AUTOMATED ONLINE REMOTE PROCTORING SYSTEM

1Anagha, 2Hemanth U, 3Sheetal C S, 4Dr. Shobha T

1,2,3,4BMS College of Engineering
Bengaluru, India

Abstract: E-learning and online-courses have gained popularity over the last decade due to its adaptability, accessibility, and user friendliness. But preserving the integrity of the examination is challenging. In this study, we describe a technique for avoiding a proctor's physical presence throughout the exam by developing a thorough multi-modal system. We have utilized hardware, such as webcam, video and audio in addition to actively capturing windows. This combination serves as the input to a smart rule-based inference engine, which can determine whether malpractice has occurred. We specifically detect head pose, eyeball and mobile phone and person detection as a vision based technique. The face of the examinee is recognized, and feature points are extracted from it to estimate a head pose. On the basis of changes in the face angle, the existence of audio, and active window capture, misconduct is discovered. The results of the experiment demonstrated that our system outperformed the existing systems.

Keywords: Online Mode, Automated Online Exam Proctoring, User Verification, Speech Detection, Vision based technique

I. INTRODUCTION

Both traditional universities and organizations specializing in distant learning have seen enormous development in e-learning over the past ten years[1][11]. Distance learning emerged as a promising methodology and remained the preferred form of education post coronavirus [8][4][15]. As a result, several colleges and universities have chosen to give their exams online[6]. It enables institutions to deliver education at a high standard while saving time and money[15]. As the demand for online education and remote examinations continues to grow, maintaining the integrity of assessments becomes a significant concern. Most of these methods are either overly complicated or ineffective.[5] Additionally, other studies have shown that online tests are more susceptible to various forms of cheating, which might diminish their credibility[11][13][14]. Cheating online is frequently unnoticed and challenging to prove[5]. Due to the lack of in-person interaction, proctoring online tests can be difficult[13][5].

To perform automatic online exam proctoring (OEP) without compromising the reliability which is challenging, we introduce a multimodal system in this research. Proposed OEP includes different models to track students’ cheating behavior while they are taking the exam. The system will have four vision based abilities and they are combined using multithreading so that they can work simultaneously. Four vision based powers are Gaze Tracking, Head Pose Estimation, Person Counting, Mobile Phone Detection. In addition, Active Window Monitoring has been included.

The rest of the paper is organized as follows: Section II presents key concepts related to automated online exam proctoring. Section III describes the proposed solution approach to OEP. Section IV will discuss the experimental results, while Section V presents the concluding state and pinpoint directions for future work.

II. RELATED WORK

For monitoring the online exams, a number of commercially accessible online proctoring technologies are available, including ProctorU[1][2][4][3][13][15], Kryterion[3][11], Mettl[4], Proctorio[5][13], ExamSoft[5], and others. But they continue to rely on someone "watching" the exam-taking. For instance, Kryterion uses a human proctor to monitor a test-taker via a webcam from a distant place. The number of students that a remote invigilator can manage at once in the event of live remote invigilation is obviously constrained. In some technologies, manual post-examination evaluation has been adopted, which is tedious and cumbersome due to the volume of recorded video materials[1].

Basically, online proctoring solutions come in a variety of forms, including controlled, recorded, and automated real-time proctoring[6][5]. When compared to using human proctors, automated online real-time proctoring is far more scalable and affordable[5] because it can operate without human intervention. An automated online remote proctoring system leverages advanced technologies such as artificial intelligence (AI), computer vision, and machine learning to monitor and assess test-takers during online examinations. By employing a combination of audiovisual recording, facial recognition, and behavior analysis, these systems aim to detect and deter cheating behaviors, ensuring the authenticity and credibility of the assessment process.

