Grenze International Journal of Engineering and Technology, June Issue

Automated Cheating Detection in Examination Halls using YOLOv8 and Computer Vision

Jayasri B S[1] and Shashank R[2]
[1-2]The National Institute of Engineering, Department of CSE, Mysuru, India
Email: jayasribs@nie.ac.in, shashanksavitha2003@gmail.com

***Abstract*— Ensuring the integrity of examinations remains a significant challenge, particularly in large classrooms where manual supervision is constrained by human fatigue and limited attention. This paper proposes an automated cheating detection framework that leverages computer vision and deep learning to monitor student behavior in real time. A custom dataset was created by recording classroom scenarios with multiple students per frame, extracting frames using OpenCV, and annotating key behaviors with LabelImg. The system employs YOLOv8, a state-of-the-art object detection model, to recognize four primary behaviors: not cheating, turning left, turning right, and using a mobile phone. Experimental results demonstrate that the framework achieves high precision and recall while maintaining real-time processing speed. Limitations include the dataset size, reliance solely on visual cues, and the current inability to detect simultaneous cheating behaviors per student. Future work will expand the dataset, improve detection of overlapping actions, and incorporate multimodal inputs such as audio and pose estimation. The proposed approach provides scalable, unbiased monitoring, reducing reliance on manual invigilation and improving examination fairness.**

***Index Terms*— Automated exam monitoring, Cheating detection, YOLOv8, Computer vision, Deep learning.**

I. INTRODUCTION

Examinations play a crucial role in evaluating student performance and maintaining academic standards. Ensuring their integrity is essential, yet traditional monitoring methods face significant challenges, including limited invigilator availability and difficulty detecting subtle malpractice such as head movements, whispering, or the use of concealed devices. Human supervision is prone to fatigue and distraction, which can lead to undetected cheating and compromise fairness, particularly in large classrooms. Recent advances in artificial intelligence (AI) and computer vision provide promising alternatives by enabling automated monitoring of examination environments. Deep learning models, especially Convolutional Neural Networks (CNNs), can extract complex features from images and video streams for tasks such as object detection, activity recognition, and behavior classification. Real-time detection models in the YOLO (You Only Look Once) family combine speed with accuracy, making them suitable for live monitoring scenarios. This study presents an automated cheating detection framework that leverages YOLOv8 for real-time recognition of suspicious behaviors in classrooms. A custom dataset was created by recording examination scenarios containing multiple students per frame, extracting frames using OpenCV, and annotating actions such as not cheating, turning left, turning right, and using a mobile phone with LabelImg.

Grenze ID: 01.GIJET.12.2.99_21

The dataset was divided into training and testing subsets, and YOLOv8 was fine-tuned to accurately classify behaviors for each student individually. By analyzing video input frame by frame, the system can detect and alert invigilators to potential cheating in real time, complementing human supervision. While the framework reliably detects visual behaviors for multiple students simultaneously, it has some limitations. The dataset size is moderate, detection relies solely on visual cues, and the system currently struggles with identifying overlapping actions for the same student. Future enhancements will expand the dataset, improve detection of simultaneous behaviors, and explore multimodal monitoring approaches, including pose estimation and audio analysis.

II. Related Work

The existing body of literature demonstrates a progressive shift from lightweight object-detection models toward more comprehensive and integrated cheating detection frameworks. Initial studies by Afandi Nur Aziz Thohari et al. [1], S.U. Kadlag et al. [2], and Elta Maunika Priya et al. [3] establish that well-tuned YOLO variants and structured annotated datasets can deliver high accuracy for identifying behaviors such as head turning and mobile usage. These works collectively confirm that efficient deep-learning architectures, even in their compact forms, are capable of supporting real-time supervision in examination settings.

Subsequent research, including the contributions of Sonawane S. M. et al. [4], Veeramani T. et al. [5], and Tripty Singh et al. [6], highlights the advantages of integrating detection models with motion cues, pose estimation, audio analysis, and orientation tracking. Such hybrid pipelines enhance reliability by capturing multi-dimensional behavioral indicators. Additional studies by Vishal Kaith et al. [7] and Yan Zuo et al. [8] further demonstrate that attention mechanisms and resource-efficient models significantly improve fine-grained behavior recognition, especially in environments with limited computational capacity.

