International Journal of Mathematical, Engineering and Management Sciences
Vol. 11, No. 1, 196-241, 2026
https://doi.org/10.33889/IJMEMS.2026.11.1.010

Toward Sustainable Online Education: AI-Powered Hybrid Proctoring with LSTM and CNN-Based Anomaly Detection to Enhance Academic Integrity

Manit Malhotra
Department of Computer Science & Applications,
Panjab University, Chandigarh, India.
Corresponding author: manitmalhotra@rediffmail.com

Indu Chhabra
Department of Computer Science & Applications,
Panjab University, Chandigarh, India.
E-mail: indu_c@pu.ac.in

(Received on July 14, 2025; Revised on October 3, 2025; Accepted on November 19, 2025)

Abstract
This 21st century is the era of e-education, and there is an urgent need to adapt to the modern needs of education in line with the 4th Sustainable Development Goal (SDG). The goal of this study is to align with SDG 4's goal of providing inclusive and equitable quality education. It aims to discuss the emergence of proctoring software that utilises artificial intelligence as a means of addressing the increasing cases of cheating in remote learning environments and online assessments, thereby reducing the need for physical infrastructure and travel, and contributing to sustainability. This study proposes a hybrid proctoring system based on two different folds: the first fold detects the significant improvement in the candidate's marks from the custom dataset of 350 students to identify the suspected candidates using Long-Short Term Memory (LSTM), and the second fold analyses the exam video recording of suspected candidates frame by frame to perform the behaviour analysis to detect the anomalies with the help of Convolutional Neural Network (CNN). In this paper, various anomalies were identified, including off-screen gazes, the use of cell phones and earphones, and talking. The proposed system obtains an accuracy of about 87.8% as well as exhibits resource-efficient performance with respect to processing time, CPU usage, along with memory usage.

Keywords- Academic dishonesty, Hybrid proctoring system, Behaviour analysis, Long Short-Term Memory (LSTM), Convolution Neural Network (CNN).

1. Introduction

The COVID-19 pandemic is an example of the phenomenon, which has necessitated an already increased transfer to an online education system, which proves the usefulness and convenience of digital learning aids. This has forced academic institutions to be concerned about the issue of academic dishonesty because of this swift transition to distance learning (Karthika et al., 2019; Nurpeisova et al., 2023). Anonymous and unsupervised online areas have inadvertently served in enhancing the level of academic dishonesty. It is the objection of SDG 4 to offer decent education that is fair and credible because this cannot be compromised. This includes a wide range of unethical behaviors including plagiarism, cheating, impersonation, and utilization of unauthorised aids throughout an exam (Adoga, 2023). The use of evolutionary proctoring software has been used to tackle these emerging dangers and threats on academic integrity in online learning, which provides a secure and honest evaluation. Digital divide in terms of resources is also a major challenge since it brings inequalities through students in rural and urban settings that must be bridged to curb the gap (SDG 10).

Proctoring software that uses AI is to provide security during the exams. When a student logs in into this system, a camera at the rear of the computer records the activities of the user. The AI technology allows capturing the entire head, movements, and eyes and hands. As an example, it can see the usage of books or

mobile phones by ensuring that the fingers are not close to the keyboard and nobody is present in the room. It oversees all activities through the camera also known as the "third eye" of the application. Alerts are brought to high levels whenever a student turns or moves or steps out of a particular position. When two warnings to a student are made, the student is subject to being barred in the examination (Aurelia et al., 2024).

1.1 Academic Misconduct and the Online Environment

An overview of some of the recent research results shows that the problem of academic dishonesty in higher education is still acute (Al-Airaji et al., 2022). Examples of breaches in an online learning ecosystem are more and of a greater nature than they are in a physical ecosystem. Though online examinations provide students with more privacy, the risk of cheating will always be more significant since in contrast to the conventional approach of one proctoring students immediately, in remote environments, the application of common cheating-detection strategies is hard to implement (Imah et al., 2023). The pandemic has significantly influenced the education sector, overcome the barriers of digital proctoring systems in educational establishments by conducting organised and fair virtual exams. In this respect, it is also found that biometric monitoring and such methods as facial recognition can be promising in terms of keeping the level of academic integrity in the online learning setting (Aurelia et al., 2024). In this paper (Mohammadkarimi, 2023), the take of EFL teachers on the usage of AI to derail the learning process were considered and the paper shows some positive concerns as well as negative ones. The author also highlighted the need to have proper training of teachers and code of conduct as well as the use of AI to maintain integrity in education.

1.2 Importance of Quality Proctoring Systems

The legitimacy of the academic qualifications will be based on the quality and honesty of examinations taken (Wan et al., 2021). Automated proctoring systems also assist in promoting social responsibility in education towards the aim to make the society fairer, more ethical (SDG 16) by making assessments fair, honest, and accountable. Conventional proctoring mechanisms that are based on physical containment have been flawed in nature and might not be adequate in online learning. This is the reason why there is an urgent need to integrate advanced digital proctoring systems in the online learning sphere to maximise the ability of tracking and reducing incidences of cheating (Atoum et al., 2017).

The digital proctoring platform consists of the associations of various technologies, which includes video monitoring, browser locking, plagiarism detectors, and ML-based software to detect behaviour (Alguacil et al., 2024; Arianti et al., 2023; Ngo et al., 2024). These technologies are designed to work in much the same way a real invigilator would, supervising students during an exam and ensuring they meet the criteria outlined in the testing guidelines. In practice, the efficiency of the above solutions is questionable because they must overcome problems related to privacy, usability, and reliability. Moreover, fraudulent acts are constantly evolving, and countermeasures against them are also constantly improving with the technological advances in monitoring. This back and forth nature of this game is significant to the fact that a proctoring system containing integrated forms of detection needs to be in place to work together to prevent the different types of academic cheating. Through the recent monitoring and surveillance solutions, not only can scholastic dishonesty be stopped but also the privacy of information of students and transparency in the educational system can be promoted (Noorbehbahani et al., 2022). The available digital proctoring technologies have potential in reducing the problem of academic integrity but, their application has not yet overcome the natural security, privacy, and effectiveness concerns. The recent developments in the AI-based proctoring systems (AIPS) that is meant to deliver more sophisticated detection mechanisms introduce new ethical, technological, and multiple trust-based issues. These technologies and their potential and restrictions concerning the online education are identified with the help of a systematic review of the

recent research that identifies a need to balance the implementation of these technologies to advance, but not disrupt, the online education (Nigam et al., 2021).

1.3 Motivation and Contributions

(i) Due to the appearance of new educational systems, where more and more educational establishments contain online courses of study, it was only natural to vigorously enforce online tests. The newer mode of providing education in online mode has its benefits of high flexibility and openness. On the negative side, it has introduced new methods of cheating, which is degrading the trust aspect which is the pillar of every education system in the world. This does not only make education underrated but also makes the democratic worth of education irrelevant to all learners.

(ii) Our desire to work is based on the need to reduce such challenges by designing and piloting an automated proctoring system. It is basic research undertaken to try to save academic destruction in the wake of the new millennium technology and it is hoped to come up with a thorough system through which cheating can be detected and prevented early. A further advanced role of a such system besides curbing an unethical conduct is that the evaluations actually present the actual performance of a student and thereby uphold the integrity of certifications acquired. A key emphasis on environmental sustainability (the decrease in the expense of traveling), social sustainability (equity and equity of the remote setting), and economic sustainability (expenses involved in contemporary exam arrangements) was maintained.

(iii) Besides, we align our proposed framework with the current research of the state-of-the-art methods to expand the existing knowledge of the optimal practices in the field of online proctoring. This comparison is significant because it points out the worst and the best characteristics of the systems available, thus it assists in coming up with systems that are smarter, fair, and less intrusive to the privacy of students.

(iv) This piece of work touches upon one of the crucial issues of the modern learning landscape in terms of equity and validity of online tests facilitating inclusivity and equity in learning the cornerstone of SDG 4. By building and thoroughly testing a new automated proctoring system, we are planning to help decrease the possibility of cheating, increase credibility, and increase the quality of online education. Additionally, inclusive populace can also benefit through the work to have a fair chance of online education through the reduction of the infrastructure constraints. The following are the major contributions of this research.

- The paper presents a systematic literature review of proctoring systems to cast light on the models implemented, the issues tackled, and the reliability of outcomes. Through a thorough comparative analysis, it reveals existing areas of research limitations and aims to provide significant recommendations that will aid future development in the area of online academic integrity.
- This study proposes a hybrid proctoring system capable of detecting dishonesty in online exams.
- This proctoring system works with two different folds: the first fold detects the significant improvement in the candidate's marks from the custom dataset of 350 students to identify the suspected candidates using Long-Short Term Memory (LSTM), and the second fold, we developed a dataset of video recording in higher resolution for precise details, the proposed system analyses the exam video recording of suspected candidates frame by frame to perform the behaviour analysis to detect the anomalies with the help of Convolutional Neural Network (CNN).
- The paper also identified various anomalies, such as off-screen gazes, cell phones, earphones, talking, etc.

- This system also demonstrates better performance in object detection and facial landmark identification, specifically obtaining 87.8% object detection accuracy, when compared to prior models.

The rest of the paper is organised into suitable sections. Section 2 presents a literature review that navigates the maze of previous studies, identifying the tools and strategies employed to combat academic dishonesty and highlighting the gaps in current methodologies. Section 3 will give methodology where a detailed explanation of how the automated proctoring system would be designed and developed will be given. Section 4 can offer the implementation with the detailed description of the proposed system development, the challenges on how various components can be incorporated into our system, and the ways that our idea can become a reality. In Section 5, the results of the suggested methodology are given, exposing and comparing the results of the system through state-of-the-art techniques. The discussion on the results and their implications in Section 6 gives a general understanding of the world-wide implications of the findings on the educational world. It explains how the given system might redefine the method of safeguarding the academic integrity and how the research might proceed in the future. Lastly, the paper is concluded in Section 7 with the aim of encapsulating the paper to bring out the key issues and contributions to the field with the aim of ensuring a secure and equitable online learning environment. The Industry, Innovation, and Infrastructure (SDG 9) horizons that still have to be pursued are identified in the future work, and the way of further researches is shown to establish a solid base of the sphere.

2. Literature Review

In their research, Putra et al. suggested an enhancement of online exam integrity by developing a proctoring system using object detection. The implemented application was developed using the R&D (Research and Development) method and the Limited Application Development (RAD) methodology. The fame of the model was 73.1% in deterring test cheating. Author strengths and limitations were also noted, such as a high falses positive rate and dependency on the environment (Arianti et al., 2023).

The study by Ngo et al. (2024) presented an exam proctoring system, since it aims to identify abnormal behaviours in an online exam in real-time with the help of machine learning (ML) methods. In the real case, the accuracy in the performance of this system was high (78.5% at 27 FPS). The key points to note were: real time monitoring, alerts in case of abnormalities and artificial intelligence-based proctor decision-making process. The same case is with Alguacil et al. (2024), who treated the opportunities to check cheating in an online test by observation of the user behaviour and exclusion of cheating behaviour. The author asserted that there should be strong systems of monitoring in order to support academic integrity. Noorbehbahani et al. (2022) in the presented systematic review examined the approaches of detecting cheating online, outlined the most frequent cheating behaviours, and evaluated the strategies of detecting the cheat, once again, underlining the necessity to come up with highly efficient proctoring schemes that would be used to administer justice to cheaters during assessment.

Holden (2021) In his study, he uncovered the problems in realising academic integrity and presented a few primitive solutions by considering online proctoring. This study revealed the effectiveness of various proctoring methods and the impact of clear definitions of academic misconduct in reducing instances of cheating. Moving forward, Ege and Ceyhan (2023) presented an object detection and face recognition-oriented online examination proctoring system developed on the client's side using human voice detection. It works on the user's device, ensuring that safety is maintained while having higher accuracy in detecting various cases of cheating cost-effectively.

Ahmad et al. (2021) formulated a deep learning-based online proctoring system for facial recognition. Their system monitors students to identify any unfair, unethical, and illegal behavior during classes and exams. They employed biometric approaches, including facial recognition using the histogram of gradients (HOG) face detector and the OpenCV facial recognition algorithm, achieving accuracies of 97.21% and 99.3% for face detection and face recognition, respectively. Nevertheless, they have only used the YOLOv3 detection model in their system, which still provides sufficient opportunities to conduct further research.

Dadak et al. (2022) developed a real-time cheating detector concerning online tests through the implementation of different facial recognition systems, which also resulted in a cheat detection system. The solution was designed to operate in a web client, which makes it a viable and scalable proctoring solution. The other article written by Bilen and Matros (2021) tried to address the problem of cheating in the online settings, in specific regard to the rising trend of cheating during the COVID-19 crisis. Examining the rates of cheating and offering remedies, the work leveraged the data of online chess communities and online examinations in case of the COVID-19 lockdowns. The method utilised timestamps from students' Access Logs that tracked the behaviour of students on online tests.

AI-based Large Language Models (LLMs) are becoming popular cheating aids in online examinations. Surahman and Wang (2022) investigated the emergence of issues that have been brought by LLMs, particularly, ChatGPT, in upholding academic integrity to detect AI-generated texts which will be used to cheat. These authors revealed the drawbacks of the modern anti-plagiarism technologies, and they suggested that new policies, AI awareness, and better tools have to be provided. In a similar fashion, Rane et al. (2024) addressed the question of fairness and honesty that the LLMs, specifically ChatGPT, bring to the concerns within educational institutions of higher learning. The paper explained the methods of how latent plagiarism may be identified using traditional plagiarism detection software to avert the immoral applications of AI. The paper implied that institutions had to change their honor codes, encourage the use of AI, and start with implementing new technologies to further improve stereoscopic checks and detracting dishonest practices in education.

2.1 Comparative Analysis

After a review on many studies, it became very clear the extent to which authors have already advanced in this field and the things to be discovered. A number of major insights were made on the basis of this comparative analysis. For instance, we identified the model the authors primarily used to create the proctoring system, the issues they addressed in their study, and the accuracy of their conclusions. We have conducted a comparative analysis of all these studies, as presented in **Table 1** below. Following a thorough analysis of each study, we have also highlighted any identified limitations or research gaps.

Table 1. Comparative analysis of recent studies.

