International Journal of Science, Strategic Management and Technology
Volume **02** Issue **06** June-**2026** | ISSN: **3108-1762 (Online)** | Impact Factor: 3.8
An International, Peer-Reviewed, Open Access Scholarly Journal Indexed in recognized academic databases

AI-Based Offline Exam Hall Monitoring System Using YOLOv8, DeepSORT, and MediaPipe with Multi-Angle Supervisor Registration

Laukik Pradip Ingale
Dept. of Electronics and Telecommunication Engineering
GES's R. H. Sapat College of Engineering, Management Studies & Research
Nashik – 422005, India
22_laukik.ingale@ges-coengg.org

Mehul Tulsidas Katakiya
Dept. of Electronics and Telecommunication Engineering
GES's R. H. Sapat College of Engineering, Management Studies & Research
Nashik – 422005, India
22_mehul.katakiya@ges-coengg.org

Prathmesh Bhagwat Wagh
Dept. of Electronics and Telecommunication Engineering
GES's R. H. Sapat College of Engineering, Management Studies & Research
Nashik – 422005, India
22_prathmesh.wagh@ges-coengg.org

https://doi.org/10.55041/ijsmt.v2i6.199

Cite this Article: Ingale, L. P., Katakiya, M. T. & Wagh, P. B. (2026). AI-Based Offline Exam Hall Monitoring System Using YOLOv8, DeepSORT, and MediaPipe with Multi-Angle Supervisor Registration. International Journal of Science, Strategic Management and Technology, *02*(6). https://doi.org/10.55041/ijsmt.v2i6.199

ABSTRACT

Malpractice during offline pen-and-paper examinations remains a persistent and largely unsolved problem because a small number of human invigilators cannot maintain consistent attention across a crowded hall for several hours at a stretch. This paper presents the design, implementation, and experimental evaluation of an AI-based, fully offline examination monitoring system that uses computer vision to detect suspicious student behaviour in real time. The proposed pipeline combines YOLOv8n for person detection, DeepSORT for persistent multi-object tracking, and MediaPipe for facial, postural, and hand-landmark extraction, feeding a lightweight rule-based behavioural-scoring engine that accumulates a suspicion score per student. A distinguishing contribution is a multi-angle supervisor-registration module, built on K-Means clustering over landmark features captured from four viewing angles, which allows the system to reliably recognise the invigilator and exclude their own movement from malpractice checks. On real classroom video, the system sustained an average of 18.7 frames per second on commodity laptop hardware, generated alerts with a mean latency of 5.39 seconds, and achieved 93.17% head-pose classification accuracy, 96.0% supervisor-identification accuracy, and 94.0% accuracy for combined cheating and passing detection, entirely without internet connectivity. These results indicate that a carefully engineered combination of existing open-source computer-vision components can deliver a practical, affordable, and privacy-respecting alternative to manual-only invigilation.

Keywords—*Computer Vision, Deep Learning, Exam Hall Monitoring, Image Processing, Machine Learning, Malpractice Detection, Offline System, Real-time Monitoring, YOLOv8, DeepSORT, MediaPipe.*

I. INTRODUCTION

Examinations have always sat at the centre of academic evaluation, and the trust placed in their results depends on an assumption that is rarely questioned closely: that the hall is being watched closely enough, by enough people, for long enough, to make dishonesty costly. Frequent reports of malpractice in offline examinations underscore the urgent need for an efficient, reliable, and scalable monitoring approach.

Conventional invigilation relies almost entirely on manual supervision. A single invigilator is often responsible for thirty to fifty students across two or three hours, and sustained human attention simply does not hold up under that load — focus narrows, certain rows get watched more than others, and fatigue accumulates as the session continues, often without any institutional negligence to explain it.

The proliferation of computer vision and embedded AI has opened transformative possibilities for automated invigilation. Object detectors such as YOLOv8, multi-object trackers such as DeepSORT, and landmark-extraction frameworks such as MediaPipe are now light enough to run in real time on a standard laptop rather than a research cluster. Edge inference further means that no footage needs to leave the examination room, sidestepping the privacy concerns that surround cloud-based proctoring tools.