Complementary works from Chinmaya Nilakantha Naik et al. [9] to Fatima Mahmood et al. [13] broaden the scope by incorporating identity verification, gesture recognition, and large-scale monitoring, showcasing the practical feasibility of deploying such systems in real exam environments. Ethical and survey-based perspectives presented by Waheeb Yaqub et al. [14] and Tariq Mostaf Radwan et al. [15] emphasize privacy concerns, system transparency, and long-term research trends within automated invigilation. Building on these foundations, the present work focuses on fine-tuning YOLOv8 to detect targeted behaviors such as head deviations and mobile phone usage, with the objective of offering a scalable and reliable solution aligned with modern examination supervision requirements.

III. Proposed System

The proposed cheating detection framework addresses the shortcomings of existing approaches by combining computer vision and deep learning into a unified, real-time monitoring pipeline. The system begins by recording examination sessions, from which OpenCV extracts frames at regular intervals. These frames are annotated using LabelImg into classes including neutral, turning left, turning right, and using a mobile phone. The labeled dataset is divided into training and testing subsets, and a YOLOv8 model is fine- tuned to recognize these activities with high accuracy.

Unlike traditional methods, the proposed system provides real-time analysis, detecting suspicious activities frame by frame. Alerts are generated whenever predefined behaviors are identified, enabling invigilators to respond immediately. The lightweight nature of YOLOv8 ensures efficiency on standard GPUs, while the modular design allows integration with additional modalities such as pose estimation or audio cues in the future. The system is implemented in Python, with YOLOv8 for detection and OpenCV for video processing, ensuring both accessibility and scalability.

This approach enhances examination integrity by providing timely, accurate, and unbiased detection of cheating behaviors. By reducing reliance on manual invigilation, it minimizes human error, lowers workload, and supports fair evaluation in both physical and online examination environments.

A. Advantages of the Proposed System

- Detects suspicious behaviors (turning left, turning right, using a mobile phone) in real time, supporting quick interventions by invigilators.
- Leverages lightweight YOLOv8 models, making it suitable for deployment even on modest hardware.
- Focuses on activity recognition rather than identity tracking, reducing privacy concerns and addressing actual cheating behaviors.
- Provides a modular, extensible design that can integrate future cues such as pose estimation or audio analysis.

➢ Improves fairness and transparency in examinations by offering consistent, unbiased monitoring.

IV. SYSTEM DESIGN

System design defines the overall structure of the proposed cheating detection framework and explains how its components interact to achieve real-time monitoring. The design is modular, allowing flexibility for future extensions such as audio-based monitoring or integration with biometric authentication systems.

B. High-Level System Architecture

The high-level architecture is as shown in "Fig.1". It consists of six interconnected layers

- Input Layer: Captures exam footage from live cameras or uploaded recordings.
- Preprocessing Layer: Uses OpenCV to extract frames, resize images, normalize pixel values, and reduce noise.
- Detection Layer: YOLOv8 identifies suspicious behaviors such as turning left, turning right, or using a mobile phone.
- Output Layer: Generates bounding boxes on detected actions and maintains logs with timestamps.
- Evaluation Layer: Produces system performance metrics such as Precision, Recall, F1-score,
- and mAP.
- User Interface Layer: Provides invigilators with an interactive dashboard to upload videos, monitor detections, and access reports.

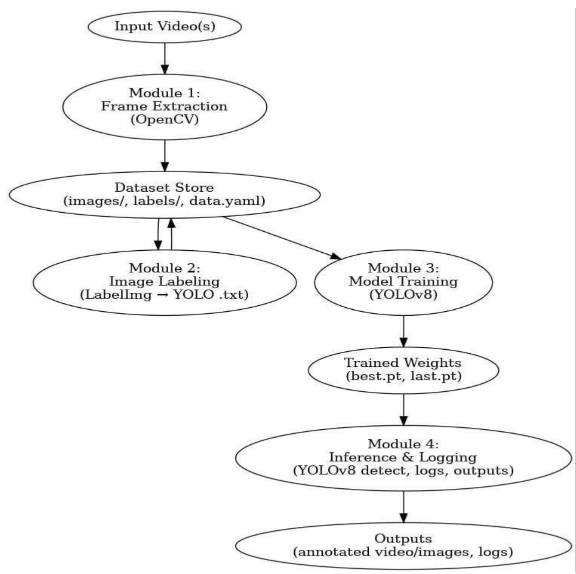

Figure 1. System architecture of the proposed cheating detection framework.

