Leveraging Machine Learning for Real-Time Crowd Control and Safety at Kumbh Mela

Conference Paper · April 2025

DOI: 10.1109/ICCSAI64074.2025.11064144

Pallavi Shukla
United Group of Institutions

Leveraging Machine Learning for Real-Time Crowd Control and Safety at Kumbh Mela

Krishnakant Kumar
Department of CSE
UIT
Prayagraj, India
krishnakantgod454@gmail.com

Augustya Shukla
Department of CSE
UIT
Prayagraj, India
augustyashukla394@gmail.com

Pratoosh Garg
Department of CSE
UIT
Prayagraj, India
pratoosh10garg@gmail.com

Sanskar Sahu
Department of CSE
UIT
Prayagraj, India
sanskarsahu765@gmail.com

Pallavi Shukla
Department of CSE
UCER
Prayagraj, India
pallavirntu@gmail.com

***Abstract*—The Kumbh Mela, one of the largest religious gatherings worldwide, presents significant challenges in crowd management and security due to its immense scale. This project introduces a technology-driven solution leveraging machine learning, computer vision, and real-time video processing to address these challenges. The system integrates YOLO, a real-time object detection model, for crowd density estimation; ByteTrack for tracking and detecting blockades; and DeepFace for facial recognition to locate lost individuals. In addition, weapon detection is incorporated to enhance security. The framework features a responsive front end, a FastAPI powered back end, and real-time communication via WebSockets. A mobile application delivers real-time alerts to facilitate timely interventions, ensuring safer and more efficient crowd management. Preliminary tests demonstrate a 90% accuracy in crowd density estimation. This scalable solution can be adapted for other large-scale events, such as festivals and concerts, offering improved safety and operational efficiency.**

***Index Terms*—Keywords:- Kumbh Mela, Machine Learning, Crowd Management, Crowd Detection, Weapon Detection, Facial Detection, Computer Vision**

I. Introduction

The Kumbh Mela, one of the largest religious gatherings in the world, attracts millions of devotees from across India and beyond. Held every 12 years at one of four sacred locations—Prayagraj (formerly Allahabad), Haridwar, Ujjain, and Nashik—the event is a grand spectacle of faith, spirituality, and cultural heritage. The Kumbh Mela at Prayagraj, in particular, witnesses the largest influx of attendees, with the 2025 gathering projected to exceed 450 million people. While the festival symbolizes devotion and unity, the sheer scale of this gathering presents an urgent and unprecedented challenge in crowd management, safety, and security. Without effective control measures, the risk of stampedes, blockades, lost individuals, and security breaches escalates, making real-time intervention not just a necessity but a critical imperative for public safety.

Traditionally, Kumbh Mela management has relied on manual surveillance and on-ground interventions, including security personnel monitoring the crowd, setting up barricades, and using human observation to detect congestion, lost individuals, or security threats. However, these conventional methods are inherently slow, reactive, and prone to human error, making them insufficient for handling the complexities of modern-day crowd dynamics. In the past, tragic incidents such as stampedes and lost individuals taking hours or even days to be located have underscored the grave shortcomings of these approaches. In dense gatherings where every second matters, delays in identifying threats or responding to emergencies can lead to disastrous consequences.

In response to this urgent need, our solution integrates cutting-edge technologies such as machine learning, computer vision, and facial recognition to enhance crowd management and security at the Kumbh Mela. Our system offers real-time crowd detection, blockade identification, weapon detection, and an advanced lost and found system powered by facial recognition, ensuring swift identification and intervention. By leveraging live video feeds and AI-driven analytics, it provides law enforcement and event organizers with actionable intelligence, enabling them to react before situations escalate into crises.

Additionally, the implementation of an SOS alert system ensures rapid emergency response, reducing critical delays that could mean the difference between life and death in high-risk scenarios. The need for scalable, technology-driven solutions has never been more urgent, and this paper outlines the methodology, technical implementation, and life-saving potential of our system in addressing one of the most complex challenges of large-scale event management.