Study	Objective	Work done	Methods used	Results achieved	Limitations/Research gaps
Arianti et al. (2023)	Develop an online exam system with object detection	Developed an object detection-based exam system	Research and Development (R&D), RAD model	73.1% effectiveness in reducing cheating	High false positive rate, environment-dependent
Ngo et al. (2024)	Detect abnormal behaviour in online exams	Developed a real-time monitoring system	Automated behaviour detection, Mediapipe	78.5% accuracy, 27 FPS processing speed	High false positive rate
Alguacil et al. (2024)	Evaluate academic dishonesty and monitoring practices	Compared academic performance under different proctoring methods	Empirical study, data analysis	Identified the effectiveness and cost of proctoring methods	Limited geographical scope, user perception focus

Table 1 continued…

Noorbehbahani et al. (2022)	Review research on online cheating and prevention techniques	Reviewed various online cheating detection methods	Systematic review	Highlighted common cheating behaviours and detection methods	Diverse cheating contexts, evolving cheating methods
Holden (2021)	Define academic dishonesty and its impact on online assessments	Discussed integrity challenges and solutions in online assessments	Conceptual analysis, literature review	Identified key integrity challenges and potential solutions	Limited empirical data, theoretical focus
Ege and Ceyhan (2023)	Detect cheating in online exams	Proposed end-to-end client-based system	Object detection, face recognition, voice detection	Effective cheating detection reduced server costs	Performance depends on the user's computer
Ahmad et al. (2021)	Detect cheating using Face Recognition, Eye Blinking, and Object Detection	Developed a video monitoring software system	Video monitoring, automatic alerts	97.21% accuracy in detecting cheating	Lack of real-life deployment with a large number of users.
Dadak et al. (2022)	Develop a real-time cheating detection system	Developed a real-time detection system	Deep learning, object detection, voice detection	High accuracy in detecting various cheating behaviours	Requires high computational resources
Bilen and Matros (2021)	Discuss the issue of internet cheating, paying particular attention to its growth since the COVID-19 pandemic began.	Investigating the frequency of cheating and offering remedies.	Examined instances of test cheating using timestamps extracted from Access Logs.	During the COVID-19 lockdowns, it was discovered that online exam cheating was commonplace.	Inconclusive evidence of cheating and challenges in executing specific solutions, such as the installation of required cameras.
Surahman and Wang (2022)	To understand the impact of AI-generated content on concerns of academic integrity and to discuss and suggest policies for addressing its misuse for academic purposes.	Summarised prior work to understand AI's involvement in academic dishonesty and how AI-based proctoring systems can alleviate dishonesty.	Systematic review	Stated current issues with currently available software for the identification of plagiarism, came up with recommendations for the proactive approach for educational institutions, and pointed out the need to accustom oneself to AI tools as well as update the current Honour Code.	Ethical and Policy Challenges
Rane et al. (2024)	Examine the impact that AI-generated post-variegated text has on the concept of scholarly integrity, to review and discuss the efficiency of current plagiarism detection tools, and to explore the measures institutions could undertake to address these challenges.	Presents literature on academic dishonesty, discusses AI-based tools to check plagiarism, and explains their ethical implications.	Systematic review	Established that previous plagiarism-detection techniques cannot identify plagiarized work generated by AI. Emphasised the modern requirement of educational institutions to update and redefine Honour codes, raise awareness of Artificial Intelligence, and apply novel detection technologies.	Ineffectiveness of Traditional Detection Tools, and Ethical and Policy Uncertainty

2.2 Comparative Positioning of Contributions

Recent literature explored as part of this research offers several valuable AI-enabled proctoring systems; however, a few methodological gaps persist. For example, models proposed by Arianti et al. (2023) and Ngo et al. (2024) exhibited moderate accuracy but had high false positive errors and environmental dependence, which restricts their validity across a wide range of testing conditions. Similarly, computationally expensive methods by Dadak et al. (2022) and Ege and Ceyhan (2023) depended on the performance of the device, which may pose a problem with scalability. On the other hand, the given hybrid algorithm employs the two-step pipeline of detection (LSTM + CNN) which first sifts the suspicious cases according to the performance abnormalities and then selectively analyses the videos (using resources) to reduce the amount of

computational time and false alarms. Alternatively, the works by Holden et al. (2021) and Alguacil et al. (2024) were theoretical, geologically narrowed or those resting on impressions rather than an experimental finding about anomaly detection. The contribution that we make varies by building and testing our own dataset of 350 students and high-resolution exam records up giving us a good empowering empirical foundation to test.

Certainly, it is obvious that the ethical and policy issues raised by Surahman and Wang (2022) and Rane et al. (2024) are important but poorly investigated in systems design. We refer to this dimension in our work by expressly centering our attention on student privacy, equity, and equal access, which is in line with SDG 4 (quality education). Lastly, although the models recommended by Ahmad et al. (2021) achieved the best accuracy (97.21%) among all; however, they did not explore specificity, the issue of false positives, and the diverse applicability of their proposals. Our hybrid method combines temporal performance with behavioral video analysis, offering not only high detection rates but also contextual interpretation (off-screen gazes, phone use, earphones, talking, head movement) of anomalies, making it more accurate and interpretable.

Based on the critical analysis of the findings of state-of-the-art literature, this study intends to cover the methodological shortcomings of the previous AI-based proctoring systems. Studies including Arianti et al. (2023) and Ngo et al. (2024) had high false positive rates and relied on particular testing conditions, whereas other researches, for instance, Alguacil et al. (2024) and Noorbehbahani et al. (2022) did not provide generalizability, specificity and bias exploration. On the other hand, the proposed hybrid scheme integrates performance-based anomaly detection (LSTM), and behavioural video-analysis (CNN), to reduce error of misclassification while adapting to various online contexts. Additionally, the two-fold system design guarantees the computational effectiveness, scalability to MOOCs, ethical transparency – filling critical gaps of fairness, privacy, and sustainability that were mostly ignored in earlier studies.

2.3 Datasets Used

Table 2 below provides a brief description of the datasets used by researchers in previous studies.

Table 2. Dataset used in previous studies.

Study	Dataset description
Arianti et al. (2023)	Data derived from interviews at SMK Pasudan 1, Cimahi.
Ngo et al. (2024)	Google search trends data of 2020 Advanced Placement (AP) exams.
Alguacil et al. (2024)	Custom dataset of student volunteers with facial landmarks, analyzed using the MediaPipe library.
Noorbehbahani et al. (2022)	Custom dataset of video recording during the exam.
Holden (2021)	No dataset is used as the primary focus of the study was to develop a RAD system.
Ege and Ceyhan (2023)	Custom dataset of photographs and videos with different angles and accessories of 100 students of the university.
Ahmad et al. (2021)	FDDB (Face Detection Data Set and Benchmark) and LFW (Labeled Faces in the Wild) datasets.
Dadak et al. (2022)	Custom dataset containing 1 audio and 2 videos of each subject, and collected for 24 different subjects.
Bilen and Matros (2021)	COCO dataset for phone detection, Crime Investigation and Prevention Lab (CIPL), custom dataset of 200 videos, and CASIA-Web Face.

2.4 Limitations of Existing Systems

The growing field of online education necessitates the effective management of academic honesty, which entails identifying innovative methods to track and prevent cheating. These systems are also important improvements; however, they also have some drawbacks, and some spheres that should be investigated. In short, the given section discusses complicated issues and problems that have to be resolved in combating academic dishonesty.

2.4.1 Specific Object Detection and Evasion Techniques

A significant issue current with online proctoring is that, it is not always evident what exactly students are attempting to conceal. As a rule, the current systems rely on the artificial algorithms of detecting potential cheating, including irrational movement patterns or the identification of the predefined objects that are not supposed to be there (Beck, 2014; Allen and Seaman, 2015). Such parameters can never entirely satisfy the creativity of the strategies hence igniting a never-ending battle amid system developers and users (Kulkarni et al., 2011). This gap marks the necessity of the increasingly flexible and adaptable approaches that could be used to combat the new types of cheating.

2.4.2 Empirical Efficacy Versus User Perception

Moreover, a significant void, in this case, is that certain studies just have an inclination to study how the user perceives being a cheater in her work without actually performing the functionality of showing the real world, which in this case is the discovery of serial cheaters (Berkey and Halfond, 2015; Corrigan-Gibbs et al., 2015). Systems should be rigorously tested in an extensive variety of real-life situations to make sure that they can be used to identify dishonest behaviour, although user-feedback goes a long way in enhancing the user experience and the user interface design.

2.4.3 Implementation, Training, and Adaptation Challenges

The questions of using cheating detection systems in real world are not an easy matter to implement because the individuals would be required to train and adapt to the new forms of cheating (Grijalva et al., 2006). Most of the current models need big datasets to train which at times cannot be accessing or may not cover all forms of cheating (Guo et al., 2008; Cluskey Jr et al., 2011; King and Case, 2014). Another factor that makes the issue of cheating in modern contexts fast is that the methods of cheating keep evolving and hence the detection system should be regularly updated. This may be time consuming and resource consuming, finding it hard to match the emerging threats at a high rate.

2.4.4 Detection Specificity and False Positives

Specificity of methods of detection and problem of false positives need more research. Further on, to achieve the right accuracy, the systems must reduce the number of false positives, i.e. they must not label the innocent as the cheaters (Rosen and Carr, 2013). This proves difficult as detection techniques that are over-bearing may damage the students' trust and portray the system as less legitimate than it is.

2.4.5 Privacy Concerns and Practical Implementation

Previous research that is related to the system of designing the cheating detection model does not focus on more relevant questions, including the applicability of the research in a real-life scenario and the safety of student privacy (Etter et al., 2007; Anohina-Naumeca et al., 2020). The technologies of proctoring raise essential ethical concerns, mostly within the domains of monitoring and data handling (Park, 2017). A fine line exists between protecting privacy of the students and keeping a strong watch and enquiry on instances of cheating all within well spelt out guidelines.

2.4.6 Quantitative Outcomes and Generalizability

Finally, before implementing the systems into the real-life context, it is essential that the authors adequately state the quantitative results of their proposal. A limited number of studies discussed such measures in the literature, and other researchers are struggling to objectively measure the effectiveness of their systems, which defines the lack of clearly specified quantitative outcomes or measures (Chen et al., 2020).

3. Methodology

The design methodology of the proposed system, as shown in **Figure 1**, may be considered to consist of two independent layers, the first layer working with numerical data and the second one with the video data. It can be divided into the following sub-sections, which detail the entire system design:

3.1 System Design

The hybrid proctoring system is an integrated system that incorporates learned performance data and video tracking to comprehensively identify academic dishonesty. Data analysis and computer vision-based analysis is designed in such a way as to find the trend and behaviour hence deducing the chance of fraud. The sections ahead give a detailed insight into system design, and its elements and the logical working behind the system.

3.2 System Architecture

Consequently, we have developed a hybrid proctoring system, which takes advantage of being data-driven system with behavioral analysis thus is an excellent tool of uncovering cheating in educational institutions. Ultimately, there are two major modules that the system must fall under, which are:

3.2.1 Data Analysis Module (Fold-1)

In the given module, the trends of the scores summerised in exams of the students will be picked to detect any anomalies and these anomalies might result as a result of the students being dishonest with their reported scores. It works on the data and processes it to analyze it and then arranges it chronologically in the form of scores of different students through LSTM networks. It then executes a tracking and analysis algorithm which gives real time feedback of the performance of this analysis.

3.2.2 Video Analysis Module (Fold-2)

The module involves the implementation of the high-level computer vision techniques used to track the movements of students when taking tests. It is able to identify cheating eyes, faces and objects of the individuals and also study their behavior. The detection of the objects in particular was accomplished with a ready-made YOLOv3 model (Darknet-53 backbone) and the identification of facial features was done using a 68-point CNN-based face detector (ResNet architecture) that is given by the Dlib toolkit. The trial and error values were choosing the confidence level of the YOLO detection at 0.5 and non-max suppression IoU at 0.4. We also applied MTCNN to ensure that face detection would be made possible under varying conditions of lighting. These CNN models were starved with default weights with their thresholds heuristically adjusted on sample frames.

3.3 Data Analysis Module (Fold-1)

The workflow of the first module of the proposed system is as follows:

3.3.1 Dataset Description

A total of 350 students were characterized to make up a dataset of anomaly detection. The data has 8 variables that include ID = gender = 5 further exam scores, stature (0/1). In this case, 1 would mean a pass and 0 would mean a failure denoting the status or performance of a student. **Table 3** below explains each attribute.

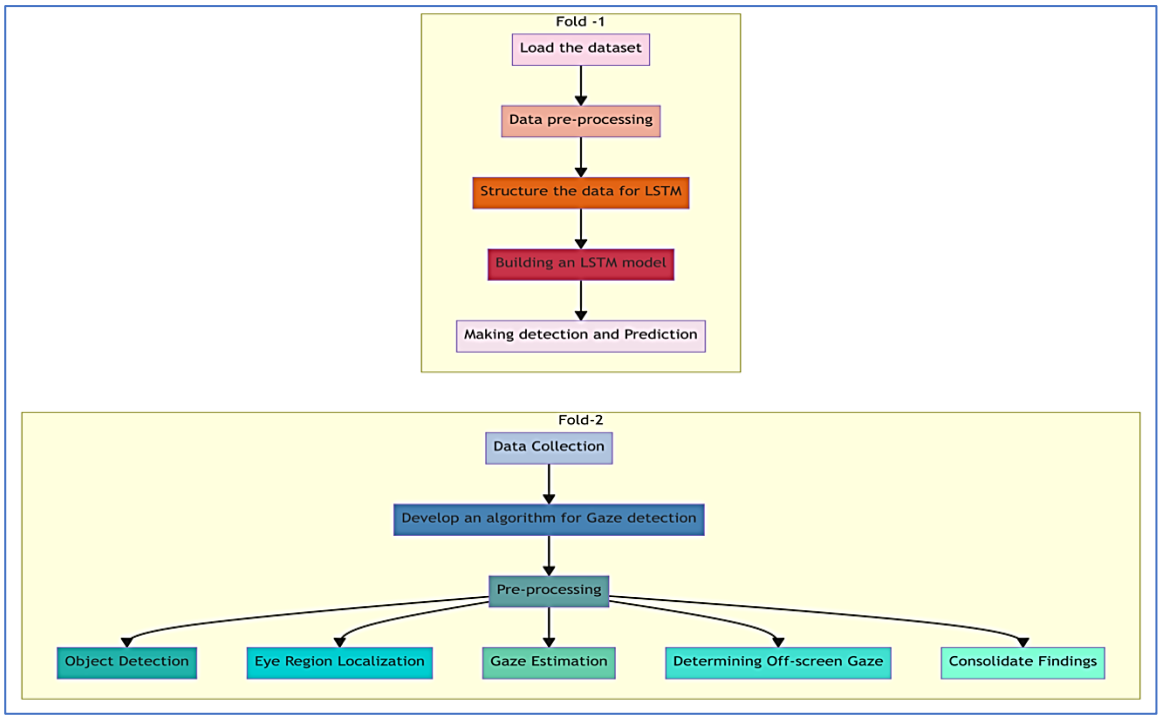


Figure 1. Fold process flow.

