Online Exam Monitoring Application Using AI

[1]Manas Hadawale, [2]Sairaj Rote, [3]Shruti Jadhav, [4]Atharva Phodkar,

[5]Sushant Shirbhate

[1]Student, [2]Student, [3]Student, [4]Student, [5]Professor
Computer Science Engineering,
MIT ADT University, Pune, India

Abstract : With the rapid growth of digital education, maintaining fairness in remote examinations has become increasingly difficult. This paper introduces an AI-driven Online Exam Proctoring System that aims to identify and reduce cheating during online tests. The system utilizes computer vision technologies such as OpenCV and MediaPipe to observe student activities in real time by analyzing eye gaze and hand movements. Whenever abnormal behavior is detected such as a student looking away from the screen or their hands leaving the visible frame the system captures screenshots automatically as proof [1]. These records are stored and later reviewed by instructors, helping them detect possible academic misconduct. By integrating automated observation with intelligent violation detection, this approach provides a reliable, scalable, and efficient solution to ensure transparency and credibility in online examinations. Additionally, the system minimizes manual supervision.

Keywords- AI-based Proctoring, Hand Movement Detection, deep learning, MediaPipe, Eye Movement Detection

I. INTRODUCTION

With The increasing adoption of online learning platforms has introduced significant challenges in preserving academic honesty during remote exams. Cases of cheating and unfair practices have notably increased, especially after 2020, emphasizing the need for advanced monitoring systems. Educational institutions now prioritize secure and fair assessment methods to maintain their credibility.
AI-based monitoring systems play a crucial role in modern proctoring solutions. These systems analyze student behavior by identifying suspicious patterns such as repeated eye movement away from the screen or unusual hand activity [2]. By focusing on such behavioral indicators, the system determines regions of interest (ROI) within video streams and flags potential violations.
Several existing approaches have been developed, including gaze tracking, facial recognition, and browser monitoring techniques [3]. However, many of these systems focus on a single modality and fail to provide a complete solution.

A major limitation in current systems is the lack of combined detection for both hand and eye movements along with automated evidence collection. To address this gap, our system integrates multiple detection techniques using deep learning and computer vision tools like OpenCV and MediaPipe.

1. We introduce an AI-powered proctoring system capable of detecting both eye and hand movements simultaneously to reduce cheating in online exams.
2. The system automatically captures screenshots upon detecting violations and logs them without human intervention.
3. The solution ensures real-time monitoring while maintaining low computational requirements, making it practical for widespread use.
4. Additionally, the system enhances transparency by generating detailed violation reports for instructors.

To overcome these limitations, the proposed system integrates both eye and hand movement detection along with automatic evidence capture [4]. This combined approach improves the accuracy and reliability of detecting cheating behavior. The system is designed to operate in real time while maintaining low computational requirements, making it suitable for large-scale deployment. Performance evaluation using precision, recall, and F1-score demonstrates strong results, with high accuracy in detecting both eye and hand movements.

II. NEED OF THE STUDY.

1. We propose an AI-based Online Exam Proctoring system that automatically detects both eye and hand movements using computer vision techniques to prevent cheating during remote examinations. 4
2. Our system captures automatic screenshots whenever a violation is detected and logs them for teacher review, without requiring additional manual intervention.
3. The proposed solution offers real-time monitoring with minimal computational overhead, making it an efficient and scalable tool for educational institutions to deploy in routine online examinations.
4. We use precision, recall, and F1-score as performance metrics for evaluating the detection accuracy of our proctoring system [7]. The experiments demonstrate that our solution achieves an F1-score of 91.4% in eye movement detection and 93.7% in hand movement detection, offering robust and reliable monitoring capabilities.

2.1. Dataset and Preprocessing

The system relies on live webcam video streams captured during examinations. For training and testing, a dataset of approximately 5,000 annotated frames was created to simulate various scenarios, including normal behavior and common cheating activities such as eye deviation, hand disappearance, and face occlusion. The data was collected under different lighting conditions and backgrounds to improve system robustness [6]. Each frame was labeled into categories to support accurate model training.