II. Literature Review

Effective crowd management at large-scale gatherings, such as the Kumbh Mela, is a complex challenge that has garnered significant research attention over the years. Traditional crowd management methods, including manual monitoring and on-ground interventions, have been widely employed but often fall short in addressing the dynamic and high-density nature of such events. Recent technological advancements, particularly in machine learning and computer vision, have paved the way for innovative solutions.

Baranwal et al. (2015) [2] explored the logistical and safety challenges associated with the Kumbh Mela, emphasizing the need for robust crowd control mechanisms to mitigate risks such as overcrowding, stampedes, and lost individuals. Their findings underscore the importance of integrating technology into large-scale event management to enhance safety and operational efficiency. Similarly, Yamin (2019) [13] analyzed the role of technology in crowd management at religious gatherings, comparing strategies used during the Hajj pilgrimage and Kumbh Mela, and highlighted the potential of automated systems in improving situational awareness.

The YOLO (You Only Look Once) algorithm has emerged as a state-of-the-art model for real-time object detection. Jiang et al. (2022) [5] provided a comprehensive review of YOLO's development, illustrating its efficiency in identifying and tracking objects in crowded environments. By enabling the detection of individuals and potential security threats in milliseconds, YOLO offers a significant advantage over traditional methods. ByteTrack, a multi-object tracking algorithm, has further enhanced the ability to monitor crowd dynamics. Zhang et al. (2022) [14] demonstrated ByteTrack's superior performance in maintaining identity consistency across video frames, which is critical for detecting stationary groups or blockades in large gatherings.

Habib et al. [4] developed a novel framework to identify abnormal activity for pilgrims at Makkah. A lightweight CNN model was trained on the dataset of pilgrims. The system will make an alarm when an emergency occurs, such as an accident or violent activity, to inform the authorities to take the appropriate action. They have performed experiments on two violent activity datasets: Hockey Fight and Surveillance Fight. The model achieved good accuracies of 98.00 and 81.05, respectively. However, this model sufers from the shortcoming of recognizing violent activity from one perspective only. And it is the best that recognizes violent activity from multiple perspectives to obtain insights into the activities.

Facial recognition technologies have shown remarkable progress in real-time identification of lost persons. DeepFace, introduced by Taigman et al. (2014) [11], bridges the gap between human-level accuracy and machine learning performance in face verification. This model has been successfully applied in dynamic and dense environments, making it an ideal choice for identifying lost individuals at events like the Kumbh Mela. In high-density events, weapon detection is a critical aspect of maintaining public safety. Narejo et al. (2021) [9] explored the application of YOLOv3 in smart surveillance systems for weapon detection, achieving significant accuracy and speed in identifying potential threats in real-time. Such advancements emphasize the role of automated systems in enhancing security measures at large-scale gatherings.

The integration of machine learning models with real-time communication protocols has proven to be a transformative approach to crowd management. Wu (2024) [12] highlighted FastAPI's capabilities as a backend framework for handling real-time video feeds and model predictions, enabling swift and accurate decision-making during emergencies. Additionally, Firebase Cloud Messaging has been leveraged to ensure timely alerts to stakeholders, further improving response efficiency.

III. Proposed Framework

In large-scale events such as the Kumbh Mela, effective crowd management is a critical challenge due to the sheer number of attendees and the potential for emergencies, such as overcrowding, lost persons, and security threats. Traditional methods that rely on human surveillance are often insufficient, as they are slow and prone to errors, especially in dynamic, densely populated environments. The problem faced by the research team was to design a system capable of monitoring the crowd in real time, detecting critical situations, and ensuring timely alerts to prevent potential hazards. The solution proposed by the team is an integrated system that combines machine learning, real-time video processing, and modern communication technologies to automate crowd monitoring, detect security risks, and assist in locating lost individuals.

The foundation of the system is built on YOLO (You Only Look Once) [5], a state-of-the-art object detection algorithm used for real-time crowd density estimation. YOLO is a deep learning model [6] designed to recognize multiple objects within an image or video frame in a single pass, making it highly efficient for detecting people in crowded environments. The team employed this model to monitor crowd density and trigger alerts when the number of people in a given area exceeds a predefined threshold, signaling potential overcrowding. This approach addresses the problem of manual monitoring, which can be slow and unreliable, by providing automated, real-time insights into crowd movements.