Table 3. Dataset description.

Features	Description	Possible values
ID	Unique ID of the student, such as a Roll Number. / Registration Number	1-Inf
Gender	Gender of the student	M- Male F- Female
1st	Marks obtained in the 1st Exam	Value is between 0 and 10 (may have decimals)
2nd	Marks obtained in the 2nd Exam	Value is between 0 and 10 (may have decimals)
3rd	Marks obtained in the 3rd Exam	Value is between 0 and 10 (may have decimals)
4th	Marks obtained in the 4th Exam	Value is between 0 and 10 (may have decimals)
5th	Marks obtained in the 5th Exam	Value is between 0 and 10 (may have decimals)
Status	Result	1- Pass 0- Fail

3.3.2 Operational Logic

Fold-1: Data Analysis Module

Data pre-processing

The system begins with loading and cleaning of dataset. It determines the absent more and the outliers, and normalizes the data in order to provide consistency and reliability in analysis. Missing values from the dataset were handled using mean, median, and mode imputations. More precisely, we imputed missing students' exam scores using the column mean and added missing categorical values (gender) based on the mode. Specifically, we imputed missing numerical exam scores with the column mean and filled missing categorical values (gender) with the mode. Outliers were detected using the Z-score ($|Z| > 3$) and interquartile range (IQR) ($1.5 \times$ IQR) rules and were removed. Data normalization was performed by scaling the data to a range of 0 to 1, and data correlations were identified. Afterwards, the entire dataset

was balanced using SMOTE to avoid oversampling and undersampling. Lastly, to manage the class imbalance in the "status" label, SMOTE (Synthetic Minority Oversampling) is utilised to oversample the minority group. The visual depiction of data processing is shown in **Figure 2**.

Missing value

D_i presents the i^{th} feature column in the dataset D. Let D be the dataset with n features. We can use the following method to identify the missing values in each feature column before imputation, known as $missing_{total\ before}[i]$.

$$missing_{total\ before}[i] = \sum_{j=1}^{m} 1_{\{D_{ij}=NA\}} \quad (1)$$

The number m in the dataset denotes the number of observations. We call the value of the j^{th} observation in the i^{th} feature column D_{ij}. The indicator function gives '1' if the condition inside is true, which in this case is if the data value D_{ij} is missing (NA), and 0 otherwise.

Before any imputation, we would add up all of the feature fields to determine how many missing values there are in the entire dataset:

$$missing_{total\ before} = \sum_{i=1}^{m} missing_{before}[i] \quad (2)$$

Identification of numerical columns

Let C be a collection of columns in dataset D. Each column c has a data type (c) that follows it. We develop a function named *NumericalCols()*, which accepts a dataset and outputs a list of columns:

$$NumericalCols(D) = \{c \in C | type(c) \in \{'float64', 'int64'\}\} \quad (3)$$

This function finds all the columns in C (the set of all the columns in dataset D) where the type of each column c is either "float64" or "int64." These are the most common methods for storing numerical data in computer languages and data analysis tools.

Missing value handling

D is the dataset with columns $c_1, c_2, \ldots, c_n$. Each column c_j has m items, which are $c_{j1}, c_{j2}, \ldots, c_{jm}$. Let μ_{cj} represent the mean of column c_j, a collection of numbers, without accounting for any missing values (NA). The imputation can be represented as follows:

$$\begin{cases} c_{ji} & if\ c_{ji}\ isn't\ NA \\ \mu_{c_j} & if\ c_{ji}\ is\ NA \end{cases} \quad (4)$$

where,

- c_{ji} is the i^{th} element of column c_j in the dataset D.
- c_{ji} is the i^{th} element of column c_j after imputation.
- μ_{cj} is the mean of column c_j calculated as $\mu_{cj} = \frac{1}{m_{not\ NA}} \sum_{i=1}^{m} c_{ji}$, where, m_{not} NA is the count of non-NA elements in column c_j.

Identification of categorical columns

Let C be the set of all the columns in D, and type (c) be a function that tells us what kind of data column c has. This set of category columns is referred to as C_{cat}.

$$C_{cat} = \{c \in C \mid Type(c) = object\} \quad (5)$$

where,

- C_{cat} is the set of categorical columns.
- c is an element representing a column in the dataset.
- C is the set of all columns in the dataset.
- Type(c) is a function that returns the data type of column c.
- The equality Type(c)= object is used as a condition to include a column in the set of categorical columns if its data type is 'object'.

Fill in missing categorical values

The dataset D is called at first place. The set of category columns is c, and the data value in row i and column j is d_{ij}. The model-based correction takes the following form:

$$\begin{cases} mode(C_j) & if\ d_{ij}\ is\ missing \\ d_{ij} & else \end{cases} \quad (6)$$

where,

- d'_1' is the imputed dataset value at row i and column j.
- mode(c_j) is the most frequently occurring value in column j of dataset D.

Calculate and record the sum of missing values after imputation

Let $1_{\{data\_i=NA\}}$ be the indicator function such that:

$$\begin{cases} 1 & if\ data_i{=}NA \\ 0 & else \end{cases} \quad (7)$$

If that's the case, the following formula gives the total number of unknown values after imputation:

$$missing_{after} = \sum_{i=1}^{n} 1_{data_i} = NA \quad (8)$$

where,

- n is the total number of data points in the dataset,
- $data_i$ represents the i^{th} data point,
- NA represents a missing value.

Outlier detection using Z-score and IQR

To find the value of x in a dataset:

$$Z(x) = \frac{x-\mu}{\sigma} \quad (9)$$

In this case, μ represents the dataset's mean, and σ represents its standard deviation.

Z-score outliers are called:

$$outliers_Z = \{x \mid Z(x) > Z_{threshold}\} \quad (10)$$

$Z_{threshold}$ is a predefined number that serves as a threshold. Data points with a Z-score higher than this level are considered outliers.

Outliers' removal

$$data_{clean} = data - outliers_Z \cup outliers_{IQR} \quad (11)$$

where,

- data represents the original dataset.
- $data_{clean}$ represents the dataset after outlier removal.
- $outliers_Z$ is the set of data points identified as outliers by the Z-score method.
- $outliers_{IQR}$ is the set of data points identified as outliers by the IQR method.

Data normalisation

Here's how to figure out the normalised version $x_{normalized}$ of each feature column x in the cleaned dataset:

$$x_{normalized} = \frac{x-\mu_x}{\sigma_x} \quad (12)$$

where,

- x is a vector representing the feature column before normalisation.
- μ_x is the mean of the feature column x in the cleaned dataset.
- σ_x is the standard deviation of the feature column x in the cleaned dataset.
- $x_{normalized}$ is the vector representing the feature column after normalisation.

Correlation finding

For each pair of factors in the dataset, X and Y are strongly related to each other. This is how the equation for the correlation coefficient ρ (X, Y) can be written:

$$\rho(X,Y) = \frac{\sum_{i=1}^{n}(X_i-\bar{X})(Y_i-\bar{Y})}{\sqrt{\sum_{i=1}^{n}(X_i-\bar{X})^2}\sqrt{\sum_{i=1}^{n}(Y_i-\bar{Y})^2}} \quad (13)$$

where,

- X_i and Y_i are the individual sample points indexed with i.
- $\bar{X}$ and $\bar{Y}$ are the sample means of X and Y, respectively.
- n is the number of sample points.

Sequence creation and data reshaping

An LSTM model transforms marks over terms that follow each other into overlapping patterns and reshapes them into a 3D format. Practically, we used a sliding window length of 2 for two successive exam scores to compose each LSTM input sequence, as initial experimentation indicated that this effectively represented short-term performance dynamics. To accomplish these tasks, the following steps are followed:

One hot encoding

Let C be a categorical variable in the dataset with k unique categories. The process of one-hot encoding can be seen in the form of a transformation function, which can $\phi(C_i)$ map each of the categories c_i into a binary vector e_i of length k, but where e_{ij} is the j^{th} element of vector e_i.

The one-hot encoding transformation ϕ can be defined as:

$\phi(C_i) = e_i$,

$$for\ i = 1, 2, 3, \dots, n\ and\ j = 1, 2, \dots, k.$$

If we have a list of all the category variables $C=\{C_1,C_2,\dots,C_n\}$, then the one-hot encoded matrix E can be shown as

$$E = \begin{bmatrix} e_1 \\ e_2 \\ . \\ . \\ . \\ e_n \end{bmatrix} = \begin{bmatrix} \phi(C_1) \\ \phi(C_2) \\ . \\ . \\ . \\ \phi(C_n) \end{bmatrix}.$$

So, here is the equation for putting a set of categorical variables C into a binary matrix E in a single step:

$$E = OneHotEncode(C) \quad (14)$$

Class balancing with SMOTE

X displays the feature matrix of the original dataset, while y displays its name vector. We apply the SMOTE method to this dataset, creating a new, balanced dataset (X', y').

$$(X', y') = SMOTE(X, y) \quad (15)$$

Sequence creation

Let $D = [d_1, d_2, \ldots, d_N]]$ be a dataset of N numerical data points, and let 'l' be the desired sequence length. The function create_sequences generates a set of sequences S, where each sequence s_i is a consecutive subsequence of D.

The sequence creation can be:

$$S = \{s_i | s_i = [d_j, d_{j+1}, \ldots, d_{j+l-1}], \ for\ j = 1\ to\ (N - l + 1)\} \quad (16)$$

According to this equation, S is the collection of all sequences s_i. Each sequence s_i was created by sliding a window of length 'l' from the j^{th} element d_j of the dataset D to the end of the dataset, one element at a time. Therefore, the i^{th} sequence in S from D was taken, starting at position j and ending at position $j+i-1$.

3.3.3 LSTM Model Building

The model architecture comprises input, LSTM, dropout, dense, and output layers. The model assembly includes a loss function and also an optimiser. One of the most important aspects of the training includes the establishment of batch sizes, epochs, and sources of validation data. The model includes two layers of LSTM, each having 50 units, a dropout layer with a dropout rate of 0.2, a dense layer with 50 neurons (ReLU activation), and a 1-neuron output layer (Sigmoid activation in binary classification). The model we have built was based on Adam optimiser with the likelihood of learning rate 0.001 and binary cross-entropy loss. This model was trained for 14 epochs with a batch size of 32. The model is described in detail in Equations (17) to (23). The visual depiction of the model-building process flow is shown in **Figure 3**.

$$\mathcal{M}(X; \theta) \rightarrow Y \quad (17)$$

In this case, $\mathcal{M}$ stands for the LSTM model, θ for the parameters, X for the input sequence, and Y for the output sequence.

$$\theta * \leftarrow \text{Compile}(\mathcal{M}, \Omega, \mathcal{L}, \mu) \quad (18)$$

Here,
Ω: Optimisation function
$\mathcal{L}$: Loss function
μ: Evaluation metrics

$$(\mathcal{D}_{train}, \mathcal{D}_{val}) \leftarrow \text{Split}(\mathcal{D}) \tag{19}$$

$$\theta * \leftarrow \text{Train}(\mathcal{M}, \mathcal{D}_{train}, \mathcal{D}_{val}) \tag{20}$$

$$\mathcal{V} \leftarrow \text{Visualize}(\mathcal{M}, \theta *, \mathcal{D}_{train}, \mathcal{D}_{val}) \tag{21}$$

Here,
$\mathcal{V}$: represents the set of visual objects.

$$\varepsilon \leftarrow \text{Evaluate}(\mathcal{M}, \mathcal{D}_{val}, \mu) \tag{22}$$

Here, ε the metrics score.

$$\mathcal{V}_{eval} \leftarrow \text{Visualize}(\varepsilon) \tag{23}$$

Here, $\mathcal{V}_{eval}$ denotes the visual objects.

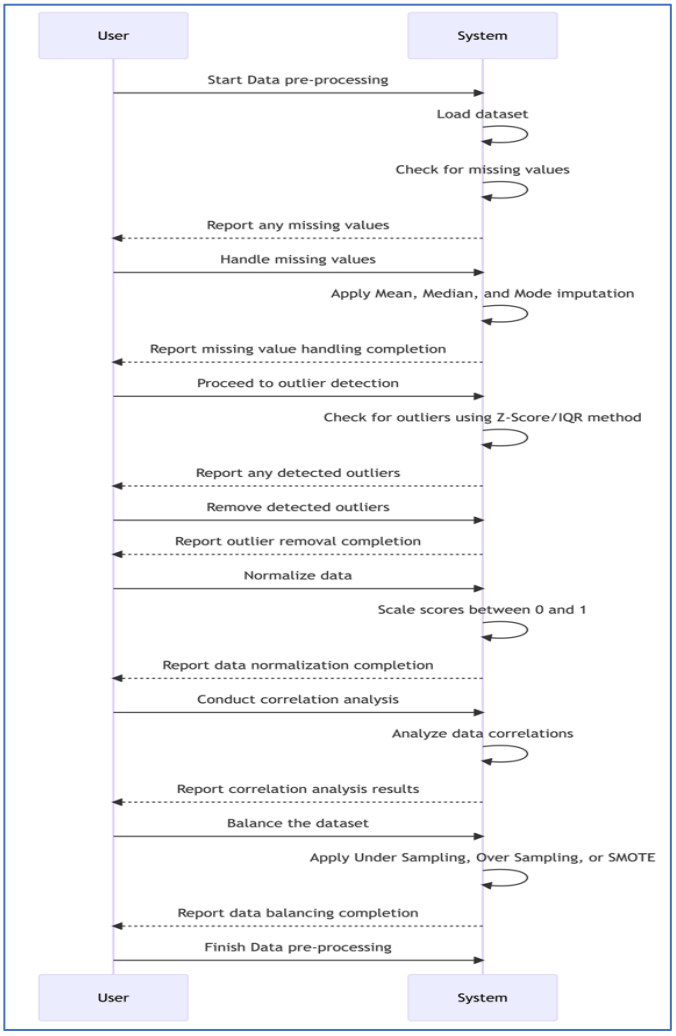


Figure 2. Process sequence of data pre-processing.