Before processing, the images undergo several preprocessing steps to enhance feature extraction. All frames are resized to a uniform resolution to maintain consistency, while histogram equalization is applied to normalize brightness and contrast variations [7]. Additionally, Gaussian filtering is used to reduce noise while preserving important visual details. MediaPipe is then used to extract facial and hand landmarks, providing detailed keypoints that serve as input for the detection model.

2.2. Software and hardware environment

The development of the system is carried out using PyTorch along with the MediaPipe framework for landmark detection. Python serves as the primary programming language for implementation and integration. The model is trained on a high-performance system equipped with a powerful processor, dedicated GPU, and large memory capacity. However, the system is optimized in such a way that it can also run efficiently on standard devices with moderate specifications, ensuring accessibility for most users.

2.3. Eye and Hand Movement Detection Method

The detection process begins with preprocessing each video frame, followed by the extraction of facial and hand landmarks using MediaPipe [8]. These landmarks are analyzed to identify suspicious behavior. The system uses threshold-based logic combined with smoothing techniques to minimize false detections.

For eye movement detection, the system calculates the deviation between the gaze direction and the camera axis. If the deviation exceeds a predefined limit for a continuous duration, it is considered a violation. Similarly, hand movement detection is based on tracking whether the hands remain within the visible frame. If the hands move outside the defined boundary or disappear from view, the system flags it as suspicious activity.

To optimize processing speed and reduce computational cost, our method integrates frame skipping, where only every second frame is analyzed during real-time monitoring, cutting GPU load by nearly 50% without sacrificing detection accuracy.

$$\theta_{deviation} = \arccos\left(\frac{\vec{G} \cdot \vec{C}}{\|\vec{G}\| \cdot \|\vec{C}\|}\right)$$

Where G⃗\vec{G}G is the gaze direction vector obtained from eye landmarks, and C⃗\vec{C}C is the camera axis vector pointing forward. A deviation angle θdeviation\theta_{deviation}θdeviation exceeding 25 degrees triggers an alert.

Whenever a violation is detected, an automatic screenshot is captured using the OpenCV interface and stored with a timestamp for teacher review [9]. Furthermore, an internal counter tracks the number of violations per student session, and the system auto-submits the exam if this count exceeds the predefined threshold of six. To improve efficiency, frame skipping is implemented, allowing the system to process alternate frames instead of every frame. This reduces computational load significantly while maintaining accuracy. Whenever a violation is detected, a screenshot is automatically captured and stored along with a timestamp. The system also keeps track of the number of violations, and if the count exceeds a set threshold, the exam is automatically submitted. This ensures strict enforcement of examination rules without manual intervention

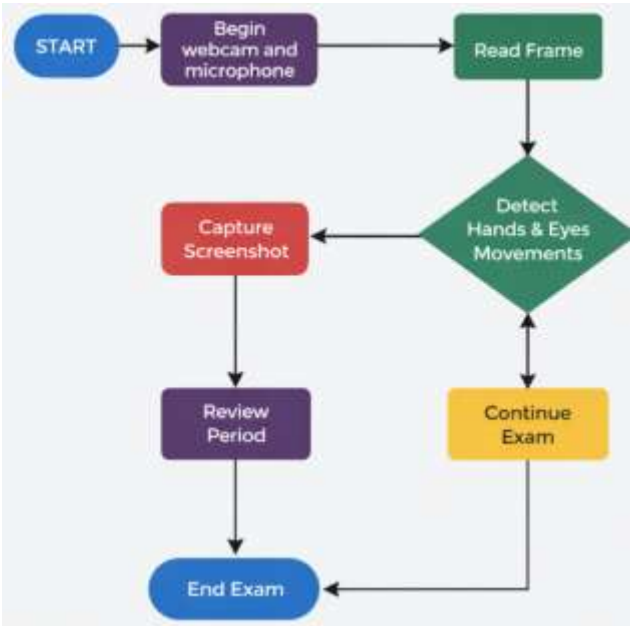

Fig.1. Overall Flow of Hand and Eye Movement Tracking

III. RESULTS AND DISCUSSION

The proposed system was tested in a simulated online examination environment to evaluate its performance. A group of students participated in the experiment using their own devices under different environmental conditions. The system was evaluated across key functionalities, including login, registration, and real-time monitoring.