V. IMPLEMENTATION

Implementation is the process of transforming the proposed design into a fully functional system. In this stage, coding, integration, and configuration of the chosen technologies are performed. The Cheating Detection System using YOLOv8 and Computer Vision was implemented using deep learning techniques and computer vision tools. The process ensures that all modules including data preparation, training, detection, and reporting are integrated into a working system.

A. Overview of Implementation

The implementation process involved the following steps:

- Dataset Preparation: Classroom videos were collected, and cheating/non-cheating behaviors were annotated using LabelImg.
- Preprocessing: Frame extraction, resizing, and data augmentation were applied to improve robustness.
- Model Training: YOLOv8 was trained on the labeled dataset with optimized hyperparameters.

- Model Testing: The trained model was evaluated using unseen test data.
- System Integration: Preprocessing, YOLO detection, and reporting modules were combined into a single workflow.
- User Interface: A simple UI was developed to allow invigilators to upload videos and view detection results.

B. Main Aim of Implementation

The primary aim was to develop a reliable and real-time system capable of detecting cheating behaviors in examination halls. The objectives included:

- Integrating YOLOv8 for accurate object detection.
- Ensuring real-time performance for live video monitoring.
- Reducing false positives and false negatives in detection.
- Providing a simple interface for invigilators with minimal technical overhead.

C. Dataset Details

The effectiveness of any deep learning model depends significantly on the quality and quantity of data. For this project, a dataset of 300 frames was created by recording examination scenarios and manually annotating them using LabelImg. Each frame was categorized into one of the following classes: Not Cheating, Turning Left, Turning Right, Using Mobile Phone. Further the dataset was divided into two subsets to enable effective training and evaluation where 240 images (80% of the dataset) were used for training the YOLOv8 model ensuring the model learned sufficient feature representations for each class and 60 images (20% of the dataset) were reserved for evaluation. These unseen samples provide a reliable measure of model generalization and detection accuracy. To enhance the robustness of the system, preprocessing steps such as resizing, normalization, and data augmentation (flipping, rotation, and brightness adjustments) were applied. These measures helped balance class distributions and improved performance in varying lighting and pose conditions.

D. Phases in Project Implementation

Phase I – Project Initiation: The goal of this phase was to define the project at a broad level. A literature survey on cheating detection systems was conducted, where gaps such as low accuracy, lack of real-time performance, and adaptability issues were identified. Based on these insights, the central research question was formulated: “Howcan YOLOv8 and computer vision be leveraged to effectively detect cheating behaviors in examination halls?” Potential stakeholders (teachers, examiners, institutions) were analyzed to ensure that the system design met real-world needs. Finally, a system architecture and data flow diagram were prepared to map the movement of information from input (exam video) to output (detected results).

Phase II – Project Planning: During this phase, the project scope and objectives were finalized. The core aim was to build an AI-driven system capable of detecting behaviors such as head turning and mobile phone usage in real time. A timeline was prepared for dataset preparation, annotation, model training, backend integration, frontend development, and system testing. Roles were distributed by dataset annotation using LabelImg, YOLOv8 training, Flask backend development, and frontend implementation with HTML, CSS, JavaScript, and Bootstrap. The technology stack was confirmed Python 3.10 for implementation, YOLOv8 + OpenCV for detection, Flask as backend, and a lightweight web-based frontend for user interaction.

Phase III – Project Execution: This was the most critical phase, as it involved converting the planned design into a functional system by preparing the dataset wherein Exam videos were recorded, and frames were extracted using OpenCV (These frames were annotated with LabelImg, where bounding boxes were drawn for actions such as turning left, turning right, or using a mobile phone followed by Preprocessing where Images were resized, normalized, and augmented (rotation, flipping, brightness adjustment) to enhance robustness. The dataset was split into 240 training and 60 testing images. Then YOLOv8 was trained using the annotated dataset. Hyperparameters such as learning rate, batch size, and epochs were tuned to maximize performance. During training, metrics such as precision, recall, and mAP were continuously monitored.