The system proposed here makes use of an internet connection and a single webcam with a microphone. With these minimal specifications, our primary goal is to create a system that is completely automated, multimodal(i.e including audio, video, and active window information) and not utilizing any unpleasant additional hardware[3]. The proposed multimodal system's ultimate objective is to protect the academic integrity of exams by offering real-time proctoring to identify the majority of test-taker cheating behaviors[11]. Also, our project's primary goal is to make the existing online assessment system more accurate and ensure that its results are more reliable[14].
III. SOLUTION APPROACH

We have developed a system in which the candidate's actions, such as face movements, are monitored by a monitoring system, such as a webcam, together with system utilization and audio data. This information is used to feed a rule-based inference system that was developed to identify the potential for malicious activities. Preparation and the exam are the two stages of our suggested online exam proctoring approach. The exam phase includes active window monitoring, vision-based strategies, and audio-based procedures, while the preparation phase encompasses candidate authentication. The architecture of the above-mentioned proposed system is shown in Fig. 1.1.

![System architecture](image)

The main modules, which are gaze tracking, head pose estimation, mobile phone detection, multiple person detection, browser monitoring and audio monitoring. The users are the students and the proctors.

The students are able to register, login and logout, join and take exams, and check exam history, check results and report problems. Meanwhile, proctors can create questions and answers, including objective and subjective type questions and answers. Proctors can view exam histories, share exam information to students, and read questions as well as change and delete them. However, between and after the exam, these functions are not functional. Proctors may view and publish the results. The professor has access to the proctoring logs of the students, live exam monitoring, and problem reporting.

The following exam features are some of those that our suggested system supports. The types of exams that our OEP system supports include objective, subjective, and practical. The timer won't update if the web page refreshes, support for the use of a calculator during exams of the mathematical kind, support for the use of negative marking, all question statistics will be displayed to the student at the moment of exam submission for confirmation.

The proposed system is compatible with the proctoring functions listed below, which include making logs about window events each time a user opens a new tab or changes the active tab, and recording the students' audio frequency every five seconds, detection of a mobile phone, multiple people participating in the exam, taking student picture logs every five seconds, estimating the gaze. In the meantime, the professor can look through all of the exam-takers' logs and govern whether or not a student has cheated. In addition to these features, a sole feature has been included that enables the system to repeat a student's voice if he talks during the exam in order to let him know that he is being proctored the entire time.

TECHNIQUES USED FOR OEP

A. STUDENT VERIFICATION

The test taker must authenticate himself throughout the preparation phase by using a password and facial authentication before starting the exam. Face recognition technology will be used to validate the user's image every time they log in and throughout an exam. The system only permits one login per user, preventing unfair behavior.
B. VISION BASED TECHNIQUE

We will try to keep track of the test-taker's eyes to see if he is glancing up, left, or right, which he might do to signal or glance at a notebook. The face keypoint detector from Dlib and additional image processing from OpenCV can be used for this. We used face keypoint recognition in which it detects 68 facial keypoint. We need this to determine the position of eyes which is shown in Fig 1.2 shape_predictor_68_face_landmarks.dat.bz2 Pretrained model is used to determine the face keypoints. A rectangle object of the dlib module that contains just a face's coordinates is what the facial keypoint detector uses as input. The frontal face detector built into dlib can be used to find faces. Any classifier will work for this purpose. If we want accuracy and speed, dlib is the better choice.

Similarly we are able to build a head pose estimation model to track the test-taker. There are many phases needed to perform head pose estimation that make it a difficult problem in computer vision. We must first identify the face in the frame before identifying the numerous facial landmarks. The nose tip, the chin, the outermost points of the lips on the left and right, and the outermost points of the eyes on the left and right, total six points on the face. Using the usual 3D coordinates of these facial landmarks, we attempt to estimate the rotational and translational vectors at the tip of the nose. We need camera intrinsic parameters, such as the focal length, optical center, and radial distortion parameters, in order to make an accurate assessment. Even with the image displayed, we were able to test it on an i5 processor and obtain a respectable 6.76 frames per second.

We have also used the YOLO v3 model to count the number of people and mobile phone detection in our OEP. We need wget-python (to download YOLOv3 weights), OpenCV, and Tensorflow. Additionally, you can manually download them. We can also use the Faster RCNN method or SSD method for better accuracy but we are training our model in CPU so we used YOLO v3. The YOLOv3 pre-trained model, which is incredibly rapid and nearly as accurate as SSD, can classify 80 items. It consists of 53 convolutional layers, followed by batch normalization and leaky RELU activation layers. They have used a stride of 2 in convolutional layers to downsample rather than pooling. If you're using OpenCV, don't forget to convert the RGB photos because it requires them in the input format. The image sizes might range from 320x320 to 416x416 to 608x608 and the input type is float32.