3.3.4 Model Evaluation and Tuning

Several variables were used to measure performance and adjust the model as necessary to achieve the best results. All hyperparameters, including the number of LSTM layers, module units, dropout rate, learning rate, batch size, number of epochs, and YOLO detection thresholds, were manually tuned. We used performance validation to fine-tune these values, trying 1 vs. 2 LSTM layers, different dropout rates (0.1 and 0.5), and various learning rates (0.01 and 0.001) until they achieved stable performance. The tuning was heuristic, and no grid search or Bayesian optimisation was carried out. Tools for the visualizing models, for instance TensorBoard, can help for the better understanding purpose of the training as well as validation losses.

3.3.5 Detection and Prediction

To define the possibly suspicious cases of academic dishonesty, suspicious performance indicators of the test set were used to reveal the suspicious records.

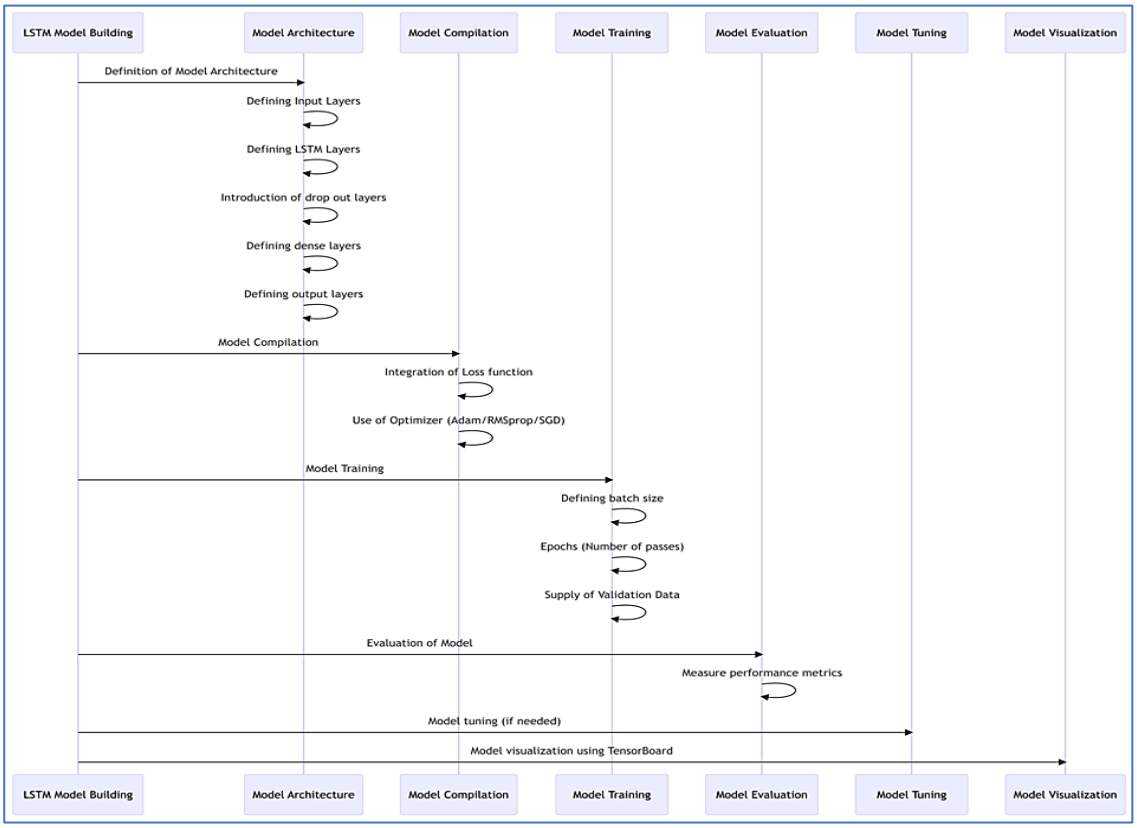


Figure 3. LSTM model building sequence.

3.4 Video Analysis Module (Fold-2)

Fold-2: Video Analysis Module

3.4.1 Data Collection

A set of pre-recorded test videos was made, which was linked to each of the student's ID. It makes it quite easier for conducting individualized analysis.

Dataset description

In the second fold, the data is formed by high-quality, pre-recorded videos of more than 350 students to have taken part in the exam. The videos were shot at 1620 x 1080 resolution. We saved the videos in different formats; it was in .mov, MPEG-4/AAC, WebM, and AVI. Each of the videos is mapped with the specific ID of the particular student which in this case could be a roll number or even a registration number. Also, the video records the date and the time when the video was recorded, which helps to identify the necessary record. **Table 4** describes the dataset.

Table 4. Dataset description.

Attribute	Description	Values
Student Id	The Unique identification of a student	Numeric value
Date	Date of the Exam	Value in Date Format (DD/MM/YYYY)
Time	Time of start and end of the exam	Value in time format (HH:MM)
Video link	The virtual drive link of the video.	Hyperlink to the cloud storage for the respective video

3.4.2 Gaze Detection Algorithm Development

In this step, the algorithms will examine the movements of eye as well as the orientation for determining that either the participant is cheating by looking off-screen.

Video pre-processing

Segmentation of videos in groups of frames was done in order to get more in-depth analysis. The primary goal of this analysis was to identify faces and objects, as well as to determine how people were behaving. During the video pre-processing phase, all videos were down-sampled equally to 30 frames per second to achieve uniformity across recordings and resized to 416×416 pixels to fit the YOLO input format. This calibration improved the processing speed and the detection accuracy of the proposed framework.

Behavioural analysis

The system scans the student's head, eyes, and other features in the video frame to detect any unusual movements or prohibited items, such as cell phones. Object detection, eye direction, head pose, offscreen gaze, and talking anomaly were primarily targeted during this step.

Gaze estimation and off-screen gaze detection

Vector-based estimation and Pupil centre-corneal reflection (PCCR) are two methods used to find the direction of gaze and behaviours of looking off-screen.

Consolidation of findings and data visualisation

Anomalies and look data are combined and visualized to provide a comprehensive view of potential cheating behaviours.

The mathematical formulation of all the aforementioned processes is as follows:
The eye landmarks are calculated to detect the eyes of participants using the Equation (24), where the i^{th} landmark position is represented by $\mathcal{p}_i = (\mathcal{X}_i, \mathcal{Y}_i)$

where,

$$P = \{\mathcal{p}_1, \mathcal{p}_2, \mathcal{p}_3, \dots \mathcal{p}_n\} \quad (24)$$

Once the Eye landmarks $\mathcal{p}_1, \dots \mathcal{p}_6$ are detected, then the eye aspect ratio (EAR) is calculated to detect the blinking for both eyes by calculating the Euclidean distance $\| \cdot \|$ between two points as:

$$EAR = \frac{\|p_2 - p_6\| + \|p_3 - p_5\|}{2 \cdot \|p_1 - p_4\|} \quad (25)$$

To find out if the gaze is directed off-screen, the vector from the centre of the eye to the centre of the detected pupil is calculated and compared to a threshold as follows:

$$\mathcal{G} = \mathcal{D} - \mathcal{C} \quad (26)$$

Here,
$\mathcal{C} = (c_x, c_y)$ for determining the exact location of the eye's centre by averaging its landmarks and
$\mathcal{D} = (d_x, d_y)$ the pupil centre and $\mathcal{G}$ is the Gaze.

To find out if the subject is gazing away from the screen, the direction of $\mathcal{G}$ to certain criteria is compared.

After identifying the gaze direction, it is required to check whether the gaze is off-screen or not. To do so, a threshold vector $\mathcal{T}$ is set as:

$$\mathcal{T} = (t_x, t_y) \quad (27)$$

There is a maximum permissible deviation in the on-screen look, and this is what is represented in Equation (27). If the condition given in Equation (28) is considered as an off-screen gaze:

$$if\ |\mathcal{G}_x| > t_x \ \| \ |\mathcal{G}_y| > t_y \quad (28)$$

From the Equation (24) to (28), the entirety of the procedure can be delineated in Equation (29) as:

$$P = Detect_{Landmarks}(Image) \Longrightarrow$$
$$EAR_{Left} = Calculate_{EAR}(P_{left}), EAR_{right} = Calculate_{EAR}(P_{right}) \Longrightarrow$$
$$G_{left} = Estimate_{Gaze}(P_{left}, D_{left}), G_{right} = Estimate_{Gaze}(P_{right}, D_{right}) \Longrightarrow$$
$$IsOffScree(G_{left}, G_{right}, T) \quad (29)$$

The complete flow of object detection and multiple gaze operation sequence is shown in **Figure 4** and **Figure 5**, respectively.

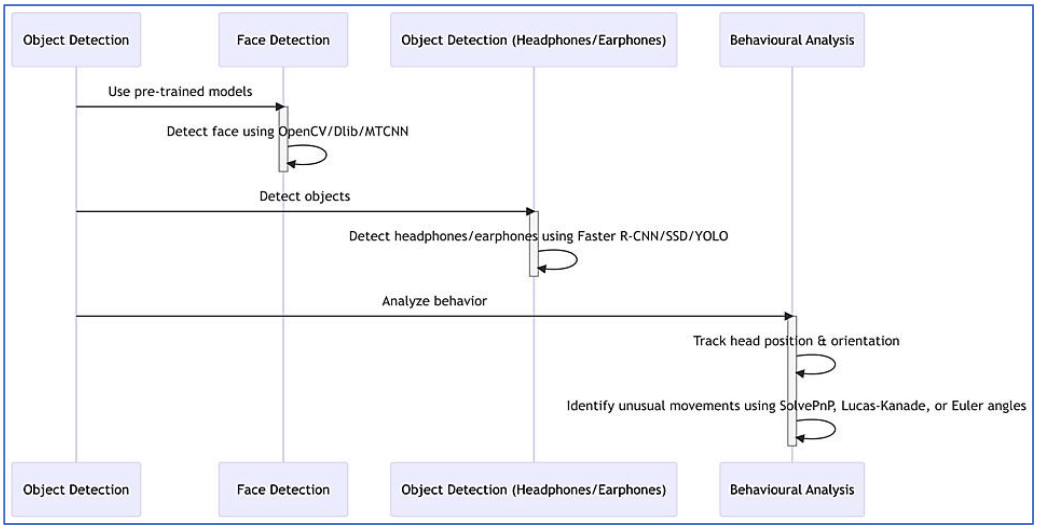


Figure 4. Object detection sequence.

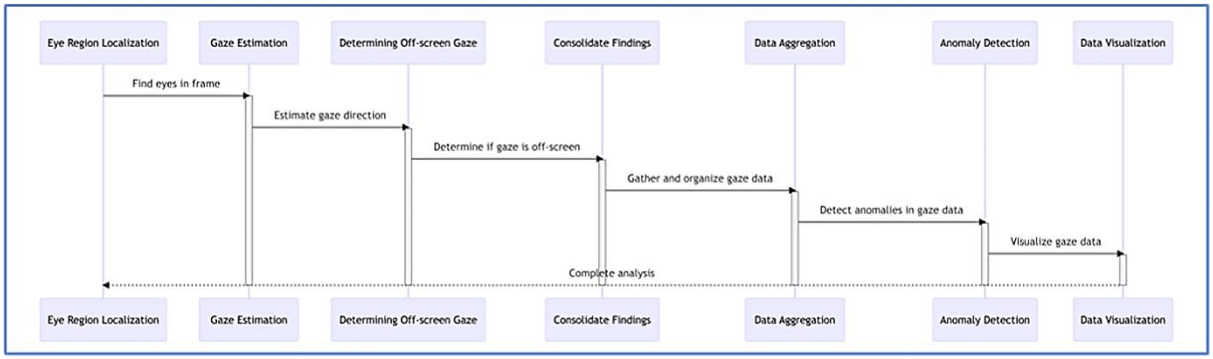


Figure 5. Multiple gaze operations sequence.

3.5 Integration and Hybrid Operation

When Fold-1 and Fold-2 are combined, the system can check the results of both the data analysis and video analysis tools against each other. This makes cheating detection more accurate and reliable.

3.6 Data Collection

Our process of data collection will be properly organized and implemented to test and assess our hybrid proctoring system, thus leading to accurate and reliable outcomes. To make this process effective we gathered video data and academic success data of students when they are doing a test. Assessed the procedures followed in gathering the data in terms of the size of the group, demographics, and ethics.

3.6.1 Academic Performance Data Collection (Fold-1)

Source: A custom dataset was made which has term results of the students. Those data appeared in the form of scores from different classes and across various quarters.

Sample Size: The LSTM model required a large sample size to achieve statistically significant and stable results. This should cover the records of thousands of students over several term examinations.

3.6.2 Video Data Collection (Fold-2)

Creation of Dataset: A custom dataset comprised the video recording of test sessions of 350 students. Videos captured the student's face, upper body, and workspace to make a proper study.

Sample Size: In particular, the video data samples should be sufficiently large in order to train and test the computer vision programs. This is very diverse, whereby the video data samples of the discrepancies of behaviours can vary in dozens up to hundreds based on the complexity of detection algorithms.

Demographics: Video clips were best taken among the wide variety of the students in terms of age, gender, race and physical features. This diversity played a significant role in developing programs that are not biased against diverse students.

3.7 Evaluation Metrics

The usefulness, speed and accuracy of the hybrid proctoring system must be tested to ensure that the hybrid proctoring system works as desired without interfering with the quality and integrity of the education system. Hence, different metrics were used to measure the success of the system in different angles. These

measures not only indicate that the system has the potential of identifying academic dishonesty, but they also show its efficiency and influence upon the experiences of the users.