Phase IV – Testing and Validation: The trained model was evaluated on the 60 unseen test samples. Standard metrics such as Accuracy, Precision, Recall, F1-score, IoU, and mAP@50 were computed. A confusion matrix was generated to analyze class-level strengths and weaknesses. The results showed strong performance, with Precision = 0.94, Recall = 0.92, and mAP@50 = 0.95. The inference speed was 25 FPS, confirming the system’s ability to function in real-time monitoring scenarios.

Phase V – Deployment: Finally, the model was deployed within a Flask backend, integrated with a simple web-based frontend. The system allowed invigilators to upload exam videos, which were processed by YOLOv8. Annotated videos with bounding boxes were then returned as downloadableoutputs. The user interface was designed to be intuitive, ensuring accessibility without requiring technical expertise. Testing confirmed that the complete pipeline (upload → detection → output) worked seamlessly in both offline and real-time exam monitoring environments.

VI. Results and Discussion

The proposed cheating detection framework was evaluated using the annotated dataset and tested on unseen video frames. The results highlight the system's accuracy, efficiency, and robustness. Key outputs include evaluation metrics, loss curves, dataset statistics, confusion matrix, and real-time test cases.

A. Quantitative Results

Table.1 presents the quantitative metrics. The system achieved a precision of 0.94 and recall of 0.92, resulting in an F1-score of 0.93. The mean Average Precision at 50% IoU (mAP@50) was 0.95, confirming reliable detection performance. Real-time feasibility was validated with an inference speed of 25 FPS.

TABLE I. PERFORMANCE METRICES OF THE PROPOSED SYSTEM

Metric	Value
Precision	0.94
Recall	0.92
F1-score	0.93
IoU	0.89
mAP@50	0.95
FPS (Inference Speed)	25

TABLE II: COMPARATIVE PERFORMANCE ACROSS YOLOV8 EPOCHS

Epoch	mAP@50	mAP@50-95	Precision	Recall
10	0.788	0.422	0.651	0.813
20	0.969	0.515	0.926	0.969
30	0.969	0.515	0.926	0.969
40	0.969	0.515	0.926	0.969
50	0.969	0.515	0.926	0.969

B. Comparative Analysis Across Epochs

The performance of the YOLOv8 model trained for 10, 20, 30, 40, and 50 epochs are as presented in Table.2 above. At epoch 10, the system achieved a moderate precision of 0.651 and recall of 0.813, with an mAP@50 score of 0.788. However, from epoch 20 onwards, the metrics significantly improved, reaching 0.926 precision, 0.969 recall, and 0.969 mAP@50. These values remained unchanged across epochs 20, 30, 40, and 50. This trend indicates that the model converged by epoch 20, and further training did not yield additional improvements. Thus, epoch 20 can be considered the point of optimal convergence for this dataset, ensuring both efficiency and accuracy without unnecessary computational overhead.

C. Training Performance

The YOLOv8 model was trained for 50 epochs, and the training curves provide valuable insights into the learning dynamics of the system. "Fig.2" illustrates the progression of losses and metrics over time and is as shown below:

- Classification Loss: The curve shows a consistent decline across epochs, indicating that the model gradually improved in distinguishing between cheating and non-cheating behaviors. The convergence pattern also reflects stability without significant oscillations.
- Bounding Box Loss: A steady decrease was observed in bounding box regression loss, meaning that the predicted bounding boxes became increasingly aligned with ground-truth annotations. By the later epochs, the loss plateaued, suggesting sufficient convergence.
- Precision Curve: Precision steadily increased and stabilized above 92%, highlighting that most detected cheating actionswere correctly classified with few false positives.
- Recall Curve: Recall improved significantlyduring the first 20 epochs, reaching nearly 97%. This indicates that the majority of actual cheating behaviors were successfully detected.
- Objectness Loss: The reduction in objectness (confidence) loss demonstrates the model's improved ability to correctly identify the presence of relevant objects (head orientations and mobile devices). The low final values show minimal false detections.