Another challenge the team tackled was the detection of blockades, which occur when people's movement in a specific area is obstructed or significantly slowed. To solve this, the team incorporated ByteTrack [14], an advanced tracking algorithm used for tracking multiple objects across video frames. ByteTrack allows the system to track individuals over time, observing their movement patterns. If a large group of people remains stationary in the same area for too long, the system flags this as a potential blockade, alerting authorities to take action. This solution ensures that the system can detect emergencies related to overcrowding or halted movement, which may not be immediately obvious to human observers.

Our team also recognized the critical need to assist with locating lost persons, particularly children, in large crowds.

For this, the solution developed a lost child recognition system using DeepFace [10], a facial recognition library based on the VGGFace model [3]. DeepFace analyzes facial features in a given image and compares them against a database of reported missing persons. When a child is reported lost, the system stores their image, which is then matched against live footage from cameras at the event. If a match is found, the system alerts authorities and parents. This solution replaces the need for manual searches and helps identify lost individuals quickly in real-time video feeds.

To enhance event security, the system also incorporates weapon detection using the same YOLO model [9]. The model is trained to detect dangerous objects, such as knives or firearms, in real-time video streams. This allows for immediate alerts in the event of a security threat, improving the overall safety of attendees.

The project's frontend, the part of the system that users interact with, was built using React.js [1], a popular JavaScript library for creating dynamic user interfaces. Tailwind CSS , a utility-first CSS framework, was used to ensure that the interface is both responsive and visually appealing. Through this interface, event organizers and law enforcement officials can monitor the crowd in real time, view reports, and receive alerts. The user interface communicates with the backend, which handles the heavy processing of video data and model predictions.

The backend of the system was developed using FastAPI [12], a modern web framework for building APIs (Application Programming Interfaces). FastAPI is known for its speed and efficiency, making it ideal for handling real-time video feeds and machine learning model predictions. The system relies on WebSockets, a communication protocol that allows for continuous, real-time data exchange between the frontend and backend. This ensures that video frames are transmitted from the frontend to the backend every 150 milliseconds, where they are processed, and detection results are returned instantly.

For storing and managing images used in lost person recognition, the team integrated Cloudinary, a cloud-based media management service. Cloudinary allows for the efficient storage and retrieval of images, which are then used for facial recognition. To ensure rapid communication in case of an alert, the system uses Firebase Cloud Messaging, a platform that sends notifications to relevant stakeholders, such as event organizers and parents, as soon as a match or detection is made.

To provide a mobile-friendly solution, the team developed a native Android application using Kotlin [8], a modern programming language for Android development, and Jetpack Compose [7], a toolkit for building native UI. The mobile app allows users to receive real-time alerts and updates, making it a critical tool for on-the-go event monitoring.

In summary, the proposed framework addresses the major challenges of crowd management by automating the detection of overcrowding, blockades, security threats, and lost persons. By integrating cutting-edge technologies like YOLO, ByteTrack, and DeepFace with fast, reliable backend communication through FastAPI and WebSockets, the system offers a comprehensive solution for managing crowds at large-scale events. The system's real-time monitoring and alerting capabilities ensure that potential hazards are detected and addressed promptly, enhancing both the safety and efficiency of crowd management efforts. Figure 1 provides a better look at understanding of working of Samraksh(The system developed for proper crowd management at crowded areas).