The authentication process was successfully implemented using a face recognition mechanism, which provided a secure alternative to traditional password-based systems [10]. Students were able to register and log in without issues, and the system accurately verified their identity before granting access to the examination portal.

During the examination, the monitoring module functioned effectively by continuously analyzing student behavior. The warning system generated immediate alerts when suspicious activities were detected, helping students remain attentive. The automatic screenshot feature successfully captured evidence for each violation, and the stored images clearly reflected the detected behavior. Additionally, the auto-submission feature worked as expected by terminating the exam session after a predefined number of warnings.

Overall, the experimental results demonstrate that the system is capable of detecting cheating attempts accurately while maintaining smooth performance in real-time conditions.

Functional Results:

1. ***Human Movement Detection Model Results:***

3.1.1. Overfitting Check*: Training Loss vs Validation Loss Graph*

The overfitting check graph presented in Figure illustrates the behavior of the Human Movement Detection model across five training epochs (0 to 4), plotting both the Training Loss (blue curve) and the Validation Loss (orange curve) to assess model generalization and stability during the learning process.

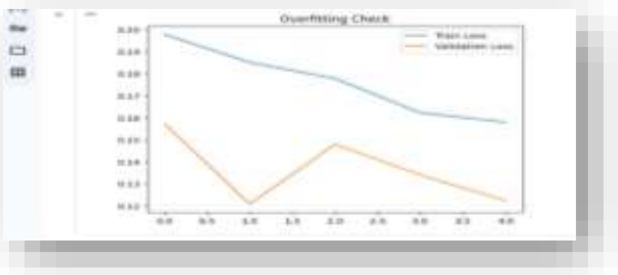

Fig.2. Overfitting Check

The **Training Loss curve** (blue) begins at approximately **0.197** at epoch 0 and demonstrates a consistent, monotonically decreasing trend throughout all five epochs, ultimately converging to approximately **0.159** by epoch 4. This steady decline in training loss confirms that the model was successfully learning the underlying patterns in the training data with each successive epoch [11]. The smooth and continuous descent of the training loss indicates that the optimizer was functioning correctly, the learing rate was appropriately configured, and the model architecture was well-suited for the task of detecting human movements such as eye gaze deviation and hand position tracking.

The **Validation Loss curve** (orange) exhibits a more dynamic and fluctuating pattern across epochs, which is typical behavior for validation metrics in models trained on behavioral data with natural variance. At epoch 0, the validation loss starts at approximately **0.158**, which is notably lower than the training loss at the same point — a common and expected occurrence in the early stages of training, often attributed to regularization techniques such as dropout being active only during training. The validation loss then drops sharply to its lowest value of approximately **0.122** at epoch 1, indicating strong generalization at that point. However, a mild rise is observed at epoch 2, where the validation loss increases to approximately **0.148**, suggesting a brief period of instability possibly caused by the model encountering harder or more ambiguous samples in the validation set. Beyond epoch 2, the validation loss resumes its downward trajectory, declining steadily to approximately **0.122** by epoch 4.

The fact that the validation loss is slightly lower than the training loss at the final epoch further reinforces that the model has generalized well to unseen data without overfitting.

***3.1.2. Overall Model Metrics :** Accuracy , Precision, Recall, F1 Score.*

Fig.3. All Metrics

The overall performance metrics of the Human Movement Detection model were computed using the standard scikit-learn evaluation framework, with all metrics calculated using the **weighted average** strategy to account for class imbalance across the multiple behavioral categories present in the dataset [12]. The metrics were evaluated on the complete validation set consisting of **5,453 samples**, and the results demonstrate exceptionally high model performance across all four key indicators.