- mAP@50 and mAP@50-95: Both metrics showed a continuous rise, with mAP@50 reaching 0.969 and mAP@50-95 stabilizing at 0.515. This demonstrates that the system not only achieved high accuracy at a 50% IoU threshold but also maintained strong performance under stricter IoU thresholds.

In summary, the training curves confirm that the model converged effectively by epoch 20, with minimal further gains in subsequent epochs. This observation aligns with the comparative epoch analysis presented in Table II above.

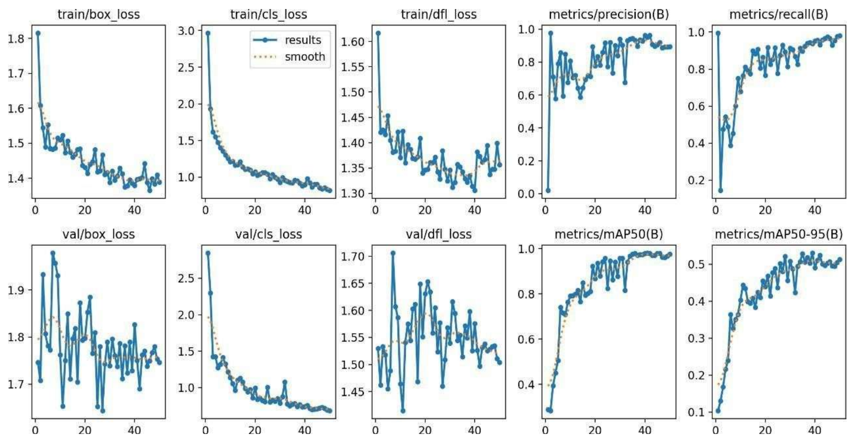

Figure 2. Training and validation performance curves for YOLOv8 model.

D. Confusion Matrix Analysis

The normalized confusion matrix is as shown in “Fig.3”, that provides a breakdown of classification accuracy per class. The system performed strongly in detecting “Not Cheating” (96%), “Turning Left” (91%), and “Using Mobile” (94%). However, performance for “Turning Right” was lower (76%), with occasional misclassification as “Turning Left” or “Background.” These results highlight the importance of balancing class samples and improving detection robustness for subtle head movements

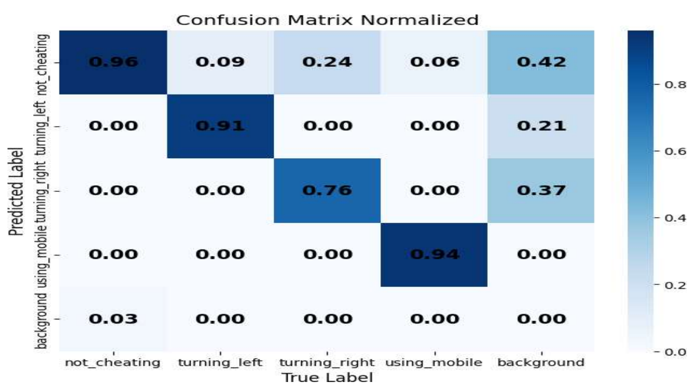

Figure 3. Normalized confusion matrix showing class-wise detection performance of the cheating detection system

To validate the performance of the cheating detection framework, several test cases were designed by considering both pass and fail test cases and is as shown in Table.3 below. These test cases focus on the main behaviors detected by the model such as neutral, turning left, turning right, and using mobile. Each test case compares the expected output with the system’s actual response.

TABLE III. TEST CASES FOR CHEATING DETECTION SYSTEM

Test Case ID	Input (Class)	Expected Output	Status
TC1	Not Cheating	No suspicious activity detected	Pass
TC2	Turning Left	Bounding box with label “Turning Left”	Pass
TC3	Turning Right	Bounding box with label “Turning Right”	Pass
TC4	Using Mobile	Bounding box with label “Using Mobile”	Pass
TC5	Using Mobile and Turning Left	Bounded box with label ”Turning Left and Using Mobile”	Fail