IV. Technical Implementation

The *SAMRAKSH* system integrates powerful machine learning models and real-time processing techniques to address challenges such as crowd management, blockade detection, lost child identification, weapon detection, and emergency alerting during the Kumbh Mela. The system employs YOLOv11n, a lightweight and efficient deep learning model for real-time object detection, enabling accurate crowd density estimation and weapon detection.Figure 2 shows crowd detection using YOLOv11 with all the necessary crowd information and Figure 5 shows how a weapon is detected using YOLOv11.YOLOv11 utilizes the COCO (Common Objects in Context) dataset, which encompasses 80 diverse classes and provides a comprehensive benchmark for these tasks. For image classification tasks, YOLOv11 models are trained on the ImageNet dataset, renowned for its extensive collection of labeled images across numerous categories. These foundational datasets equip YOLOv11 with the versatility to perform effectively in multiple computer vision applications.The COCO (Common Objects in Context) dataset is an extensive collection designed for object detection, segmentation, and image captioning tasks. It includes over 330,000 images, with more than 200,000 specifically annotated for training and validation. Spanning 80 different object categories, this dataset provides a diverse and well-structured foundation for model development. The total dataset size is approximately 37.35 GiB and ImageNet spans 1,000 object classes,it contains 1,281,167 training images, 50,000 validation images, and 100,000 test images. The entire dataset amounts to roughly 150 GB in size..

ByteTrack, a state-of-the-art multi-object tracking algorithm, is used to identify and monitor blockades or suspicious movements in dense crowds.Figure 3 shows implementation of Bytetrack and YOLO for blockade detection .ByteTrack is evaluated using several widely recognized datasets to assess its performance in multi-object tracking. The MOT17 dataset includes 7 video sequences with complex tracking scenarios, totaling around 1.8 GB in size. MOT20, a more challenging dataset with 8 sequences featuring higher crowd densities, is approximately 3.2 GB. The HiEve dataset focuses on human-centric events, offering 50 sequences with a total size of about 10 GB. Finally, BDD100K, a large-scale driving video dataset containing 100,000 diverse driving scenarios, is roughly 160 GB in size. These datasets provide a comprehensive benchmark for evaluating ByteTrack's effectiveness in various tracking environments.

For lost child identification, DeepFace(VGGFace), a robust facial recognition model, is implemented to match faces in real-time against a pre-registered database of missing individuals.Figure 4 shows how DeepFace(VGGFace) is used for face recognition which can help in missing individuals in a crowded area.DeepFace uses several key datasets for facial recognition tasks, with the VGGFace2 dataset being one of the primary sources for training its models. VGGFace2 contains over 3 million images, representing 9,131 identities, offering a diverse range of poses, illumination conditions, and age groups. This large-scale dataset is approximately 200 GB in size, making it suitable for training robust facial recognition models that can generalize well to various real-world scenarios.

The integration of these models allows SAMRAKSH to process live video feeds and deliver timely alerts through a user-friendly interface, enhancing safety during large-scale events.

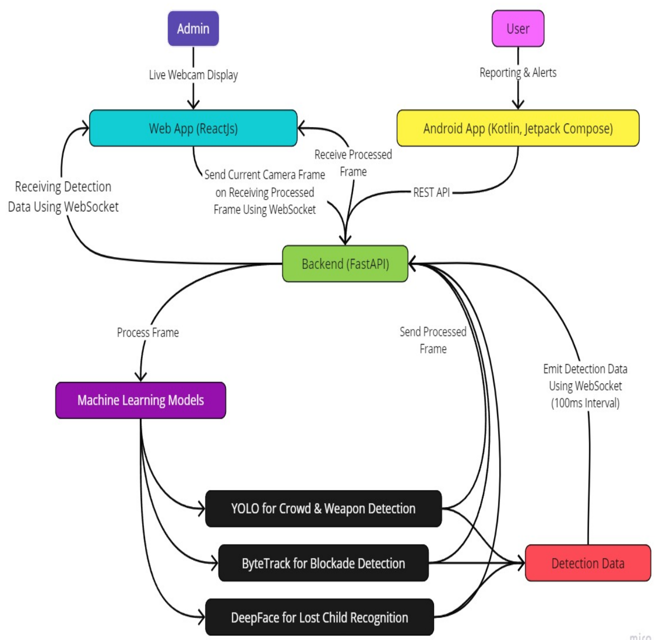

Fig. 1. Working of Samraksh

Algorithm 1 Crowd Detection Using YOLO

Input: Video frame, YOLO model, threshold for overcrowding
Output: Alert for overcrowded area
Step 1: Load the live video feed and pass the current frame into the YOLO model.
Step 2: Detect objects in the frame.
Step 3: Extract all detected objects classified as ”Person”.
Step 4: Count the number of detected ”Person” objects.
Step 5: Compare the count against the predefined crowd density threshold.
Step 6: If the count exceeds the threshold then trigger an alert for overcrowding