3.8 Effectiveness Metrics

Accuracy

It checks that how many of the guesses that the system made, were right which includes both of the cases either cheating or not cheating. High accuracy means that the system can reliably tell the difference between honest and dishonest behaviour (Jierula et al., 2021).

$$Accuracy = {}^{TP + TN}/_{TP + TN + FP + FN} \quad (30)$$

where,

TP stands for True Positives, i.e., when an instance is correctly identified as positive. TN would be the True Negatives, meaning when an instance is correctly identified as a negative instance. FP is the number of wrong predictions for an instance to be positive, and FN is the number of wrong predictions for an instance to be negative.

Precision

It checks whether the system can identify the actual cases of cheating out of those that are labelled as fake (Foody, 2023). This is done to prevent the system from falsely accusing innocent students.

$$Precision = {}^{TP}/_{TP + FP} \quad (31)$$

Recall (Sensitivity)

It determines if the system can afford to find every instance of cheating. This will keep recall high to provide assurances that the overall integrity of the testing process is maintained, where no instance of cheating is overlooked (Foody, 2024).

$$Sensitivity = {}^{TP}/_{TP + FN} \quad (32)$$

F*1 *Score

It combines precision and recall into a single measurement by computing their harmonic mean. The *F*1 score reflects the effectiveness of the method quite well, especially when classes are presented in an imbalanced distribution (Yarlagadda et al., 2024).

$$F1_{Score} = 2 \times \left({}^{(Precision * Sensitivity)}/_{(Precision + Sensitivituy)} \right) \quad (33)$$

4. Implementation

4.1 Development Tools and Technologies

Python: Python platform is the foundation of our development process which is a versatile as well as widely used programming language. It is readable with a large library ecosystem. It supports most of the tools used for data science and machine learning, ranging from TensorFlow for deep learning to Pandas for data analysis and manipulation. This makes it an excellent choice for video processing and data analysis within our system.

- **Keras and TensorFlow:** In this project, these high-level neural network frameworks are employed, which run on top of TensorFlow, to build and train our model for analyzing student performance from marks and videos (Joseph et al., 2021).
- **OpenCV:** The Open-CV (Open-Source Computer Vision Library) is one of the least important

materials of our video analysis module. It has many functions to manage the pictures and videos on the spot. It is effective when it comes to finding faces, tracing the look and identifying objects (Zelinsky, 2009).

- **Dlib:** It is popular since it includes machine learning algorithms and computer vision and pictures processing tools specifically (Yang and Fan, 2023). We apply the facial landmark detector of 68 points used by Dlib, to follow the smooth facial features which are important in examining gaze and behaviour.
- **Multi-Task Cascaded Convolutional Networks (MTCNN):** MTCNN based face tracking and recognition system is more precise particularly in the detection of faces under various types of lighting, which is important because it helps in preserving the integrity of the proctoring process.
- **MediaPipe:** This model was what Google had created as a cross-platform model to create machine learning pipelines based on video, music, and any form of time series (Bora et al., 2023). It is particularly useful in such activities as recognizing facial expressions and gestures, thus allowing our system to be easier in identifying the manner in which students would conduct themselves throughout tests.
- **TensorBoard:** This instrument is important in revealing data on the training process and performance of the neural network models. TensorBoard makes our models perfect by visualizing the way parameters and metrics evolve through time (Huang and Le, 2021).
- **Jupyter Notebook:** This open-source web application allows us to create and share documents with live code, equations, visualisations, and textual narrative. Exploratory data analysis, machine learning, and sharing early results with the study team are all good uses (Dombrowski et al., 2023).

4.2 Integration with Learning Management Systems (LMS)

Fold-1: All operations of Fold-1 are divided into four distinct sections: data pre-processing and visualisation of the analysis results; data encoding, balancing, and sequence creation; model building, training, and evaluation; and model prediction and suspicious record detection. The complete workflow of the Fold-1 is shown in **Figure 6**. The pseudocode for these four sections is given below:

Algorithm 1: Data Preprocessing and Visualisation

```
procedure DATA_PREPROCESSING_AND_VISUALIZATION
IMPORT necessary libraries for data manipulation, statistics, and visualisation
SET file_path to the location of the dataset
LOAD data from file_path into a dataframe
DISPLAY the first few entries of the dataframe for an overview
CALCULATE and record the sum of missing values before imputation
IDENTIFY numerical columns in the dataset
FILL missing numerical values with the column's mean and median
IDENTIFY categorical columns in the dataset
FILL missing categorical values with the mode
CALCULATE and record the sum of missing values after imputation
VISUALIZE missing data before and after imputation using heatmaps
DETECT outliers using Z-score and IQR methods
REMOVE outliers and visualise data using boxplots before and after outlier removal
NORMALIZE the data using StandardScaler and visualise using boxplots
CALCULATE correlation matrix and visualise using heatmap
DISPLAY the sorted correlation values and visualise class distribution
end procedure
```

Algorithm 2: Data Encoding, Balancing, and Sequence Creation

procedure DATA_ENCODING_BALANCING_SEQUENCE_CREATION
ENCODE categorical variables using OneHotEncoder as:
$Let\ f: C \rightarrow E$
$f(c_i) = e_i\ for\ i = 1, 2, 3, \ldots, n$
BALANCE the classes using SMOTE as:
$SMOTE: (X, y) \rightarrow (X', y')$
$(X', y') = SMOTE(X, y)$
VISUALIZE the new class distribution as:
$Let\ V: S \rightarrow G$
$V(S) = G$
DEFINE create_sequences function to generate data sequences as:
$create_{Sequences}: (D, l) \rightarrow S$
$S = create_{Sequences}: (D, l)$
CREATE sequences from numerical data as:
$Seq = \{seq_1, seq_2, \ldots, seq_m\}, with\ seq = (d_{i1}, d_{i2,} d_{i3} \ldots, d_{il})\ for\ i = 1,2, \ldots, m$
SPLIT the data into features and labels as:
$S \rightarrow (X, y)$
end procedure

Algorithm 3: LSTM Model Building, Training, and Evaluation

procedure LSTM_MODEL_BUILDING_TRAINING_EVALUATION
DEFINE the LSTM model architecture with specified layers as:
$\mathcal{M}(X; \theta) \rightarrow Y$
Here, $\mathcal{M}$ denotes the LSTM model, θ as parameters and X as an Input Sequence and Yas an Output sequence.
COMPILE the model with optimiser, loss function, and metrics
SPLIT the data into training and validation sets
TRAIN the model using the training set and validate with the validation set
VISUALIZE the training and validation accuracy and loss
EVALUATE the model using accuracy, precision, recall, F1 score, and MAE
VISUALIZE the evaluation metrics
end procedure

Algorithm 4: Model Prediction and Suspicious Record Detection

procedure MODEL_PREDICTION_SUSPICIOUS_RECORD_DETECTION
LOAD the trained LSTM model
PREDICT on new test data sequences
IDENTIFY suspicious records based on predictions
OUTPUT the indices of suspicious records
end procedure

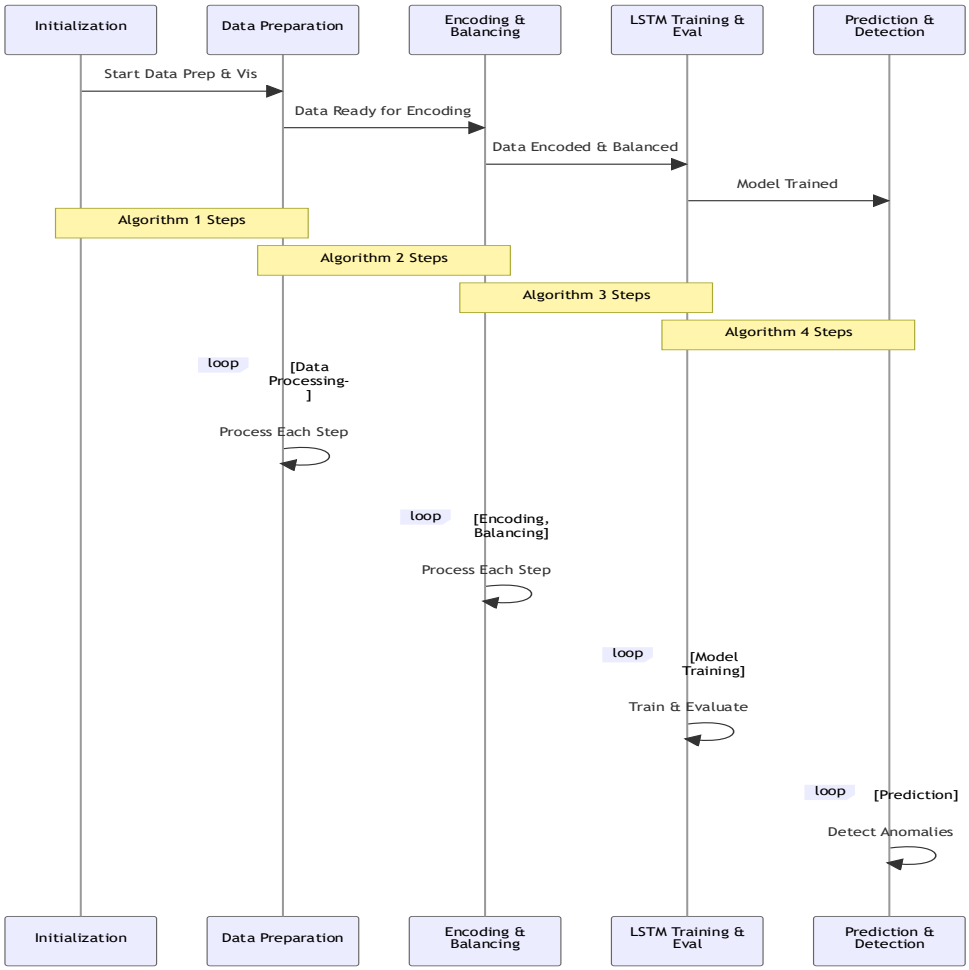


Figure 6. Workflow of fold-1 algorithms.

Fold-2: In Fold-2, the YOLO algorithm is implemented for facial landmark and object detection. It involves the CNN face predictor, shape predictor, and video processor. In this flow, the anomalies are also detected. The complete workflow of this Fold-2 is visualised in **Figure 7**.

Algorithm 1: Facial Landmark Detection and YOLO Object Detection
PROCEDURE INITIALIZE_MODELS
LOAD YOLO model from YOLO_WEIGHTS and YOLO_CONFIG
LOAD CNN_FACE_DETECTOR from CNN_FACE_DETECTOR_PATH
LOAD SHAPE_PREDICTOR from PREDICTOR_PATH
END PROCEDURE
PROCEDURE PROCESS_VIDEO(video_path, frames_dir, frame_rate)
CALL INITIALIZE_MODELS
SET net, output_layers to output from LOAD_YOLO()
CALL video_to_frames(video_path, frames_dir, frame_rate) to extract frames

```
INITIALIZE frame_count, previous_opening
FOR EACH frame_file in sorted frames_dir
READ image from frame_path
CALL detect_faces_cnn() to detect faces
CALL estimate_head_pose() for head pose estimation
FOR EACH face in faces
DETECT facial landmarks using SHAPE_PREDICTOR
CHECK for yawning with is_yawning()
CHECK for mouth opening with is_mouth_open()
CALL localize_eye_regions_and_estimate_gaze()
CALL detect_objects_yolo() for YOLO object detection
CALL draw_yolo_detections() to visualize detections
INCREMENT frame_count
END FOR
CLOSE all windows
CALL detect_anomalies() for post-processing analysis
END PROCEDURE
PROCEDURE MAIN
SET video_path, frames_dir, frame_rate to user-defined paths and rate
CALL PROCESS_VIDEO(video_path, frames_dir, frame_rate)
END PROCEDURE
```

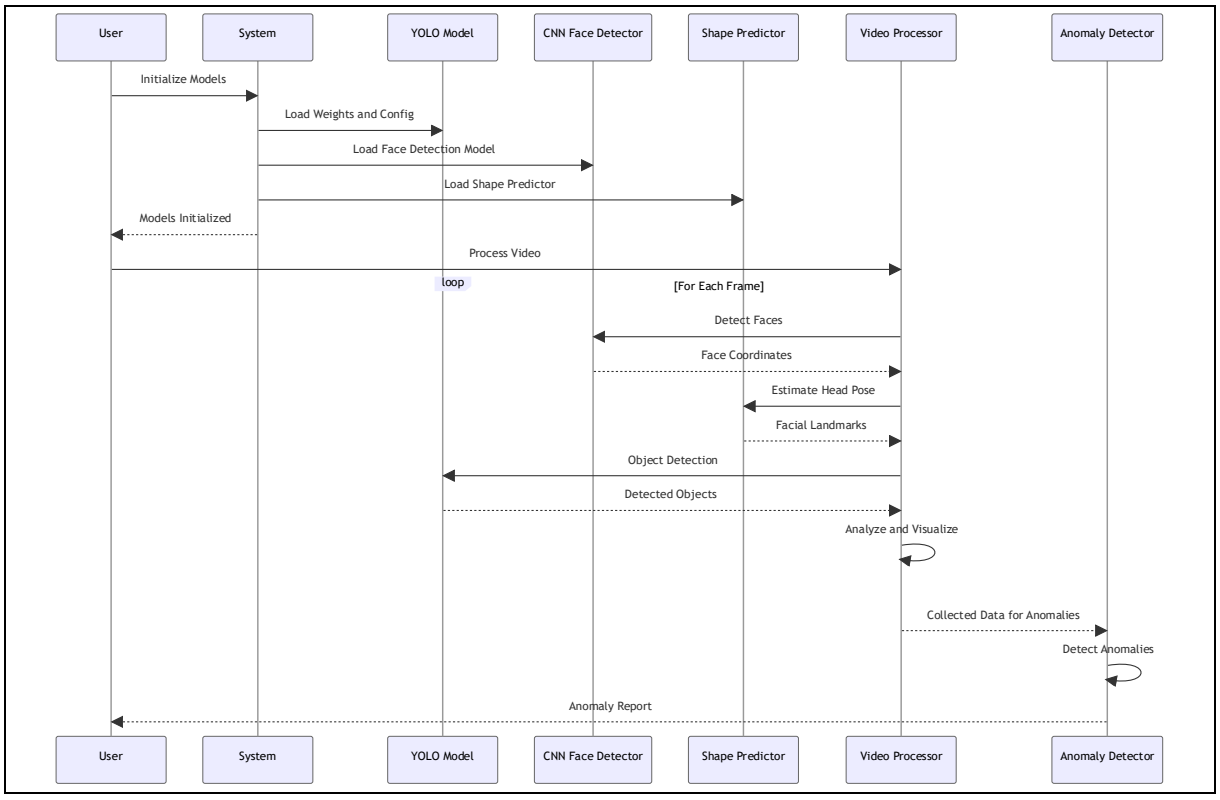


Figure 7. Flow of the fold-2 algorithm.