The system proposed in this paper leverages exactly this combination. YOLOv8n performs person detection, DeepSORT maintains persistent student identities across frames, and MediaPipe extracts head-pose, posture, and hand-landmark data used by a behavioural-scoring engine to flag suspicious activity. A multi-angle supervisor-registration module, built on K-Means clustering, allows the system to recognise the invigilator from any direction and exclude their own movement from malpractice checks — addressing a gap left open by systems that register only a single frontal image.

The primary objective of this work is to deliver a reliable, low-cost, and technologically robust solution for enhancing examination integrity through fully offline, AI-based monitoring.

II. LITERATURE REVIEW

Substantial research has been conducted on AI-assisted examination monitoring, employing a diverse range of techniques to enhance detection accuracy. The following subsections critically review notable prior work.

Sushmita et al. proposed a real-time cheating-detection system for offline exams using YOLOv3 combined with ShuffleNet on CCTV feeds, achieving 88.03% accuracy and demonstrating that lightweight deep-learning pipelines can run practically in real classrooms [1]. Ramzan et al. treated exam-hall behaviour as a temporal sequence, combining 2D and 3D CNNs over motion-based key-frames to reach an AUROC of 0.94, showing that context across frames captures patterns invisible in single images [2]. Bancud et al. used OpenPose with an XGBoost classifier to detect cheating from body posture rather than facial cues alone, reaching approximately 90% accuracy at 10 FPS [3].

Genemo introduced a hybrid architecture (L4-BranchedActionNet, a modified VGG-16) combined with entropy coding, ant-colony optimisation, and SVM/KNN classification, achieving 92.99% accuracy [4]. Hussein et al. demonstrated that handcrafted descriptors such as HOG and SURF, fed into a multi-class SVM, still achieve 91% accuracy without large datasets or heavy compute [5]. Singh et al. estimated head-pose angles via facial landmarks and DNN regression, achieving a mean angular error under three degrees — directly relevant since sustained head turning is a primary visual cue for copying [6].

Asadullah et al. took an audio-only approach, using FFT analysis in MATLAB to detect whispering patterns, achieving a 1% false-acceptance rate and 3% false-rejection rate [7]. Muchangi et al. developed a deep-learning system for online proctored exams observing posture, gaze, and movement [8]. Tong Liu proposed a two-stream CNN separately processing spatial appearance and temporal motion, reaching 89.1% accuracy [9]. Li et al. fused behavioural observation with academic performance data via a feed-forward network and LSTM, reaching 79.4% accuracy and 81% recall [10].

Unlike many existing systems that depend on a single sensing modality or assume online-exam connectivity [8], [9], the system proposed here

combines detection, tracking, and dense landmark analysis into one fully offline pipeline. Compared to more complex multi-technology approaches [4], [9], this design remains comparatively lightweight while still delivering real-time tracking, behavioural scoring, and a dedicated supervisor-exclusion mechanism not addressed by prior work.

III. SYSTEM ARCHITECTURE

The proposed system consists of several hardware and software components working together to monitor student behaviour and raise alerts.

Main components include:

- Laptop or desktop processing unit (CPU-based, with optional GPU acceleration)
- USB or IP camera (1080p minimum)
- YOLOv8n person/object detector
- DeepSORT multi-object tracker
- MediaPipe landmark-extraction framework
- Tkinter-based invigilator dashboard

Fig. 1 System Architecture

As shown in Fig. 1, the system begins with video ingestion from CCTV, IP, or USB cameras. Each frame is pre-processed and passed to the YOLOv8n detection layer, which locates students and relevant objects. DeepSORT then assigns a persistent tracking identity to every detected student, surviving brief occlusion or movement out of frame.

The cropped region around each tracked student is analysed by MediaPipe, which extracts head-pose, posture, and hand-landmark data. A behavioural-intelligence layer interprets these signals through a gaze classifier, an event engine that accumulates and decays a suspicion score, and an interaction detector that watches for coordinated behaviour between neighbouring students. The output layer renders annotated video with bounding boxes and alert labels directly on the invigilator's dashboard, while all events are logged locally as timestamped CSV records for post-exam review.

A. Hardware Components

The hardware configuration is built around three components: a camera module mounted six to eight feet up for an unobstructed view of the seating area; a processing unit (laptop or desktop with an Intel i5/i7 or AMD Ryzen 5/7 CPU, minimum 8 GB RAM, and an optional NVIDIA GPU such as the RTX 3050 for CUDA acceleration); and a display unit that renders the dashboard directly on the invigilator's own screen, requiring no external monitor.