Fig. 2. Crowd Detection Using YOLO

Algorithm 2 Blockade Detection Using YOLO and ByteTrack

Input: Video frame, YOLO model, ByteTrack tracker, X(people), Y(time)
Output: Blockade detection alert
Step 1: Detect individuals using YOLO and assign a unique ID for each detected person using ByteTrack tracking.
Step 2: For each detected person record the time of first detection (first_detected_at) and continuously update last_detected_at for each frame where the person remains in view.
Step 3: For each individual or group of X people, calculate the time difference between first_detected_at and last_detected_at.
Step 4: If the difference exceeds a predefined time threshold (Y seconds) then classify the area as a blockade and trigger an alert.

Fig. 3. Blockade Detection Using YOLO and ByteTrack

Algorithm 3 Lost Child Detection Using DeepFace

Input: User-submitted child image, lost child dataset, live video frames
Output: Alert when a lost child is identified
Step 1: User submits a form with the child’s details, including a photo.
Step 2: Detect face using Deepface(VGG Model).
Step 3: Store the validated image and child details in the lost child report dataset.
Step 4: Continuously process live webcam footage by detecting face in each frame and compare each detected face with the lost child images in the dataset using DeepFace.
Step 5: If a match is found then send alerts to stakeholders using Firebase Cloud Messaging with the child’s details.

Fig. 4. Face Detection Using DeepFace

Algorithm 4 **Weapon Detection Using YOLO**

Input: Video frame, YOLO model
Output: Alert for detected weapon
Step 1: Pass the live video frame into the YOLO model.
Step 2: Detect objects in the frame.
Step 3: Identify and classify objects matching specific weapon labels, such as ”Gun” or ”Knife.”
Step 4: If a weapon is detected then classify the type of weapon detected and then immediately trigger an alert to the authorities.

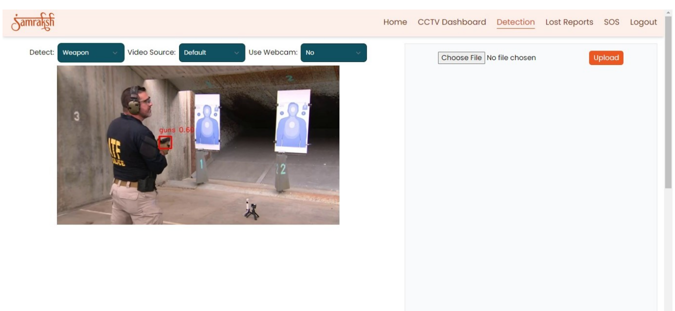

Fig. 5. Weapon Detection

V. Results

A. Crowd Detection Results

Traditionally, crowd management at Kumbh Mela relied on manual surveillance, involving police patrols and volunteers spread throughout the area. Over the years, CCTV cameras were introduced to assist in real-time monitoring, but these systems required constant human attention and intervention. In contrast, our system, utilizing YOLO for real-time crowd detection, processes video feeds autonomously. With our implementation of YOLOv11, we achieved a mean Average Precision (mAP) of 54.7% as shown in Figure 6, ensuring reliable real-time detection of crowd densities. YOLOv11 processes video streams and detects crowd congestion in under 2.5 seconds, significantly enhancing response time compared to manual methods.
Model for the ’Person’ class in YOLO 11n(Coco Dataset) achieves 75.5% precision, meaning most detected individuals are correctly classified, and 66.5% recall, indicating some individuals remain undetected. The F1-score of 0.707 balances these metrics, showing strong precision but room for improvement in recall. Enhancing feature extraction and multi-angle analysis could further optimize detection for large-scale events like Kumbh Mela.