C. AUDIO BASED TECHNIQUE

The idea is to use Google's speech recognition API to record audio from the microphone and convert it to text. The audio is captured in chunks so that there is no mandatory space requirement when using this approach because the API requires a continuous voice from the microphone, which is unfeasible. To ensure uninterrupted recording, a distinct thread is utilized to access the API. The API processes the most recent recording, adds its contents to a text file, and then deletes it to conserve space. To stop cheating, this technique can be used in conjunction with a secure browser. As a proctor is necessary to carry out specific tasks, this initiative does not completely eliminate the necessity for one. Also, this feature is still being developed, not yet utilized completely in our system.
D. ACTIVE WINDOW CAPTURE

We have attempted to use active window monitoring in our system in addition to the candidate verification, vision-based, and audio-based methodologies. As soon as the user logs in and begins the exam, the system automatically records audio every five seconds and records window events whenever the user changes tabs or opens a new tab. Additionally, every 5 seconds, student image logs are taken. The functions to cut, copy, paste, and take screenshots are also blocked.

IV. RESULTS AND DISCUSSIONS

From this study we came to know that Dlib is a C++ toolbox with ML techniques that can be used to address practical issues. Despite being built in C++, it can be executed in Python thanks to python bindings. It also features the fantastic facial landmark keypoint detector that we used to create a real-time gaze tracking system. The Histogram of Oriented Gradients (HOG)-extracted features used in the frontal face detector offered by dlib are then subjected to an SVM. The distribution of gradient directional information is employed as a feature in the HOG feature descriptor. Additionally, Dlib offers a more sophisticated CNN-based face detector, however it is not real-time on a CPU. Although the DNN module of OpenCV’s face detection model performs well, it can have issues with really large images.

If you're working with little photographs, make sure to upscale them because Dlib cannot recognise faces less than 80x80. However, this can lengthen the processing time. Therefore, taking into account both of the above, MTCNN would be the greatest option if we were dealing with extreme face sizes and is now in the lead among competitors. Similarly for eye detection we also used Dlib because it gives predictions in real time. If speed is not the issue then CNN is a good option, also CNN gives good predictions for non-frontal faces. Also for number of person and mobile phone detection is typically known as object detection we used YOLO v3, we could also have used Faster RCNN or SSD but we are building our proctoring system using CPU so we have chosen YOLO v3 but YOLO is also nearly as accurate as SSD. We used Anchor Boxes, Non-Maximal Suppression that uses IOU for better predictions.

Also we worked on audio as well, in the future implementation we focus more on audio side for now what we implemented is while writing the exam audio system works as mic and it will echo every sound while writing exam so that time student get cautious about his activities. In Fig 1.3 and Fig 1.4 attached a few screenshots of our whole experimental results.

V. CONCLUSION

The intended objective of this project is to create a comprehensive multi-modal system that will enable the instructor to monitor the students taking an online examination. The proposed system could provide improved convenience and accessibility, reduced costs when compared to traditional in-person proctoring and existing online proctoring systems. Out of the several system attributes that have been proposed, we have created the abilities to do feature point extraction, gaze tracking, mouth opening detection, head position estimation, person count detection, mobile phone detection and active window monitoring. While these form the key components, the ultimate objective is to develop the full fledged proctoring system. Advanced design choices have already made for this system, which is still being modified. To maintain the effectiveness and fairness of the system, it is crucial to take into account and deal with the potential problems such as technological issues, privacy issues, and algorithmic biases.

VI. FUTURE WORK

In the future, we intend to apply and research a variety of human behaviors, like staring out the window, talking to people, concentrating on other directions, moving about, and so forth. We'll concentrate on these strategies in the upcoming study and compare them with the system we've already proposed. Future work will also focus on improving detection algorithms for detecting complex cheating techniques, improving accessibility features for students with disabilities, addressing privacy concerns through data minimization and secure protocols, integrating seamlessly with learning management systems, and providing thorough training and support for instructors to ensure the systems' effective implementation and use, all while incorporating advancements in AI and behavior analysis.
REFERENCES


