Safevision:Automated PPE Detection and Violation Monitoring using Yolov8 and Deepsort

Krishnendu Patni, Sachin Kumar, Swati Dwivedi, Nitin Kumar

krishnendupatni5@gmail.com

*Department of BCA (*Batch: 2023–2026*), Haridwar University, Roorkee, Haridwar*

Internal Guide: **Mr. Gaurav Kumar**, Asst. Professor -nayakgaurav682@gmail.com

1. Abstract

Industrial workplaces such as construction sites, chemical plants, steel industries, and warehouses face significant safety challenges due to non-compliance with Personal Protective Equipment (PPE) regulations. Manual supervision methods are inefficient and prone to human error. This paper presents **SafeVision**, a real-time Artificial Intelligence (AI)-based PPE detection and safety monitoring system designed to automate compliance verification using computer vision and deep learning. The system integrates YOLO-based object detection, DeepSORT-based multi-object tracking, and a web-based dashboard built with modern web technologies. SafeVision detects safety violations, logs events in a centralized database, triggers real-time alerts, and provides actionable analytics for safety administrators. Experimental evaluation demonstrates high detection accuracy with real-time processing capabilities, making the system suitable for industrial deployment. The proposed solution contributes toward Industry 4.0 by enhancing automated workplace safety monitoring.

Keywords

Artificial Intelligence, Computer Vision, PPE Detection, YOLO, DeepSORT, Workplace Safety, Real-Time Monitoring, Industrial Automation

2. Introduction

Workplace safety remains a critical concern across industrial sectors. Despite strict safety regulations, non-compliance with PPE requirements continues to cause injuries and fatalities. Supervisory monitoring is traditionally manual, making it inefficient and inconsistent in large-scale industrial environments.

With rapid advancements in Artificial Intelligence (AI) and Deep Learning (DL), automated visual monitoring systems have emerged as practical solutions. Object detection models such as YOLO (You Only Look Once) enable real-time detection with high precision. When combined with multi-object tracking algorithms like DeepSORT, persistent identification of individuals across frames becomes feasible.

SafeVision is proposed as a comprehensive AI-based PPE monitoring system that:

- Detects safety helmets, reflective vests, gloves, and configurable PPE.
- Tracks workers across video frames.
- Logs violations in real-time.
- Generates alerts via dashboard and optional hardware modules.
- Provides safety analytics through an interactive web interface.

The primary objective is to reduce workplace accidents, improve compliance rates, and assist industrial management in proactive safety enforcement.

3. Literature Review

Recent research in workplace automation emphasizes AI-driven safety monitoring systems. Convolutional Neural Networks (CNNs) have significantly improved object detection accuracy. YOLO models are widely used for real-time detection tasks due to their single-stage detection architecture, which reduces inference latency.

Faster R-CNN and SSD models provide high accuracy but may suffer from slower processing speeds compared to YOLO. Multi-object tracking methods such as DeepSORT enhance object detection systems by maintaining identity consistency across frames using Kalman filtering and appearance embeddings.

Several existing PPE detection systems focus only on helmet detection. However, most lack integrated dashboards, multi-PPE detection capability, scalable backend architecture, or hardware alert mechanisms. SafeVision bridges this gap by combining detection, tracking, database logging, web analytics, and optional IoT alerts into a unified architecture.

4. System Architecture

SafeVision follows a modular architecture designed for scalability and real-time industrial deployment.

4.1. Overall Architecture

1. Input Layer
- The system receives live video streams from CCTV or IP cameras installed in the workplace.
- Video is divided into individual frames using OpenCV for processing.
- These frames serve as the input to the AI detection module.

2. Frame Preprocessing
- Each video frame is preprocessed to improve detection performance.
- Operations include:
  - Resizing images
  - Noise reduction
  - Image normalization
- This step ensures faster and more accurate object detection.

3. YOLO Object Detection
- The preprocessed frames are passed to the YOLO deep learning model.
- YOLO detects objects in real time such as:
  - Workers (person)
  - Helmets
  - Safety vests
  - Masks
- The model outputs bounding boxes, object labels, and confidence scores.