B. Supervisor Registration Technique

Before the exam begins, four images of the invigilator are captured from different directions — front, back, and both sides. MediaPipe extracts a landmark feature vector from each image, and K-Means clustering ($k = 4$) builds a multi-angle profile stored as cluster centroids. During the exam, any detected person within a distance threshold of 0.35 from a centroid is classified as the supervisor and excluded from all malpractice checks. This multi-angle approach is markedly more robust than registering a single frontal photograph, which fails as soon as the invigilator turns away from the camera.

IV. REAL-TIME MONITORING — WORKING PROCESS

This section describes the sequential operational workflow of the proposed system, from initialization through continuous real-time monitoring.

Stage 1 — Initialization: On launch, the system configures the camera feed, loads the YOLOv8n, DeepSORT, and MediaPipe models, and verifies that all components are ready for operation.

Stage 2 — Person Detection: YOLOv8n performs single-pass detection across each frame, returning bounding boxes with confidence scores; Non-Maximum Suppression (IoU = 0.45) removes

duplicates, and only the "person" class above 0.5 confidence is retained.

Stage 3 — Tracking Assignment: DeepSORT assigns a persistent student identifier to each detection using a Kalman filter for motion prediction, a deep appearance descriptor for re-identification after occlusion, and the Hungarian algorithm for optimal track assignment.

Stage 4 — Landmark Extraction: MediaPipe analyses the cropped region around each tracked student, extracting head-orientation keypoints (nose, both ears), shoulder/wrist posture, and hand-landmark configuration.

Stage 5 — Head-Pose Classification: Using nose (N) and ear (L, R) keypoints with confidence threshold $\tau = 0.5$, horizontal distances $d_L = |x_R - x_N|$ and $d_r = |x_R - x_N|$ are compared. With sensitivity factor $k = 2.5$: $d_L > k \cdot d_R$ classifies as "Looking Right"; $d_R > k \cdot d_L$ as "Looking Left"; otherwise "Forward". If the nose is undetected but an ear remains visible, the subject is classed "Looking Back".

Stage 6 — Suspicion Scoring: Each tracked student maintains a suspicion score S_i that accumulates weighted increments for head-turns, leaning, hand activity, and proximity interaction, decaying 0.5 units per clean frame.

Stage 7 — Alert Dispatch: When S_i exceeds a threshold of 15, an alert is generated, logged with a timestamp, and displayed on the dashboard; a 90-frame cooldown then suppresses duplicate alerts for that student.

Stage 8 — Dashboard Monitoring: The invigilator views annotated live video, behavioural-state counts, and alert notifications on a single Tkinter-based screen, with no technical training required to operate it.

Stage 9 — Continuous Monitoring Loop: The system cyclically repeats detection, tracking, landmark extraction, scoring, and alerting until the session ends, ensuring uninterrupted real-time coverage of the entire hall.

V. EXPERIMENTAL SETUP AND IMPLEMENTATION

A. System Setup

The proposed monitoring system was implemented through the integration of a standard camera, a commodity processing unit, and an open-source computer-vision software stack. The software environment comprises Python with OpenCV for video ingestion, the Ultralytics YOLOv8 implementation for detection, the DeepSORT tracking library, MediaPipe for landmark extraction, and Tkinter for the invigilator dashboard.

B. Hardware Implementation

- HD or IP camera for capturing live video of the exam hall
- Laptop or desktop with Intel i5/i7 or AMD Ryzen 5/7 processor
- Minimum 8 GB RAM (16 GB recommended)
- NVIDIA GTX 1660, RTX 3050, or equivalent GPU (optional)
- Stable power supply for all connected components

The camera was mounted to maximise an unobstructed, elevated view of the seating arrangement. Preliminary testing confirmed that detection accuracy is sensitive to lighting and camera angle, with performance degrading under uneven illumination or significant occlusion between students.

C. Software Implementation

The detection pipeline resizes each frame to 640×640 using bilinear interpolation and converts it from BGR to RGB before inference. YOLOv8n returns bounding boxes that are passed to DeepSORT for persistent tracking, and the cropped region for each track is forwarded to MediaPipe's Face Mesh, Pose, and Hand Landmark modules. All behavioural thresholds — suspicion increments, decay rate, alert threshold, and cooldown duration — are exposed through a single configuration file, allowing institutions to tune sensitivity without modifying code.