5. Results and Discussion

5.1 Fold-1

The results for Fold-1 demonstrate that the entire procedure, including data normalization, outlier identification, missing value imputation, data manipulation, and sequence construction for LSTM model training, can accurately detect dishonest behavior. At start, it takes a dataset and fills in missing values via imputation, applying the mean to numerical columns and the mode to categorical ones. To improve the quality of the dataset, outliers were identified and removed using the IQR and Z-score techniques. Next, the data were standardised to enhance the model's training efficiency. The processed data were organised into sequences that were fed into the model input to identify possible instances of dishonesty. Statistical analysis and visual representations of suspicious records revealed potential instances of dishonest behaviour, demonstrating the model's capacity to detect patterns suggestive of cheating.

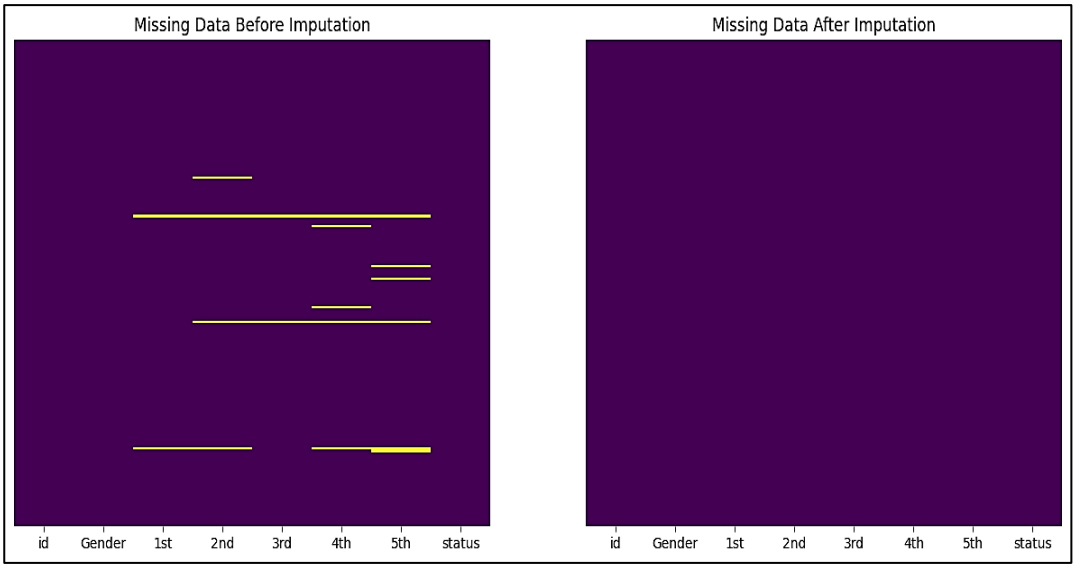


Figure 8. Missing data handling.

Figure 8 above presents a startling before-and-after comparison of a dataset with missing data. *'Missing Data Before Imputation*' is the name of the chart on the left, bordered by horizontal yellow lines, is named *'Missing Data Before Imputation'*. Each line represents gaps in the dataset across various categories, such as *'id'*, *'gender'*, and e*xam scores ('1st' through '5th')*. The '*Missing Data After Imputation'* graphic is a solid purple block with no lines. This graphical transformation is an indication of the imputation process which entails replacement of missing variables with more comprehensive analysis.

Table 5. Outlier detection summary by *Z*-score and IQR outlier method.

Features	*Z*-score outliers	IQR outliers
id	0	28
1st	0	14
2nd	0	6
3rd	0	3
4th	0	4
5th	2	4
status	0	3

Table 5 which is above placed, presents two distinct approaches for the identification of dataset outliers. Most features have a value of '0' in the *'Z-Score Outliers'* column, making it nearly empty. Except for the '5th' data point, which had 2, the *Z*-score method—which is analogous to a normal ruler—found almost no data point that appeared out of place. The *'IQR Outliers'* column, in contrast, shows a higher level of activity. Similar to a custom-fit tool, the IQR (interquartile range) uncovered more outliers. Three outliers in the *'status'* category also suggest that there were some cases that didn't follow the expected pattern.

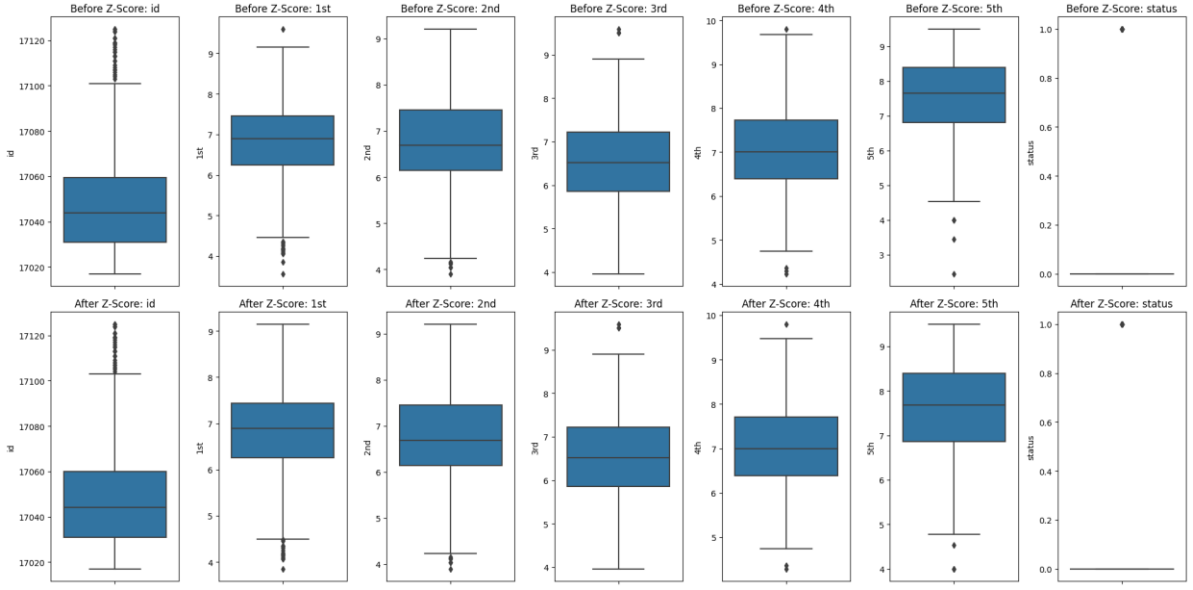


Figure 9. Outlier handling.

Each pair of box plots in **Figure 9** above shows a different variable, both before and after *Z*-score normalization. At the top, an assortment of *'Before Z-Score'* box plots can be seen displaying 'id' and scores from the first through fifth tests, including *'status'*. Median exam scores are around 7, but there is a wide range of dispersion between them; some are quite tightly packed, while others appear loosely packed, similar to 1620 × 1080. Through their transformation and standardisation, a mean of 0 and a standard deviation of 1 were achieved. It provided a constant baseline for comparison. The range consistently displayed that the whiskers and boxes are more aligned with the midline. Although the outliers are still there, they are now more prominent in contrast to the otherwise homogeneous data, making it simpler to identify anything out of the ordinary.

In **Figure 10**, there are two box plots: one showing the data before normalization and the other showing the data afterwards. The extremely dispersed ranges seen on the left side of the *'Before Normalisation'* chart suggest that the values of 'id' and exam scores from 1st to 5th are significantly off-kilter. Since the *'status'* only takes on two values, it appears to be a binary indicator.

On the right side, the *'After Normalisation'* plot can be seen, where the scores are compressed into a uniform range. The vertical lines, also called *'whiskers'*, capture the outliers—those small dots that appear to have wandered off from the pack—and expand to display the whole data range. The bands across the colourful boxes indicate the median, and the boxes themselves reflect the middle 50% of scores.

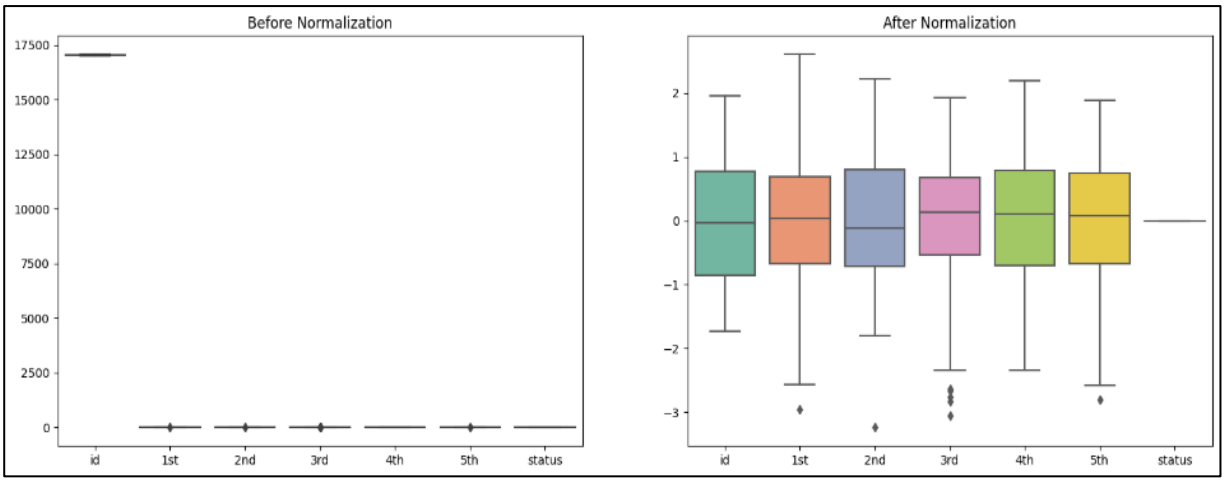


Figure 10. Data normalisation.

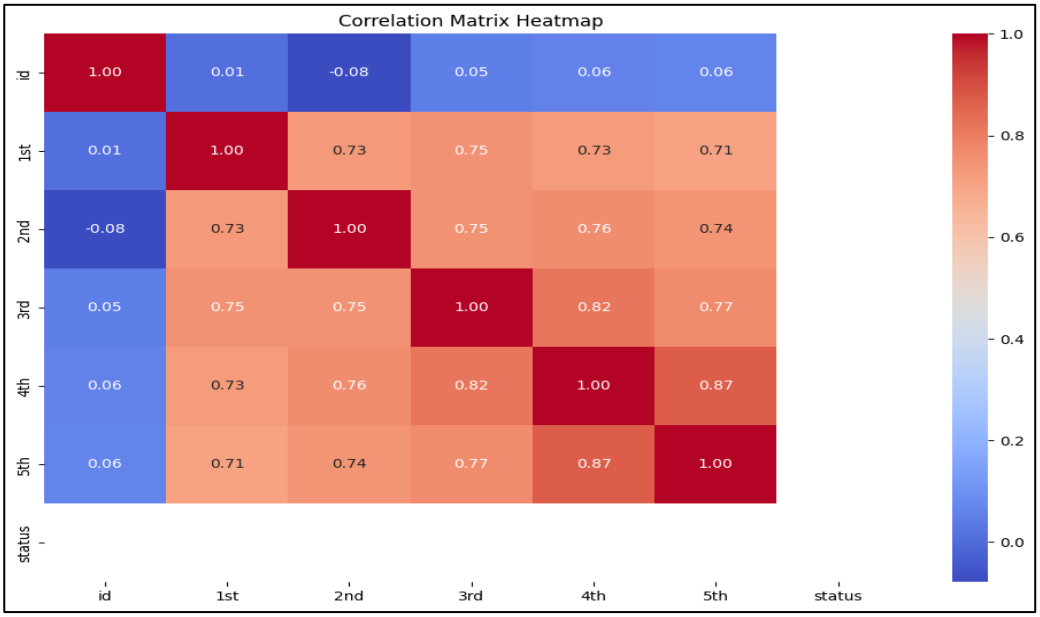
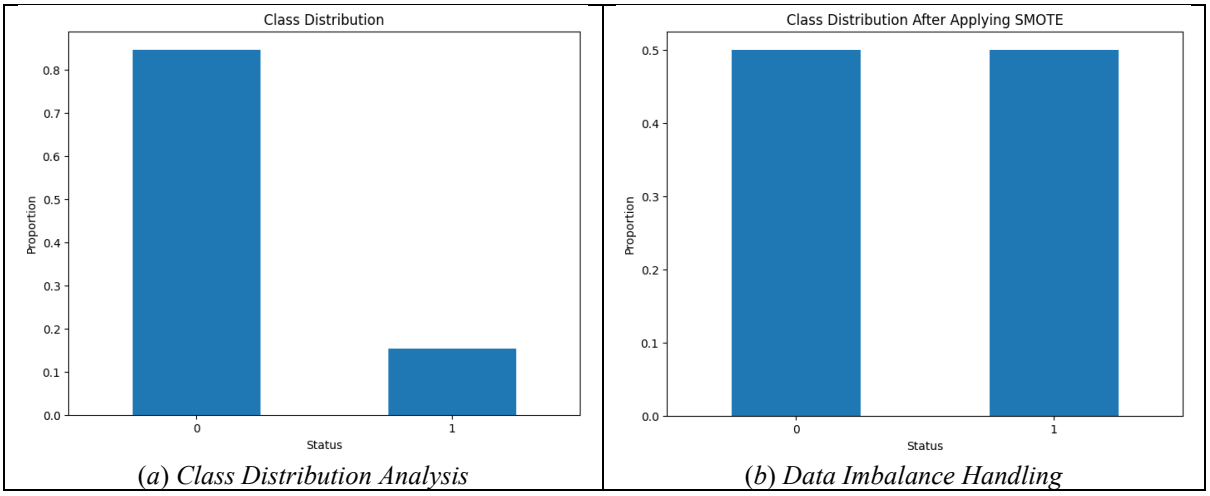


Figure 11. Correlation finding.