4. DeepSORT Tracking
- DeepSORT tracking algorithm tracks detected workers across consecutive frames.
- Each worker is assigned a unique ID.
- This allows the system to:
  - Track worker movement
  - Avoid duplicate violation counts
  - Maintain identity consistency.

5. PPE Compliance Check
- The system checks whether each worker is wearing the required personal protective equipment.
- Detected objects are compared with predefined safety rules.
- If any required PPE item is missing, the system marks it as a safety violation.

ISSN: 2583-6129
DOI:10.55041/ISJEM05783

6. Flask API Communication

- Violation data is sent through a Flask API.
- The API transfers information such as:
  - Worker ID
  - Type of violation
  - Timestamp
  - Camera location.

7. Backend Processing (Node.js + Express)

- The Node.js backend server receives data from the Flask API.
- It handles:
  - API requests
  - Data processing
  - Authentication
  - Communication with the database.

8. MongoDB Database

- Violation records are stored in a MongoDB database.
- The database maintains:
  - Safety logs
  - Worker compliance data
  - Historical monitoring information.

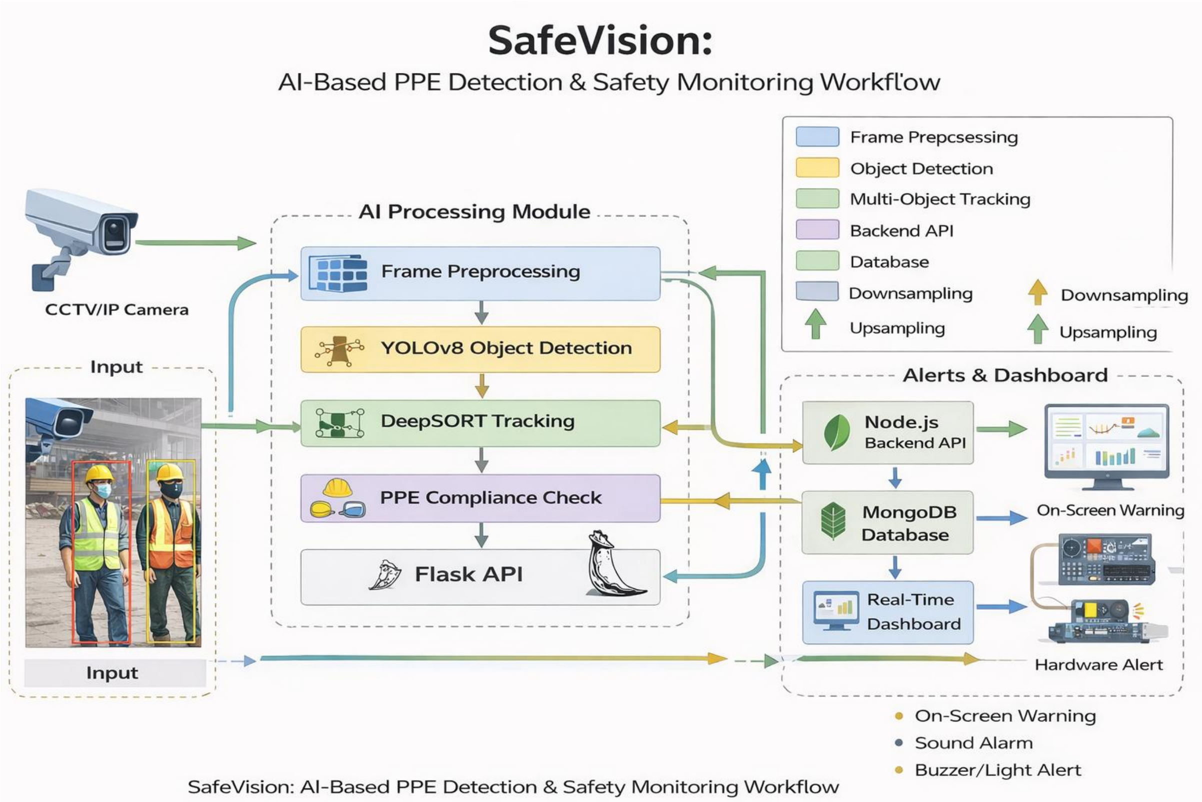
9. Real-Time Dashboard

- A React.js dashboard displays monitoring results.
- It shows:
  - Live camera feed
  - Safety violation alerts
  - Compliance statistics
  - Graphs and reports using Recharts.