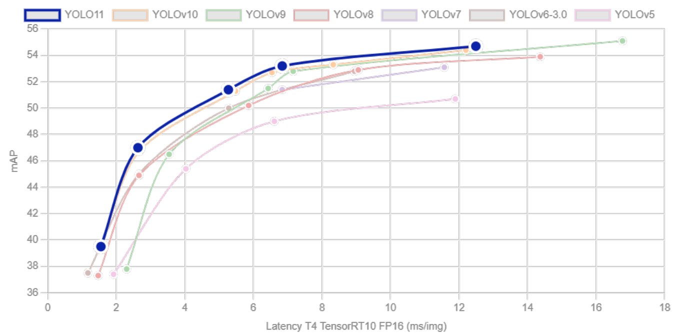

Fig. 6. YOLO Benchmark Visualization

B. Blockade Detection Result

Previously, blockade detection was done manually, where authorities would react only after noticeable delays or congestion had already escalated. The introduction of object tracking models like DeepSort helped monitor crowd movement but was limited by inaccuracies and delayed responses. Our system uses ByteTrack, which offers more reliable and accurate tracking. ByteTrack achieves MOTA(Multi-object tracking accuracy) of 76% and IDF of 79% in detecting groups of stationary individuals and identifies blockades.MOTA measures tracking errors, including misses, false positives, and mismatches, based on their ratios across all frames. ByteTrack outperforms SORT with higher MOTA (76% vs 74%) and IDF1 (79% vs 77%), ensuring better identity consistency and tracking accuracy, as shown in Figure 7 [14]. Previous crowd management techniques used simpler algorithms that struggled with real-time updates in dense environments.

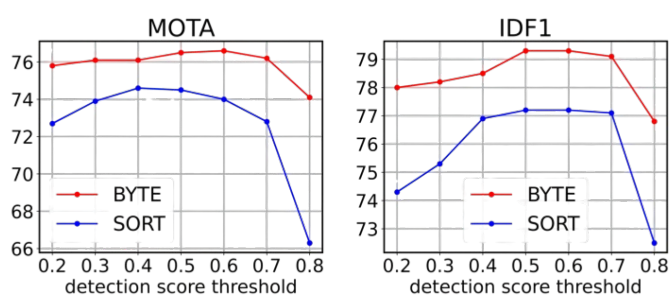

Fig. 7. Comparison of the performances of BYTE and SORT under different detection score thresholds.

C. Lost Child Detection Performance

Historically, lost individuals at Kumbh Mela were found using the Bhule Bhatke Shivir, a camp that reunited lost individuals by broadcasting announcements over loudspeakers. While effective for many years, this system often took hours or even days to locate and reunite family members, especially given the massive crowds at the event.

Our system uses DeepFace to automatically detect and recognize lost children based on user-submitted images, significantly improving response times identifying lost children, our system can send alerts within seconds of identifying a match, making it much faster and more efficient than traditional methods.

The Figure 8 [11] provided illustrates that DeepFace-single outperforms other methods in Receiver Operating Characteristic (ROC) curves on the YouTube Faces (YTF) dataset, achieving a true positive rate (TPR) of 91.4%. This is significantly higher than alternatives like VSOF+OSS (79.7%), STFRD+PMML (79.5%), APEM+FUSION (79.1%), and MBGS variants (78.9%). ROC curve is a graph that shows how well a model performs at different threshold values. These results emphasize the superiority of DeepFace in recognizing lost children in crowded environments, validating its selection for our project. The higher accuracy of DeepFace not only ensures better reliability but also reduces false alarms, which is critical in real-time applications like crowd management at events such as Kumbh Mela.

By integrating DeepFace, our model has effectively enhanced child detection accuracy, providing faster and more precise alerts. In comparison, previous systems relied on less robust methods, which may have led to higher error rates in identifying individuals in a dense crowd.

This performance boost shows the critical improvement made by our approach, helping authorities respond more efficiently to incidents of lost children.