Figure 11 above illustrates the correlation between all seven variables. Each square depicts the degree of correlation between test results. The above visualisation indicates a correlation between students' performance on the first and second exams. However, the correlation between exam scores and the *'status'* attribute appears to be less significant.



(*a*) *Class Distribution Analysis* (*b*) *Data Imbalance Handling*

Figure 12. Class distribution.

Figure 12 above illustrates the previous and current state of the dataset's class distribution. There is a striking disparity between the two classes in **Figure 12 (a)**. Class *'0'* towers over the scene, while Class *'1'* is almost invisible. **Figure 12 (b)** shows that the implementation of SMOTE nearly evenly represents both categories of the *'status'* attribute.

Table 6. Model description.

Layer (type)	Output shape	Param #
lstm_2 (LSTM)	(None, 2, 50)	11200
lstm_3 (LSTM)	(None, 50)	20200
dropout_1 (Dropout)	(None, 50)	0
dense_2 (Dense)	(None, 50)	2550
dense_3 (Dense)	(None, 1)	51
Total params:		34,001
Trainable params:		34,001
Non-trainable params:		0

This model is well-suited for jobs that require learning from time-series data, as shown in **Table 6**, because it has an LSTM layer at the top with 11,200 configurable parameters. The layers produced two 50-unit sequences. After that, there's another LSTM layer that streamlined the complicated patterns found earlier; this one has a 50-unit, simpler structure. To improve the model's intelligence, it introduced 20200 new parameters. The next step was a dropout layer, which helped the model generalise better without adding new parameters and prevented overfitting by randomly removing data points. Lastly, the design features two thick layers that are responsible for producing predictions; the first layer utilizes fifty units to enhance decision-making, and the second layer reduces this to one unit for the output. The model, that is with 34001 trainable parameters, have the ability that it can adjust its accuracy which is based on the input data.

As seen in **Table 7**, **Figure 13** compares the different performance measures from the beginning to the end of the model training by different epochs. Two metrics are exhibited in the validation and training processes, which are accuracy and loss metrics. The decrease in the loss rate after the first epoch is very steep which implies that the prediction accuracy of the model on the training sets is much higher. The training accuracy increases steeply but thereafter levels out indicating that there is a good fit between the model and training

data and this is coherent with the observations. Validation accuracy and validation loss, on the other hand, have a slight upward trend but are fairly stable along the epochs. The model is not overfitting the validation data set.

Table 7. Model training statistics.

Epochs	Time	Training loss	Training accuracy	Validation loss	Validation accuracy
1	2s 196ms/step	0.3293	0.8688	0.4056	0.8049
2	0s 38ms/step	-0.3209	-0.85	-0.4054	0.8049
3	0s 27ms/step	-0.3165	-0.8625	-0.3976	0.878
4	0s 39ms/step	-0.3178	-0.8688	-0.3845	0.878
5	0s 24ms/step	-0.3145	-0.875	-0.3966	0.878
6	0s 15ms/step	-0.3108	-0.8688	-0.3954	0.878
7	0s 19ms/step	-0.3055	-0.8875	-0.3879	0.878
8	0s 21ms/step	-0.3163	-0.875	-0.3932	0.878
9	0s 16ms/step	-0.3089	-0.875	-0.3989	0.878
10	0s 17ms/step	-0.3075	-0.8813	-0.4048	0.878
11	0s 15ms/step	-0.3156	-0.8813	-0.3891	0.878
12	0s 14ms/step	-0.3121	-0.8813	-0.3939	0.878
13	0s 14ms/step	-0.3156	-0.8813	-0.4026	0.878
14	0s 16ms/step	0.3114	0.8813	-0.403	0.878

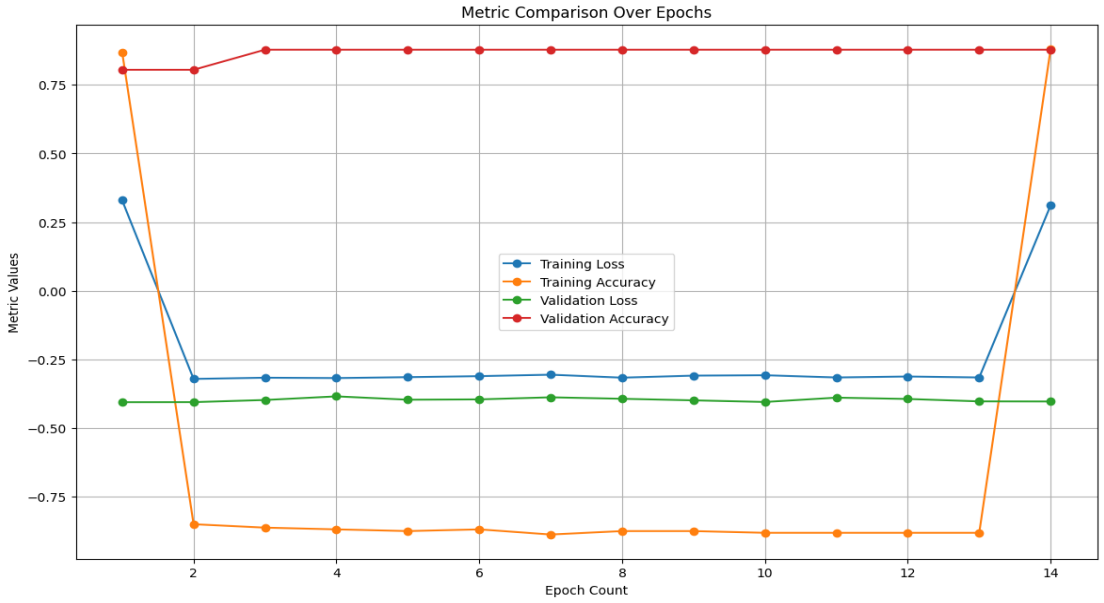


Figure 13. Metric comparison over epochs.

Table 8 gives a short description of particular students whose results in the test have garned attention. Physical addresses like student ID and gender are appropriately ordered as identified in sequence 1 to 5. The trend of analysis using the marks shows that the fourth and fifth tests have a higher average than the first three tests which show that there has been a great improvement in grades. This inclination is what defines the effectiveness of the model in the detection of an outlier.

Table 8. Suspected records identified by the model.

Id	Gender	1st	2nd	3rd	4th	5th	Status
172	M	6.96	6.24	6.85	7.21	7.71	0
173	F	4.89	4.76	4.7	6.67	8.57	1
174	F	4.25	4.04	4.06	6.53	8.95	1
175	M	6.3	6.24	5.85	6.36	7	0
176	F	4.16	4.66	4.44	6.54	8.85	1
178	M	7.11	7.41	7	7.32	8.32	0
179	M	4.34	4.59	4.66	6.7	9	1
180	F	4.06	4.24	4.43	6.76	8.89	1
183	M	5.93	5.86	4.7	5.5	6.21	0
184	F	5	4.36	4.47	6.69	8.99	1
185	F	4.62	4.13	4.98	6.94	8.75	1
186	M	4.17	4.33	4.68	6.53	8.8	1
188	F	7.48	7.55	7.67	7.39	8.65	0
189	F	4.48	4.28	4.24	6.76	8.68	1
190	M	4.5	4.58	4.24	6.58	8.93	1
192	M	7.04	7.1	6.81	7	6.92	0
193	M	4.92	4.04	4.17	6.69	8.98	1
194	M	4.96	4.12	4.11	6.79	8.87	1
195	M	4.15	4.68	4.66	6.93	8.55	1
196	M	4.06	4.47	4.12	6.73	8.99	1
198	M	6.7	6.81	6.52	5.39	7	0
199	M	4.92	4.54	4.92	6.91	8.98	1

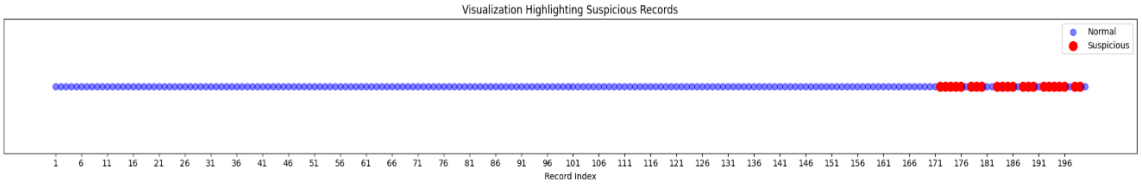


Figure 14. Visualisation of suspicious records among all records.

Figure 14 is a clear representation of student ID in line. The blue dots that we have created have represented the non-suspicious student IDs. On the other hand, the red dots are representing student IDs which have been suspected to be suspicious.

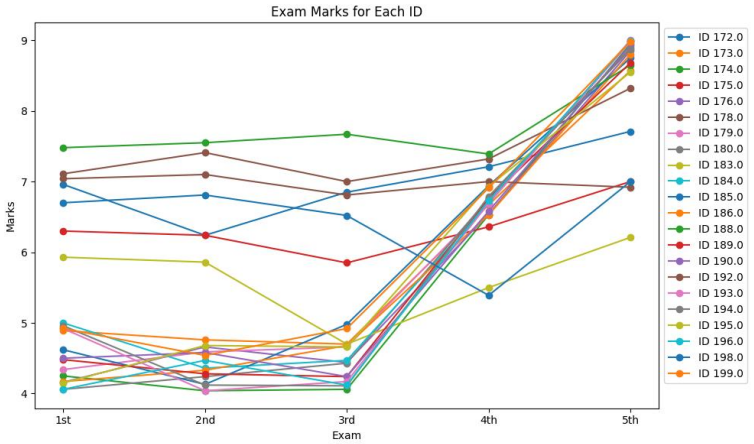


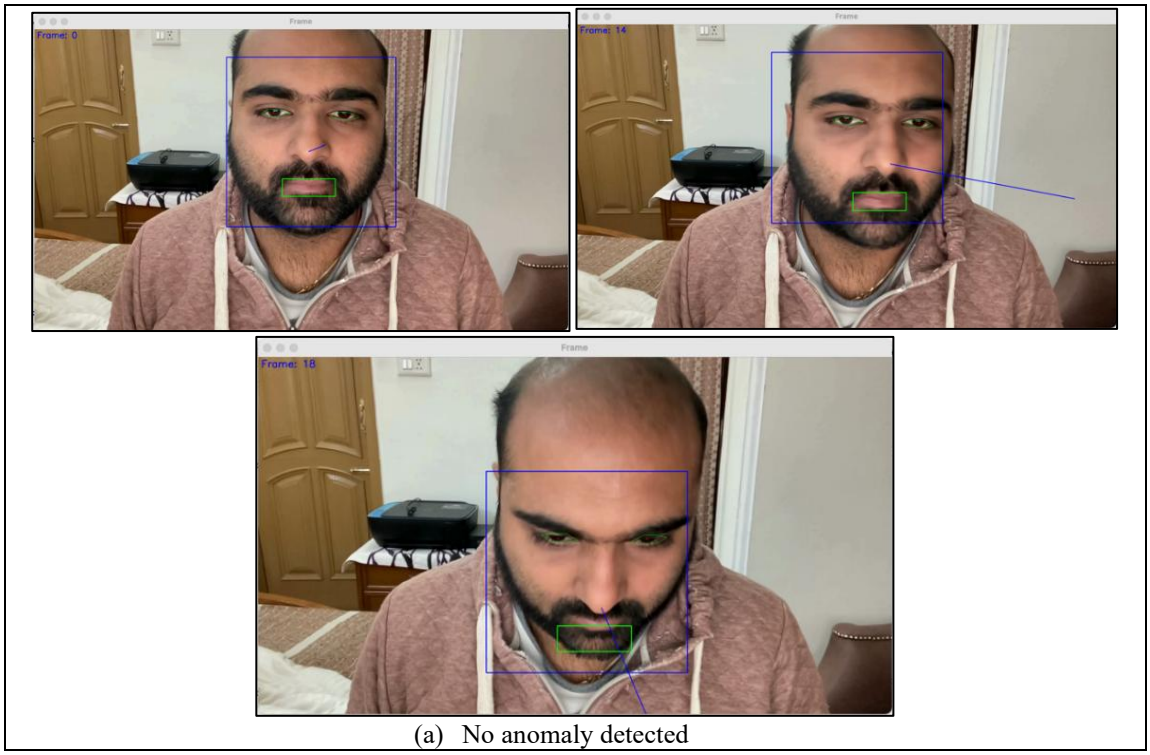
Figure 15. Marks' visualisation of the suspected student.

Figure 15 gives a summary of the findings of a group of students who are represented by an ID score upon five separate assessments. The grades of a student in one test to another that a student gets are represented by each line. The order of the exams starting with the first to the fifth is shown in the X-axis. The obtained scores are plotted on the Y-axis and seem to range between 4 and 9. There are other students whose pattern is constant because they remain within a given range of scores, and this can be indicative of constant capacity. There are some overlapping lines that indicate the changing positions of the pupils when they completed every test. The lines portray the performance trends of the various candidates in one way or another.

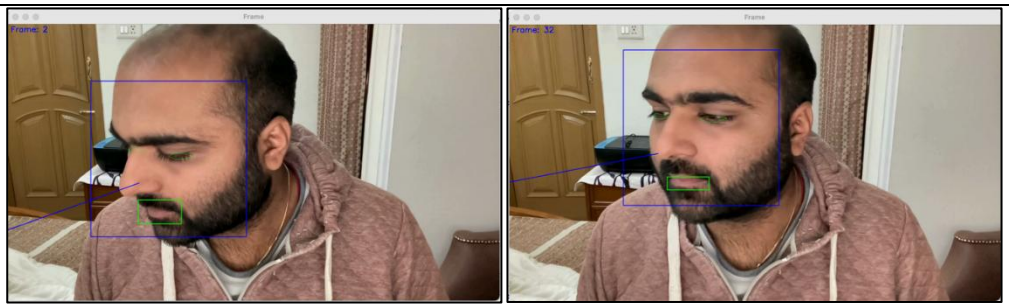
5.2 Fold-2

This section discusses the results obtained from fold-2. This discussion will analyse frames and behaviours from the participant's pre-recorded video over time. The system divides the uploaded video into specific frames and analyzes each frame for anomaly detection using various methods, including gaze direction, head pose estimation, pupil detection, talking detection, object detection, eye aspect ratio calculation, yawning detection, and eye region localization, among others.