10. Alert System

- When a violation is detected, the system generates alerts such as:
  - On-screen warning
  - Sound alarm
  - Arduino-based hardware alert (buzzer or light).

ISSN: 2583-6129
DOI:10.55041/ISJEM05783



4.2. Detection Module

The detection module is implemented using:

- Python 3.x
- OpenCV for video frame extraction
- YOLO for object detection
- DeepSORT for multi-object tracking
- Flask API for backend communication

4.3. Detection Workflow

1. Live video feed is captured using OpenCV.
2. YOLO detects persons and PPE items in each frame.
3. DeepSORT assigns unique IDs to each detected person.
4. PPE compliance logic verifies whether required equipment is worn.
5. Violations are sent to the backend API for logging.

The system achieves real-time inference suitable for live industrial monitoring.

4.4. Backend System

The backend is built using:

- Node.js
- Express.js
- MongoDB

- JWT Authentication

The backend responsibilities include:

- Receiving violation data from detection module
- Storing events in MongoDB
- Managing authentication and authorization
- Providing RESTful APIs for frontend

Role-based access control allows different privileges for Admin and Supervisor users.

4.5. Frontend Dashboard

The frontend is developed using:

- React.js
- Tailwind CSS
- Recharts

Dashboard features include:

- Real-time violation monitoring
- Zone-based safety analytics
- Graphical visualization of compliance trends
- Exportable CSV and PDF reports
- Role-based authentication

The dashboard dynamically updates through API polling or WebSocket integration.

4.6. Hardware Alert System (Optional)

SafeVision supports Arduino-based LED and buzzer alerts. When violations are detected:

- A buzzer generates an audible alert.
- LED indicators flash in violation zones.

This provides immediate physical feedback in industrial settings.

5. Methodology

5.1. Dataset Collection

The dataset consists of labeled images containing:

- Workers wearing helmets
- Workers without helmets
- Workers wearing reflective vests
- Workers without vests
- Diverse lighting and environmental conditions

Data augmentation techniques such as rotation, scaling, and brightness adjustment are applied to enhance generalization.

5.2. Model Training

The YOLO model is trained using annotated bounding box datasets. Training involves minimizing:

- Localization loss
- Confidence loss
- Classification loss

Hyperparameters such as batch size, learning rate, and epochs are optimized for best performance.

5.3. PPE Compliance Logic

For each detected individual:

- Identify overlapping PPE bounding boxes.
- Verify mandatory PPE presence.
- Log violation if missing.
- Use tracking ID to prevent duplicate violation logging.

This ensures accurate compliance monitoring.

6. Experimental Results

6.1. Evaluation Metrics

The system performance is measured using:

- Precision
- Recall
- F1-Score
- Mean Average Precision (mAP)
- Frames Per Second (FPS)