D. Experimental Procedure

The system was evaluated on real classroom video recorded under standard indoor lighting. The evaluation procedure was conducted as follows: the camera feed was initiated and the detection pipeline allowed to stabilise; YOLOv8n and DeepSORT processed each frame to maintain persistent student identities; MediaPipe extracted head-pose and posture data for every tracked student; the behavioural engine accumulated suspicion scores and triggered alerts upon threshold breach; and the supervisor-registration module was tested by having the invigilator move around the hall to confirm correct exclusion from

International Journal of Science, Strategic Management and Technology
Volume **02** Issue **06** June-**2026** | ISSN: **3108-1762 (Online)** | Impact Factor: 3.8
An International, Peer-Reviewed, Open Access Scholarly Journal Indexed in recognized academic databases

malpractice checks. Multiple sessions were run to assess head-pose accuracy, supervisor-identification reliability, alert latency, and frame-processing throughput.

VI. RESULTS AND DISCUSSION

The integrated pipeline — combining YOLOv8, DeepSORT, and MediaPipe — was evaluated across head-pose accuracy, supervisor/cheat detection metrics, alert latency, frame throughput, and false-detection rates.

Head-pose classification reached 93.17% overall accuracy and 90.94% macro F1, with the Forward and Right classes both around 95.6% and the Left and Back classes between 84% and 88%, the latter attributable to side-angle occlusion and reduced landmark visibility. Supervisor identification reached 96.0% accuracy and 94.3% F1, confirming that the multi-angle K-Means registration reliably distinguishes the invigilator from students. Combined cheating and passing detection reached 94.0% accuracy with a deliberately conservative 81.8% recall, since in a real exam setting the cost of wrongly accusing a student is considerably higher than occasionally missing a borderline movement.

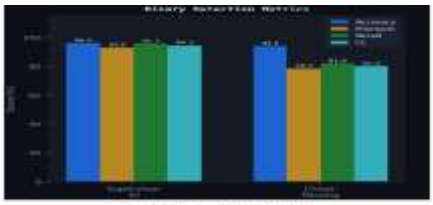

Fig. 2 Binary detection metrics — supervisor ID and cheat/passing

Mean alert latency was 5.39 seconds against a configured 5.0-second target, with 90–95% of all alerts generated within 6.3 seconds and a smoothly rising cumulative distribution indicating stable, predictable alerting behaviour across the monitoring session.

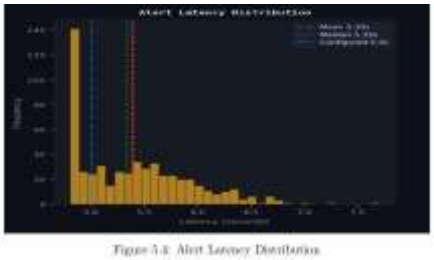

Fig. 3 Alert latency distribution

Frame throughput averaged 18.7 FPS, comfortably above the 15 FPS benchmark generally considered necessary for real-time monitoring, with most frames processed in the 17–22 FPS range even when several students were simultaneously visible. Higher frame rates were consistently associated with lower alert latency, and minor processing dips were tolerated without meaningful loss of reliability.

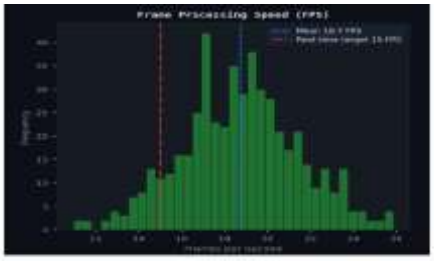

Fig. 4 Frame processing speed (FPS) distribution

Error analysis showed balanced false-positive and false-negative rates near 9% for head-pose classification, very low error on both sides for supervisor identification, and a higher 18% false-negative rate for cheat detection — the direct consequence of the conservative threshold tuning discussed above.