The test cases confirm that the system works as expected for all predefined classes. Each case consistently returned correct detections with bounding boxes and labels. This demonstrates the model's robustness in identifying suspicious activities while ensuring minimal false alarms during real-time monitoring. The visual results presented in "Fig.4, Fig.5, Fig.6, Fig.7, Fig.8", respectively demonstrate the system's capability to correctly classify different behaviors, ranging from normal conduct to suspicious activities such as head turning and mobile phone usage. The framework shows strong performance in recognizing clear and isolated actions, thereby supporting its effectiveness in real-time monitoring of examination settings. At the same time, the failed case highlights a critical limitation when multiple suspicious activities occur simultaneously. The system tends to detect only one dominant action (e.g., classifying a student turning left while also using a mobile phone as merely "turning left"). This indicates that while the model is reliable for single-label scenarios, it struggles with overlapping behaviors. Such misclassifications emphasize the need for improved multi-label learning strategies or the integration of additional modalities like pose estimation or temporal context analysis. By incorporating richer contextual information, the system can evolve into a more robust solution capable of detecting complex cheating patterns beyond isolated visual cues.

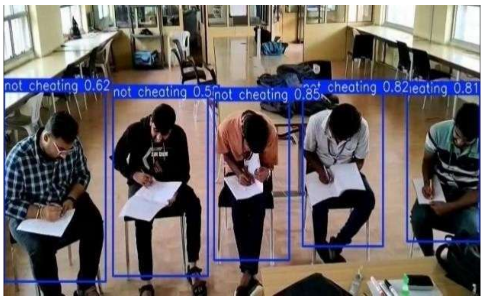

Figure 4. Not Cheating

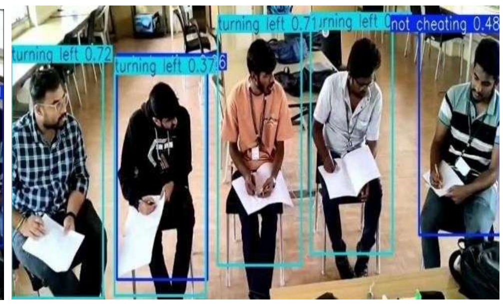

Figure 5. Turning Left

Figure 6. Turning Right

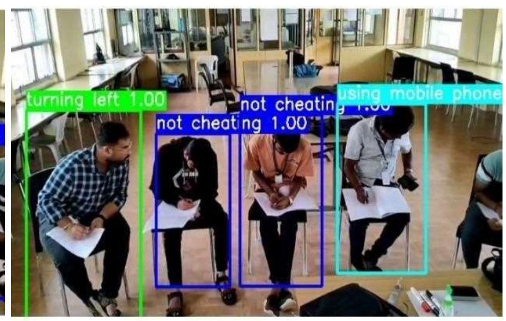

Figure 7. Mobile Phone

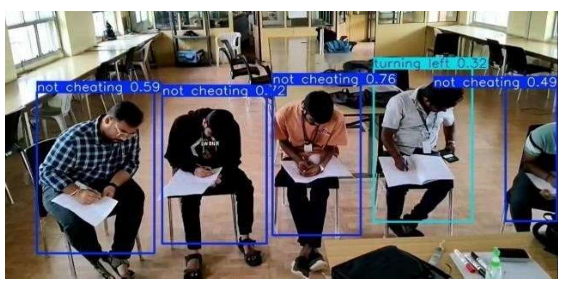

Figure 8. Failed Case - the system misclassified a student simultaneously turning left and using a mobile

VII. Conclusion and Future Work

This work presented a cheating detection framework based on YOLOv8 and computer vision techniques. The system was designed to automatically monitor examination environments and detect suspicious activities such as turning left, turning right, and using mobile phones. A custom dataset was prepared and annotated using

LabelImg, and training was performed with YOLOv8 to ensure robust detection performance. The framework was further integrated with OpenCV for preprocessing and a simple user interface for video upload and result retrieval. Experimental results demonstrated that the system achieved a precision of 0.94, recall of 0.92, F1-score of 0.93, and an mAP@50 of 0.95. These values confirm the reliability of the system for practical exam supervision. In addition, the inference speed of 25 FPS validated the model's suitability for real-time deployment. Qualitative inspection showed that the system performed well in detecting head movements and mobile phone usage, while also providing examiners with downloadable annotated video outputs as verifiable evidence. The studycontributes to the field bydemonstrating that modern object detection algorithms can effectively address real-world educational challenges, particularly those related to maintaining academic integrity. Unlike manual invigilation, which is prone to fatigue and bias, the proposed system provides continuous, unbiased, and scalable monitoring.