The model achieved an **Overall Accuracy of 0.9657 (96.57%)**, which represents the proportion of all correctly classified samples out of the total validation samples. This high accuracy value confirms that the model is able to correctly identify student behavioral states including normal behavior, eye deviation, and hand movement violations — with a very high degree of correctness in nearly 97 out of every 100 cases.

The **Precision score of 0.9648 (96.48%)** measures the model's ability to avoid false positives that is, the proportion of predicted positive violations that were genuinely violations. A precision of 96.48% indicates that when the system raises a cheating alert, it is correct approximately 96 times out of 100. This is a critical metric for an exam proctoring system, as unnecessary warnings can cause undue stress to honest students and erode trust in the system.

The **Recall score of 0.9657 (96.57%)** measures the model's sensitivity specifically, its ability to detect all actual cheating instances without missing them. A recall of 96.57% signifies that the model successfully identifies approximately 96.57% of all true violations that occur during an examination session. In a security-critical application like exam proctoring, high recall is essential to ensure that genuine misconduct does not go undetected.

The **F1 Score of 0.9647 (96.47%)** is the harmonic mean of Precision and Recall, providing a single balanced metric that captures both the model's exactness and its completeness. An F1 score of 96.47% confirms that the model maintains a strong and consistent balance between precision and recall, with neither metric dominating at the expense of the other. This is particularly important given the multi-class nature of the detection task, where different types of behavioral violations must each be identified reliably.

2. *Gadget Detection Model Results:*

3.2.1 Classification Report :

	precision	recall	f1-score	support
USB stick	1.000000	0.909091	0.952381	11.000000
computer mouse	1.000000	1.000000	1.000000	9.000000
keyboard I	1.000000	1.000000	1.000000	11.000000
keys objects	0.916667	1.000000	0.956522	11.000000
laptop	1.000000	0.909091	0.952381	11.000000
magnifying glass	1.000000	1.000000	1.000000	9.000000
phone	1.000000	1.000000	1.000000	6.000000
router	1.000000	0.875000	0.933333	16.000000
satellite dish device	1.000000	1.000000	1.000000	6.000000
server rack	0.666667	1.000000	0.800000	6.000000
accuracy	0.958333	0.958333	0.958333	0.958333
macro avg	0.958333	0.969318	0.959462	96.000000
weighted avg	0.969618	0.958333	0.960494	96.000000

Fig 4: Classification Report

The Classification Report for the Gadget Detection Model provides a comprehensive per-class and aggregate evaluation of the model's performance across **10 distinct gadget categories** on a validation set of **96 total samples**.

The Gadget Detection Model was evaluated across 10 gadget categories on 96 validation samples. The results reveal strong and consistent classification performance throughout. Seven out of ten classes namely **Computer Mouse, Keyboard I, Magnifying Glass, Phone, Satellite Dish Device, Keys Objects, and USB Stick** achieved F1-Scores of **0.952 or above**, with Computer Mouse, Keyboard I, Magnifying Glass, Phone, and Satellite Dish Device attaining **perfect scores of 1.000** across all three metrics. Notably, Phone detection achieved flawless results, which is particularly significant given that mobile phones represent the most prevalent cheating tool in online examinations [13].

Minor misclassifications were observed in **USB Stick** (Recall: 0.909) and **Laptop** (Recall: 0.909), each with one missed instance, likely attributable to visual ambiguity or unfavorable image conditions. **Router**, despite being the largest class with 16 samples, maintained perfect precision while recording a recall of 0.875, with 2 instances misclassified due to the high visual variability in router designs. The only notably weaker class was **Server Rack** (F1: 0.800, Precision: 0.667), whose reduced precision stems from a small support size of 6 samples and the inherent visual complexity of server hardware.

Overall, the model achieves a **weighted average F1-Score of 0.960 and an accuracy of 95.83%**, confirming its reliability and practical effectiveness for real-time unauthorized gadget detection in online examination environments.