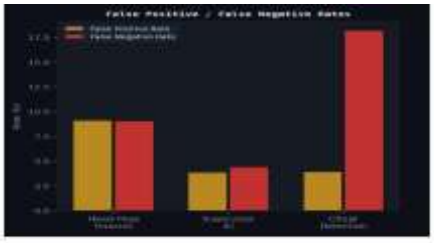

Fig. 5 False-positive and false-negative analysis

Table I. Overall System Performance Summary

Parameter	Performance
Head-Pose Accuracy	93.17%
Head-Pose Macro F1	90.94%
Supervisor ID Accuracy	96.00%
Supervisor ID F1	94.29%
Cheat Detection Accuracy	94.00%
Cheat Detection F1	80.00%
Mean Alert Latency	5.39 s
Mean Processing FPS	18.72
False Negative Rate (Cheat)	18.18%

Qualitatively, the system maintained stable per-student tracking identifiers, correctly labelled the invigilator "SUPERVISOR" and excluded them from checks throughout, and correctly flagged a

student crossing the head-pose threshold as "LOOKING AWAY" with the corresponding on-screen label, confirming that the full detection-to-alert chain operates end to end on real footage.

Fig. 6 ExamGuard monitoring dashboard

The fig.7 shows the system running in a real environment. Students are enclosed in bounding boxes and assigned tracking IDs. The invigilator is correctly identified as "SUPERVISOR" and excluded from all malpractice checks. A student detected as "LOOKING AWAY" is highlighted in red with the corresponding label, confirming that the real-time detection pipeline works as intended.

Fig. 7 Malpractice detected using bounded boxes and labels

VII. ADVANTAGES, APPLICATIONS, AND LIMITATIONS

The system offers continuous real-time monitoring without the attentional decline that affects human supervision, runs on modest hardware thanks to the lightweight YOLOv8n detector, reliably differentiates the supervisor through multi-angle K-Means registration, and presents an accessible single-screen dashboard requiring no technical background to operate. Direct applications include college and university examination halls, competitive entrance-examination centres, corporate assessment testing, and as a local fallback wherever internet-based proctoring is unavailable or undesirable.

Several limitations remain. Detection accuracy degrades in poor or uneven lighting, and camera placement strongly affects occlusion-related miscounts in tightly packed halls. A single camera cannot cover a large hall without blind spots, and the K-Means supervisor model may fail if the invigilator changes clothing or lighting shifts markedly between registration and the exam itself. The system has no audio channel, so whispering-based malpractice falls outside its current scope, and as a vision-only system it cannot detect concealed notes or devices that produce no visible movement.

VIII. CONCLUSION

This paper presented an AI-based offline examination monitoring system combining YOLOv8 detection, DeepSORT tracking, and MediaPipe pose analysis to enable continuous, unbiased supervision of an exam hall. The implemented supervisor-registration mechanism successfully excludes the invigilator's own movement from malpractice checks, and the behavioural-scoring engine reliably flags suspicious head movement and inter-student interaction in real time. Experimental results confirmed 93.17% head-pose accuracy, 96.0% supervisor-identification accuracy, and 94.0% cheat-detection accuracy, validating the system's reliability for real-world deployment. The fully offline, low-cost hardware design further enhances accessibility, making it a viable integrity solution for a wide range of institutions. Future work will focus on multi-camera integration, audio-based whisper detection, and adaptation of the same core logic toward online proctoring.

ACKNOWLEDGMENT

The authors would like to express their heartfelt gratitude to the Department of Electronics and Telecommunication Engineering, GES's R. H. Sapat College of Engineering, Management Studies and Research, Nashik, for their continuous support, guidance, and for providing the necessary facilities and resources that made this research possible. We also sincerely thank our project guide, Mrs. H. H. Kulkarni, and our colleagues for their valuable suggestions and encouragement throughout this work.

REFERENCES

[1] M. Sushmita et al., "Automatic cheating detection in exam hall," 2023.

[2] M. Ramzan et al., "Automatic unusual activities recognition using deep learning in academia," 2022.

[3] G. Emmanuel Bancud et al., "Human pose estimation using machine learning for cheating detection," 2021.

[4] M. Genemo Musa Dima, "Suspicious activity recognition for monitoring cheating in exams," 2022.

[5] F. Hussein et al., "Advances in contextual action recognition for cheating detection," 2022.

[6] T. Singh et al., "Attention span prediction using head-pose estimation with deep neural networks," 2021.

[7] M. Asadullah et al., "An automated technique for cheating detection," 2017.

[8] K. Muchangi et al., "Behavioral detection and prevention of cheating during online exams," 2023.

[9] T. Liu, "AI proctoring for offline examinations with 2-longitudinal-stream CNNs," 2023.

[10] Z. Li et al., "Multi-index examination cheating detection based on neural network," 2019.