Although the results are promising, there is scope for enhancement in several areas. Expanding the dataset, particularly for underrepresented classes like "turning right," would reduce misclassifications and improve generalization across varied classroom environments. Similarly, addressing environmental factors such as low lighting and occlusion will increase detection robustness. Beyond visual cues, future work may integrate multimodal inputs such as pose estimation and audio analysis to capture subtle behaviors like whispering. Additionally, enabling detection of multiple simultaneous cheating behaviors will make the system more comprehensive. Using a high-quality camera for data collection and real-time monitoring will further enhance detection accuracy and reliability. Optimizing the framework for deployment on edge devices (e.g., Raspberry Pi, Jetson Nano) could make large-scale adoption feasible. Ethical and privacy concerns should also be addressed by incorporating privacy-preserving methods that avoid storing identifiable video data. Finally, extending the system to online proctoring platforms would broaden its applicability in hybrid and remote learning contexts. With continued dataset expansion, refinement, and multimodal integration, along with the use of a high-quality camera for precise monitoring, the proposed system has the potential to evolve into a comprehensive, reliable, and scalable solution for both physical and online examination environments. By addressing current limitations and implementing the proposed enhancements, this framework can significantly improve the integrity, efficiency, and consistency of exam supervision.

References

[1] A. N. A. Thohari, M. F. Lathief, L. Triyono, and K. Santoso, "Closed room cheating detection using YOLOv8s with hyperparameter tuning," Preprint or Conference, 2025.

[2] S. U. Kadlag, D. Shinde, W. Bhagyashri, T. Shete, and S. Sakshi, "Behavior-based anti-cheating using OpenPose and Decision Tree," Journal or Conference, 2025.

[3] E. M. Priya, A. R. Kumar, K. Veditha, B. H. Naidu, C. Usha, A. V. Rao, and Y. S. Rao, "Annotated dataset preparation and YOLOv5-based cheating detection," Journal or Conference, 2025.

[4] S. M. Sonawane, A. Patel, Y. Ambale, and S. Shirole, "Pariksha Nirikshak: Anti-cheating pipeline using YOLOv8, OpenPose, and Decision Tree," Journal or Conference, 2024.

[5] T. Veeramani, T. Tharun, S. Krishna, T. Kumar, and L. Prasad, "Realtime proctoring demo using YOLOv5, OpenCV, and Flask," Journal or Conference, 2024.

[6] T. Singh, R. R. Nair, T. Babu, and P. Duraisamy, "Smart proctoring combining YOLO detection, face orientation, and audio cues," Journal or Conference, 2024.

[7] V. Kaith, "Cheating detection on low-resource devices using YOLOv5- nano, MTCNN, and YAMNet, "Journal or Conference, 2024.

[8] Y. Zuo, S. S. Chai, and K. L. Goh, "Enhanced YOLOv8 with attention mechanisms for subtle behavior detection," Journal or Conference, 2024.

[9] C. N. Naik, A. Shetty, V. K. Naik, and R. C. P., "Dlib and YOLO-based online proctoring system," Journal or Conference, 2022.

[10] S. Mishra, R. S., R. S., and R. S., "Classroom cheating detection using YOLOv3 with residual backbone," Journal or Conference, 2023.

[11] X. Yang, et al., "iExam: Online exam monitoring system addressing identity fraud," Journal or Conference, 2022.

[12] L. C. O. Tiong, et al., "Misconduct detection framework using IP tracking and Dense LSTM," Journal or Conference, 2022.

[13] F. Mahmood, et al., "Exam supervision system detecting gestures and whispers using Faster R-CNN and MTCNN," Journal or Conference, 2022.

[14] W. Yaqub, et al., "Privacy-preserving proctoring using image hashing," Journal or Conference, 2021.

[15] T. M. Radwan, et al., "Survey on cheating detection methods: Video analysis, pose estimation, and anomaly detection," Journal or Conference, 2022.