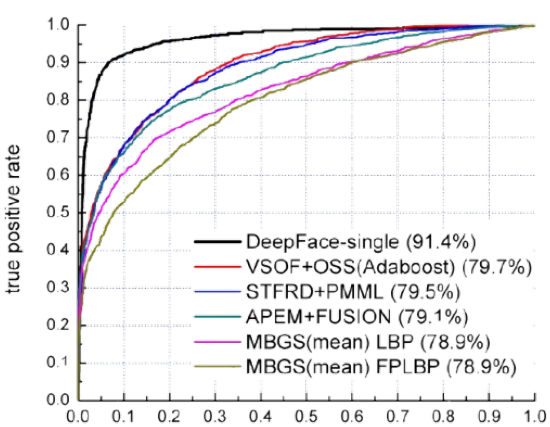

Fig. 8. The ROC curves on the YTF dataset

D. Weapon Detection Results

In past Kumbh Mela(s), weapon detection was handled manually by security personnel. Visual inspections were prone to error, especially in dense crowds, and weapon detection was slow, with a high potential for missed threats. With YOLO, our system can detect potential weapons such as knives or firearms within seconds.This enables real-time monitoring of threats, allowing security personnel to intervene promptly, reducing risks during large gatherings. In our Weapon Detection model, we leverage the same YOLOv11 architecture previously discussed for crowd detection. With this implementation, we achieved an mAP of 54.7%, which reflects the precision of our system in detecting objects such as weapons within the crowd. The real-time object detection capabilities of YOLOv11 make it a suitable model for this task, ensuring quick and accurate identification of threats in crowded environments. Its high efficiency in real-time analysis aligns with the need for robust security systems in large gatherings like Kumbh Mela.

VI. Conclusion

SAMRAKSH represents a transformative leap in large-scale event safety and crowd management by integrating cutting-edge AI technologies such as YOLOv11n, ByteTrack, and VGGFace. These advancements enable real-time crowd detection, blockade identification, lost child recognition, and weapon detection, all of which are critical in mitigating potential security threats and ensuring rapid response. By automating these essential processes, SAMRAKSH eliminates delays inherent in traditional crowd control methods, significantly enhancing public safety. Its ability to analyze vast amounts of video footage in real-time makes it a reliable and efficient solution for managing large gatherings, where quick decision-making can be the difference between order and chaos.

Despite its innovative capabilities, deploying SAMRAKSH in events of such magnitude, such as the Kumbh Mela, poses several challenges. Scalability remains a major concern, as the system must process thousands of live video feeds simultaneously while maintaining high accuracy. This demands massive computational power and high-bandwidth network availability, both of which are often constrained in temporary event setups. Edge computing offers a potential solution by reducing latency and reliance on cloud infrastructure, but it introduces additional costs and implementation complexities. Ensuring seamless real-time processing in a highly dynamic environment requires a hybrid approach that balances computational efficiency with practical feasibility.

Environmental and technical constraints further complicate the deployment of AI-driven event management systems. Unpredictable weather conditions, such as heavy rain, dense fog, or dust storms, can impact camera visibility, reducing the accuracy of object detection and crowd analytics. Additionally, network reliability is crucial for continuous monitoring and response coordination, yet large-scale outdoor events often suffer from connectivity issues due to signal congestion and infrastructure limitations. To maintain effectiveness across diverse scenarios, SAMRAKSH must incorporate sensor fusion techniques, integrating multiple data sources, such as thermal imaging and LiDAR, to enhance detection capabilities under adverse conditions. Furthermore, model adaptation is essential, as different crowd behaviors and environmental variations necessitate frequent retraining of AI models to ensure optimal performance.

Ethical considerations also play a crucial role in the responsible deployment of SAMRAKSH. As an AI-driven surveillance system, it must navigate concerns related to data privacy, transparency, and regulatory compliance. The collection and processing of personal data, such as facial recognition for lost child identification, must adhere to strict policies to prevent misuse. Implementing measures like limited data retention, encryption, and anonymization can help mitigate privacy risks. Additionally, clear guidelines on AI decision-making and human oversight will be necessary to maintain public trust and ensure accountability. Establishing regulatory frameworks that align with global ethical standards will further strengthen the legitimacy and acceptance of such technology-driven safety solutions.