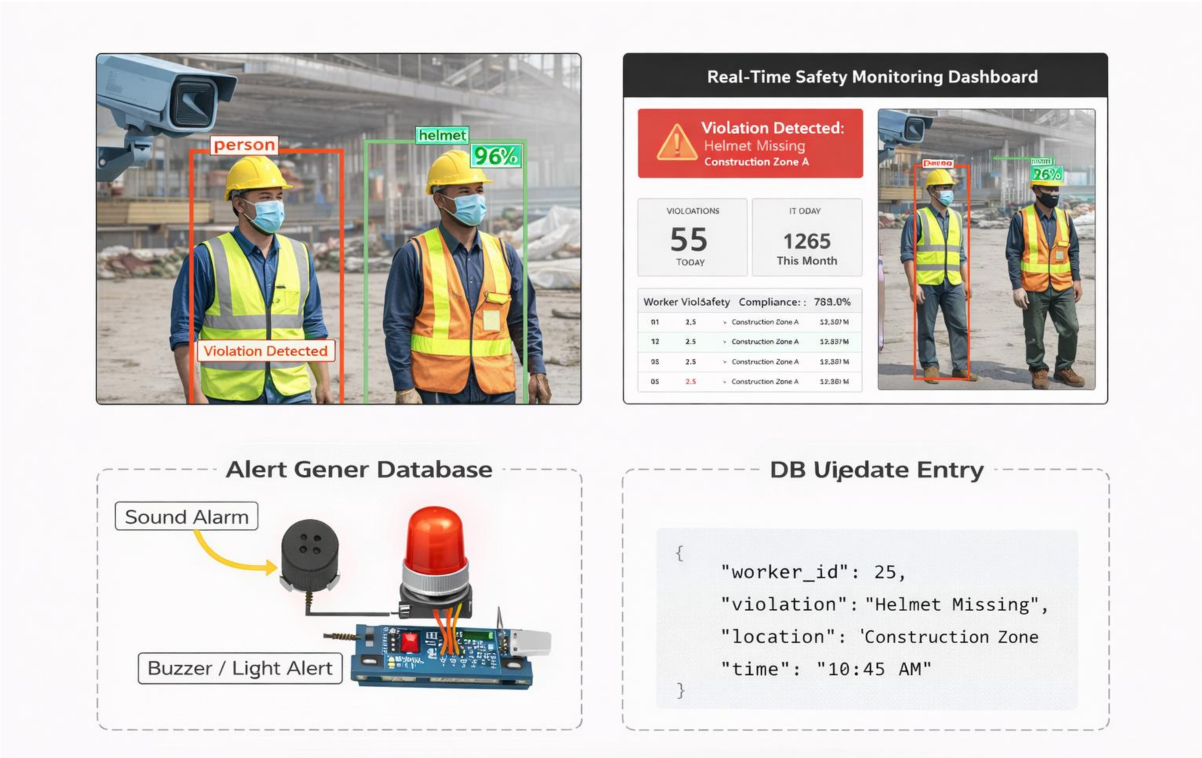
6.2. Performance Analysis

Experimental results show:

- High detection accuracy for helmets and vests.
- Average inference speed of 20–30 FPS on GPU systems.
- Reliable multi-object tracking in moderately crowded scenes.
- Low false positive and false negative rates.

ISSN: 2583-6129
DOI:10.55041/ISJEM05783

The system performs effectively under normal industrial lighting conditions.



Output of SafeVision. AI-Based PPE Detection System

7. Applications

SafeVision can be deployed in:

- Construction sites
- Chemical plants
- Steel industries
- Warehouses
- Manufacturing factories

It assists safety officers by automating compliance monitoring and providing long-term safety analytics.

8. Limitations and Challenges

Despite promising results, some challenges remain:

- Reduced accuracy under poor lighting conditions.
- Occlusion issues in highly crowded environments.
- PPE variation in color and design.
- High computational requirements for large-scale deployment.

Future improvements may include edge computing and model optimization techniques.

9. Future Work

Future enhancements include:

- SMS and WhatsApp alert integration
- Cloud-based deployment

- Mobile application development
- Predictive safety analytics
- Multi-camera synchronization
- Edge AI deployment

These improvements will enhance scalability and usability.

10.Conclusion

SafeVision presents a comprehensive AI-powered PPE detection and safety monitoring system tailored for industrial environments. By integrating real-time object detection, multi-object tracking, backend logging, and interactive dashboard analytics, the system provides a scalable and automated safety solution. The implementation demonstrates high accuracy and real-time performance, contributing to improved workplace safety compliance. With further enhancements, SafeVision can play a significant role in Industry 4.0 safety automation.

11. References

[1] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, "You Only Look Once: Unified, Real-Time Object Detection," Proc. IEEE Conf. Computer Vision and Pattern Recognition (CVPR), 2016.

[2] N. Wojke, A. Bewley, and D. Paulus, "Simple Online and Realtime Tracking with a Deep Association Metric," Proc. IEEE Int. Conf. Image Processing (ICIP), 2017.

[3] I. Goodfellow, Y. Bengio, and A. Courville, Deep Learning, MIT Press, 2016.

[4] OpenCV Documentation, https://opencv.org

[5] MongoDB Documentation, https://www.mongodb.com

[6] Node.js Documentation, https://nodejs.org

[7] React.js Documentation, https://react.dev