The **Overall Accuracy** of **0.9583 (95.83%)** confirms that the model correctly classifies approximately 96 out of every 100 gadget samples, establishing it as a highly reliable detection system.

IV. Acknowledgment

We would like to express our sincere gratitude to our guide, *Prof. Sushant Shirbhate*, for his invaluable advice, perceptive ideas, and continuous support throughout the course of this project. His expertise, encouragement, and thoughtful suggestions have significantly contributed to the quality and success of this work.

.

V. References

[1] Aisyah, S., & Subekti, L. B. (2018, October). Development of continuous authentication system on android-based online exam application. In 2018 International Conference on Information Technology Systems and Innovation(ICITSI),(171-176).IEEE.

[2] Al-Hawari, F., Alshawabkeh, M., Althawbih, H., & Abu Nawas, O. (2019). Integrated and secure web-based examination management system. Computer Applications in Engineering Education,

[3] Andersen, K., Thorsteinsson, S. E., Thorbergsson, H., & Gudmundsson, K. S. (2020, April). Adapting engineering examinations from paper to online. In 2020 IEEE Global Engineering Education Conference (EDUCON), (pp. 1891-1895)

[4 Mohammed, Hussein M. & Qutaiba I. Ali. (2023). "Cheating Detection in E-exams System Using EEG Signals." International Conference on Scientific and InnovativeStudies, 1. No. 1.

[5] Palvia, S., Aeron, P., Gupta, P., Mahapatra, D., Parida, R., Rosner, R., & Sindhi, S. (2018). Online education: Worldwide status, challenges, trends and implications. Journal of Global Information Technology Management, 21(4), 233-241

[6] Xiang, L. (2022). Application of an improved TF-IDF method in literary text classification.

[7] Garg, M., & Goel, A. (2023). Preserving integrity in online assessment using feature engineering and machine learning. Expert Systems with Applications, 225, 120111.

[8] Muzaffar, A. W., Tahir, M., Anwar, M. W., Chaudry, Q., Mir, S. R., & Rasheed, Y. (2021). A systematic review of online exams solutions in e-learning: Techniques, tools and global adoption. IEEE Access, 9, 32689-32712.

[9] Wankhade, K. V., Gulame, M., Khune, P., Pimpalkar, A., Singha, S., & Kumbhar, M. (2024, March). A Meta-Learner-Integrated Stacking Voting Ensemble Network for Cervical Malignancy Classification. In 2024 International Conference on Emerging Smart Computing and Informatics (ESCI) (pp. 1-5). IEEE

[10] Palvia, S., Aeron, P., Gupta, P., Mahapatra, D., Parida, R., Rosner, R., & Sindhi, S. (2018). Online education: Worldwide status, challenges, trends and implications. Journal of Global Information Technology Management, 21(4), 233-241.

[11] Kasinathan, V., Yan, C. E., Mustapha, A., Hameed, V. A., Ching, T. H., & Thiruchelvam, V. (2022). ProctorEx: An Automated Online Exam Proctoring System. Mathematical Statistician and Engineering Applications, 71(3s2), 876-889.

[12] Ghizlane, M., Hicham, B., & Reda, F. H. (2019, December). A new model of automatic and continuous online exam monitoring. In 2019 International Conference on Systems of Collaboration Big Data, Internet of Things and Security (SysCoBIoTS), (pp. 1-5). IEEE.

[13] Atoum, Y., Chen, L., Liu, A. X., Hsu, S. D., & Liu, X. (2017). Automated online exam proctoring. IEEE Transactions on Multimedia, 19(7), 1609-1624.

[14] Pimpalkar, A. P., Thorat, N. N., Gulame, M. B., Lokare, D. A., Kulal, N., & Khune, P. (2024, March). Partitions of Liver Cancer by Enhanced Feature Extraction and Mapping with Improved Transfer Developing Methods.

[15] Parkhi, O.M., Vedaldi, A., Zisserman, A.: Deep face recognition. In: British Machine Vision Conference (2015).