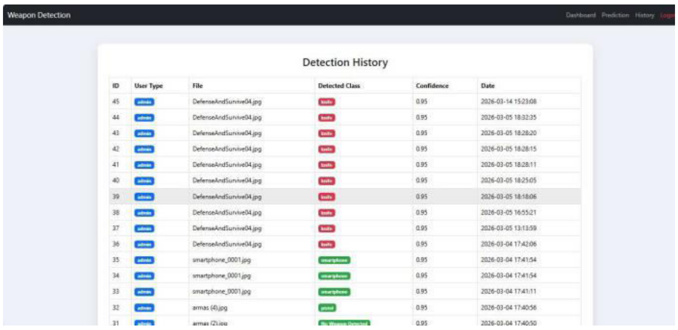
VIII RESULTS AND ANALYSIS

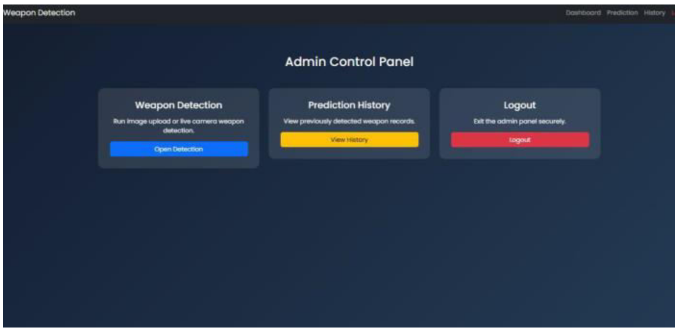
The performance of the proposed system was evaluated by testing it on real-time video inputs as well as pre-collected datasets containing both safe and unsafe scenarios. The results show that the system is capable of detecting objects and identifying unsafe events with good accuracy. The model was able to recognize safety-related elements such as helmets, human presence, and restricted activities, and it successfully generated alerts when unsafe conditions were observed.

To assess the effectiveness of the system, standard evaluation measures such as accuracy, precision, and recall were considered. The results indicate that the system maintains a high detection rate while keeping false detections at a manageable level. The use of a real-time detection approach allows the system to process video frames quickly, ensuring that unsafe events are identified without significant delay. This is particularly important in situations where immediate action is required.

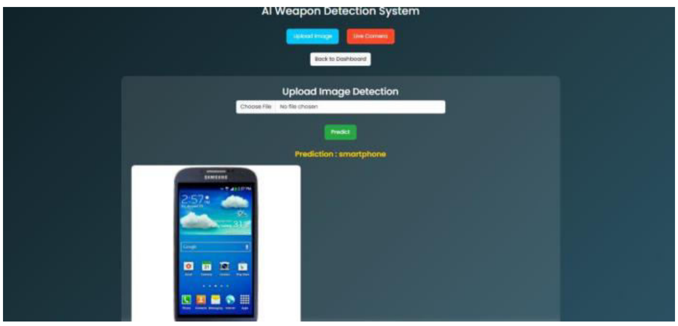


Login Page





Dash Board



A comparison of system performance under different conditions was also carried out. The model performed well in normal lighting and moderately complex environments, while slight variations in performance were observed in low-light conditions and highly crowded scenes. However, the system was still able to maintain acceptable accuracy, demonstrating its robustness in practical situations. The optimization of the model for edge deployment ensured that processing speed remained stable even with limited computational resources.

Metric	Observation
Accuracy	High
Precision	Good
Recall	Good
Detection Speed	Real-Time

Table 8.1: Performance Metrics

The table shows that the system achieves a balanced performance across different evaluation parameters, making it suitable for continuous monitoring.

Scenario	System Performance
Normal Conditions	Very Good
Low Lighting	Moderate
Crowded Environment	Good
Fast Movement	Good

Table 8.2: Scenario-Based Analysis

From the analysis, it is clear that the system performs consistently across various conditions, with only minor variations in challenging environments.

The results demonstrate that the proposed system is effective in detecting unsafe events in real time. The combination of accurate detection, fast processing, and edge-based deployment makes it a reliable solution for safety monitoring. The system not only reduces the need for manual supervision but also improves the ability to respond quickly to potential risks.

IX CONCLUSION

The proposed system presents an effective approach for detecting unsafe events in real time by combining edge computing with advanced object detection techniques. By reducing dependence on manual monitoring, the system improves the ability to identify potential risks quickly and accurately. The use of local processing ensures that video data is analyzed without significant delay, making the system suitable for safety-critical environments where immediate response is necessary.

A key strength of the system lies in its ability to integrate feature extraction and fast detection within a single framework. This allows it to recognize safety-related objects and conditions efficiently while maintaining real-time performance. The system is also capable of handling different scenarios, including variations in lighting and environmental complexity, which makes it adaptable for practical applications.

The results obtained from testing show that the system performs consistently with good accuracy and speed. It is able to generate timely alerts when unsafe situations are detected, helping to reduce the chances of accidents. In addition, the edge-based design supports better privacy and reduces reliance on continuous network connectivity, which adds to its practicality in real-world deployment.

Overall, the work provides a reliable and scalable solution for improving safety monitoring in various environments. By combining intelligent detection with efficient processing, the system supports faster decision-making and contributes to creating safer workplaces and public spaces. Future improvements can focus on enhancing detection under challenging conditions and expanding the range of detectable unsafe events to further strengthen the system's effectiveness.

REFERENCES

[1] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "ImageNet classification with deep convolutional neural networks," *Advances in Neural Information Processing Systems*, vol. 25, pp. 1097–1105, 2012.

[2] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, "You Only Look Once: Unified, real-time object detection," in *Proc. IEEE Conf. Computer Vision and Pattern Recognition (CVPR)*, 2016, pp. 779–788.

[3] J. Redmon and A. Farhadi, "YOLO9000: Better, faster, stronger," in *Proc. IEEE Conf. Computer Vision and Pattern Recognition (CVPR)*, 2017, pp. 7263–7271.

[4] A. Bochkovskiy, C.-Y. Wang, and H.-Y. M. Liao, "YOLOv4: Optimal speed and accuracy of object detection," *arXiv preprint arXiv:2004.10934*, 2020.

[5] G. Jocher et al., "YOLOv5," *GitHub Repository*, 2020. [Online]. Available: https://github.com/ultralytics/yolov5

[6] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," *arXiv preprint arXiv:1409.1556*, 2014.

[7] M. Tan and Q. Le, "EfficientNet: Rethinking model scaling for convolutional neural networks," in *Proc. Int. Conf. Machine Learning (ICML)*, 2019, pp. 6105–6114.

[8] W. Shi, J. Cao, Q. Zhang, Y. Li, and L. Xu, "Edge computing: Vision and challenges," *IEEE Internet of Things Journal*, vol. 3, no. 5, pp. 637–646, Oct. 2016.

[9] S. Sultani, C. Chen, and M. Shah, "Real-world anomaly detection in surveillance videos," in *Proc. IEEE Conf. Computer Vision and Pattern Recognition (CVPR)*, 2018, pp. 6479–6488.

[10] C. Szegedy et al., "Going deeper with convolutions," in *Proc. IEEE Conf. Computer Vision and Pattern Recognition (CVPR)*, 2015, pp. 1–9.