Despite these challenges, SAMRAKSH lays the foundation for a scalable, intelligent, and proactive approach to large-scale event management. Its modular architecture allows for customization across various high-density public gatherings worldwide, including concerts, sporting events, and religious congregations. By integrating AI-driven predictive analytics, the system has the potential to move beyond reactive measures and enable proactive decision-making, anticipating risks before they escalate into critical situations. Continuous research and on-ground testing will be essential to refine its capabilities, improve efficiency, and explore advanced AI methodologies for even more precise event monitoring.

As technology continues to evolve, SAMRAKSH exemplifies how artificial intelligence can be leveraged to enhance public safety and security in large-scale gatherings. By bridging the gap between innovation and real-world implementation, it sets a precedent for future AI-powered event management systems. With ongoing advancements in AI, edge computing, and ethical governance, SAMRAKSH has the potential to redefine how large-scale events are managed, ensuring safer, more organized, and more efficient crowd control strategies for millions of attendees worldwide.

References

[1] Sanchit Aggarwal et al. Modern web-development using reactjs. *International Journal of Recent Research Aspects*, 5(1):133–137, 2018.

[2] Annu Baranwal, Ankit Anand, Ravikant Singh, Mridul Deka, Abhishek Paul, Sunny Borgohain, and Nobhojit Roy. Managing the earth's biggest mass gathering event and wash conditions: Maha kumbh mela (india). *PLoS currents*, 7, 2015.

[3] Hongling Chen and Chen Haoyu. Face recognition algorithm based on vgg network model and svm. In *Journal of Physics: Conference Series*, volume 1229, page 012015. IOP Publishing, 2019.

[4] Shabana Habib, Altaf Hussain, Waleed Albattah, Muhammad Islam, Sheroz Khan, Rehan Ullah Khan, and Khalil Khan. Abnormal activity recognition from surveillance videos using convolutional neural network. *Sensors*, 21(24):8291, 2021.

[5] Peiyuan Jiang, Daji Ergu, Fangyao Liu, Ying Cai, and Bo Ma. A review of yolo algorithm developments. *Procedia computer science*, 199:1066–1073, 2022.

[6] Yann LeCun, Yoshua Bengio, and Geoffrey Hinton. Deep learning. *nature*, 521(7553):436–444, 2015.

[7] S Marchenko. Jetpack compose: New approaches to android ui development. *Publishing House "Baltija Publishing"*, 2023.

[8] Marcin Moskala and Igor Wojda. *Android Development with Kotlin*. Packt Publishing Ltd, 2017.

[9] Sanam Narejo, Bishwajeet Pandey, Doris Esenarro Vargas, Ciro Rodriguez, and M Rizwan Anjum. Weapon detection using yolo v3 for smart surveillance system. *Mathematical Problems in Engineering*, 2021(1):9975700, 2021.

[10] Omkar Parkhi, Andrea Vedaldi, and Andrew Zisserman. Deep face recognition. In *BMVC 2015-Proceedings of the British Machine Vision Conference 2015*. British Machine Vision Association, 2015.

[11] Yaniv Taigman, Ming Yang, Marc'Aurelio Ranzato, and Lior Wolf. Deepface: Closing the gap to human-level performance in face verification. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1701–1708, 2014.

[12] Xiaomin Wu. Fastapi as a backend framework. 2024.

[13] Mohammad Yamin. Managing crowds with technology: cases of hajj and kumbh mela. *International journal of information technology*, 11(2):229–237, 2019.

[14] Yifu Zhang, Peize Sun, Yi Jiang, Dongdong Yu, Fucheng Weng, Zehuan Yuan, Ping Luo, Wenyu Liu, and Xinggang Wang. Bytetrack: Multi-object tracking by associating every detection box. In *European conference on computer vision*, pages 1–21. Springer, 2022.