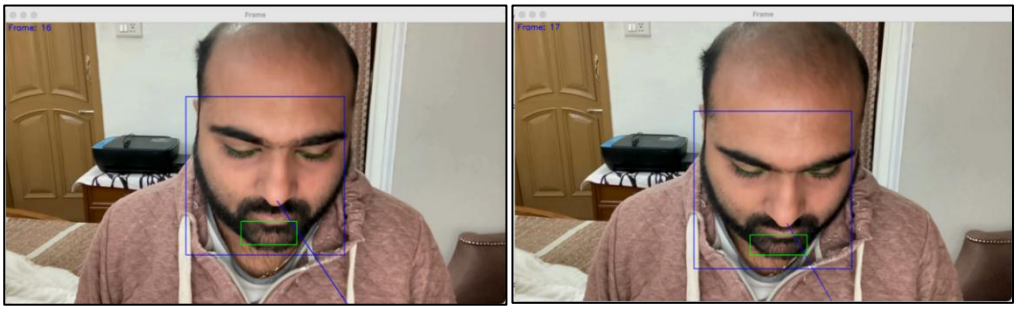
The proctoring system expects the frontal gaze during the exam, and an anomaly occurs if the expected gaze does not match the detected gaze in video frames. **Figure 16** demonstrates anomaly detection using frame-based analysis. We divided **Figure 16** into seven subfigures: (a), (b), (c),...(g). These subfigures have the representation for distinct analyses of the frames during detecting anomalies.



(a) No anomaly detected

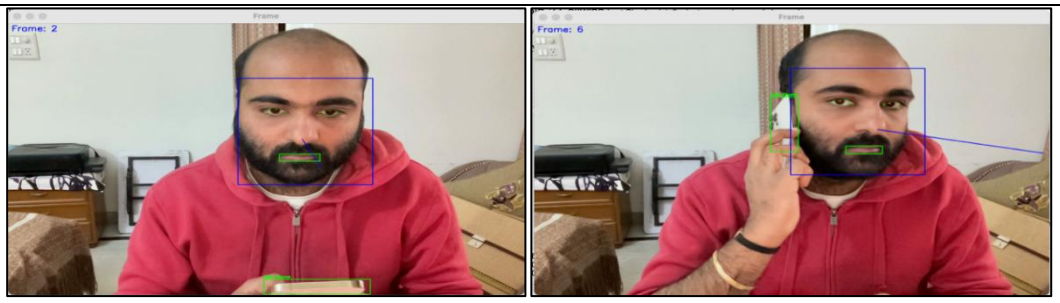


(b) Off-screen gaze: head pose (Right)

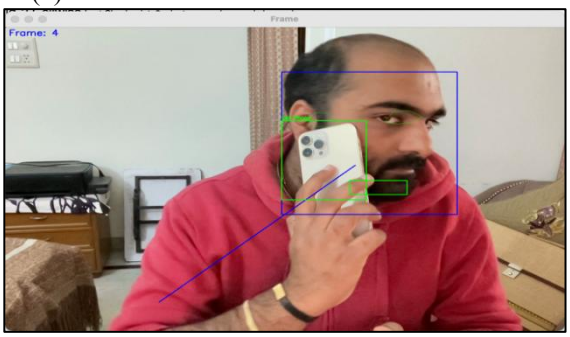


(c) Off screen gaze: head pose (Down)

(d) Off screen gaze: eye direction (Right)

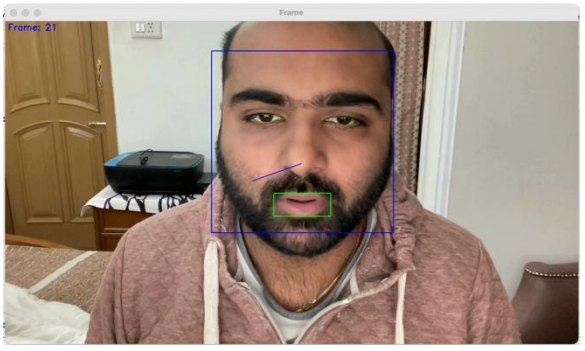


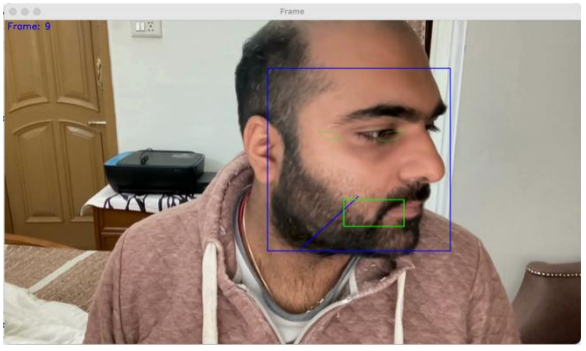
e (1) e (2)



e (3)

(e) Cell phone detected (e(1) and e(2); Off screen gaze: head pose (Right) and cell phone detected (e(3))





(f) Talking anomaly (g) Off-screen gaze: head pose (Left)

Figure 16. System's analysis (Anomaly or No anomaly).

Figure 16 (a) indicates that there are no anomalies in frames 0, 14 and 18. In **Figure 16 (b)**, an anomaly, named as *'Off-Screen Gaze: Looking Right'* is detected in frames 2 and 32 wherein the candidate is looking at his right side. **Figure 16 (c)** shows the identification of an anomaly called *'Off-Screen Gaze: Looking Down'* in frame 16 and 17, which implies the candidate will look down. **Figure 16 (d)** indicates that an anomaly called *'Off Screen Gaze: Eye Direction (Right)'* was detected in the frames 22, 24, 26, and 31 where the candidate has his eye direction on the right side. As shown in **Figure 16 (e)**, there are two separate anomalies that are observed: *'Cell Phone Detected'* in frames (1) and (2), and *'Off Screen Gaze: Head Pose (right) and Cell Phone Detected'* in frame (3). These are the abnormalities that were detected in the frame numbers 2, 6 and 4 where the candidate was captured with cell phone in his hand and looking at his right

side. **Figure 16 (f)** shows that in frame 21, there is a violation in the form of the so-called *'Talking Anomaly'* in which the mouth movement of the candidate can be likened to a conversation. In the meantime, **Figure 16(g)** shows the anomaly of the 'Off-Screen Gaze: Head Pose (Left)' when the candidate is glancing at his left side.

The proposed model after evaluation of the frame studies different attributes of this frame such as the EAR and the Blinks, Yawning Events, gaze direction analysis, landmark detection analysis in the X and Y direction, object detection frequency and performance measurements as shown in the figures below.

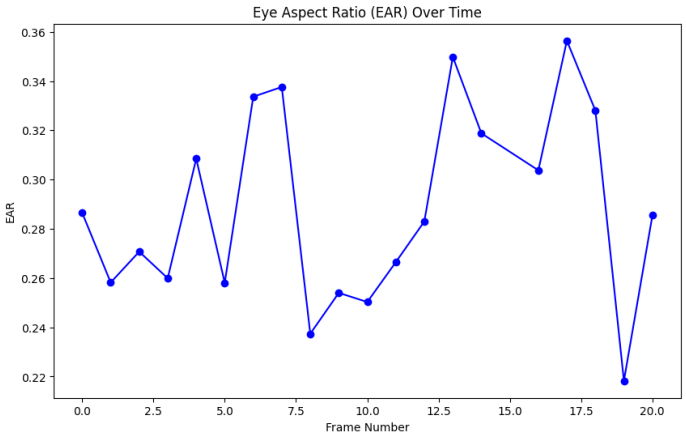


Figure 17. EAR over time.

Figure 17 illustrates EAR moving across a set of frames, which were probably the times of various timestamps. Some of the methods of measuring ocular openness include the EAR metric. Each dot in this chart symbolizes an EAR at a certain moment of the video. The EAR values are variable, running up and down like a series of eye blink or eye opening and concealed, beginning at the left. The declines in the values of the EAR can be observed around frames 5, 10, and a little after 15, indicating periods when the eyes were partially closed or blinking, respectively, when they were the least open. Specifically, the peaks around the frames numbered 7 and 20 show that those were the times when the eyes were wide open.

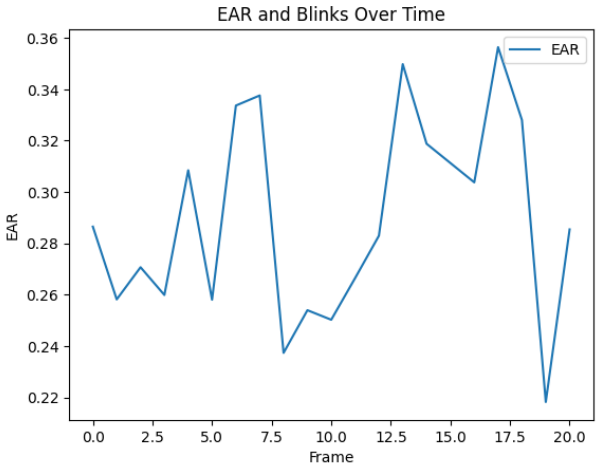


Figure 18. EAR and blinks over time.

EAR, which is a commonly applied measure in the eye tracking to determine blinks and also assess eye closure is depicted as a time-based trend in **Figure 18**. We simply make a plot of the EAR with a set of frames with the vertical axis showing the values of the EAR and the horizontal axis showing the number of frames. To the extent that the EAR is low then it implies that the eyes are closed or blinking, whereas a high value implies that they are wide open.

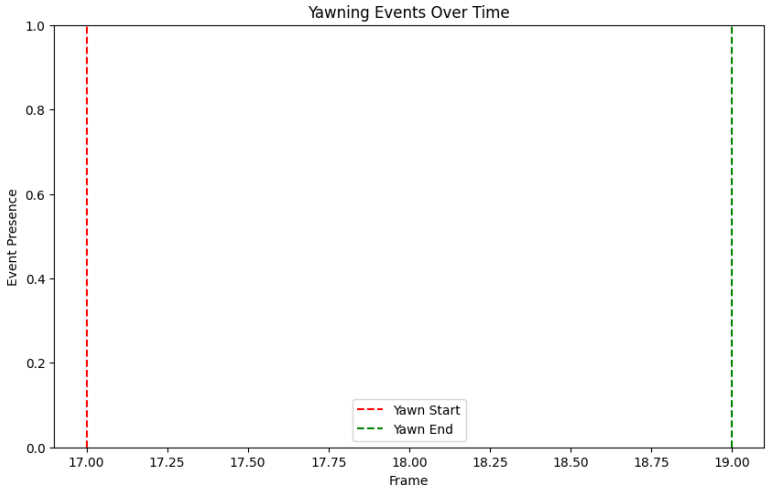


Figure 19. Yawning events over time.

Figure 19 showing a timeline chart which tracks the occurrences of yawning across the several video frames as well as records the beginning along with the ending times of each of the yawn. On the horizontal axis, the *'Frame'* represents specific frames from the video. The vertical axis displays *'Event Presence'*, indicating whether an event has occurred or not. The graph displays two distinct dashed lines, one in green and one in red. At approximately frame 17, a yawning incident begins, as indicated by the red dashed line. Near frame 19, the yawning incident comes to a close, as indicated by the green dashed line. The Figure identifies a yawn between these two lines.

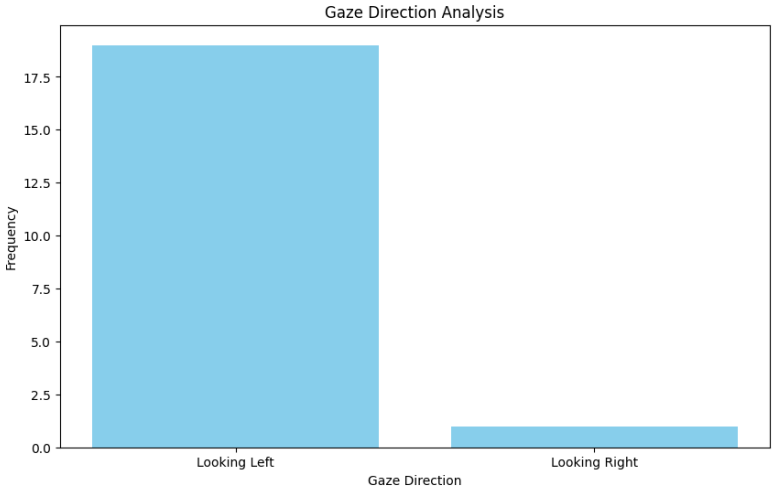
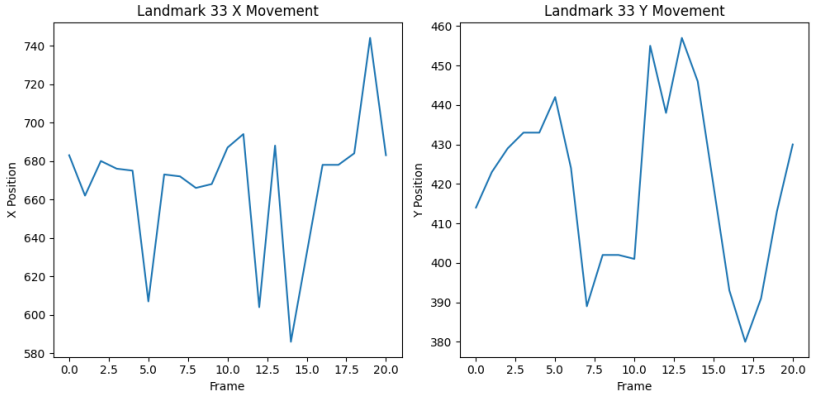


Figure 20. Gaze detection analysis.

Figure 20 captures two gaze directions, *'Looking Left'* and *'Looking Right'*. This visualisation makes it clear that the candidate spent a lot of time looking to the left, since the *'Looking Left'* bar is significantly higher than the *'Looking Right'* bar. To be more precise, their primary gaze was to the left, with occasional glances to the right.



(a) (b)

Figure 21. Landmark movements analysis.

In **Figure 21,** landmark 33 shows the movement of a single landmark, which may be any location on a human face, such as the bridge of the nose, in facial tracking systems. Individual graphs in this image track the movements of this landmark.

Positioned horizontally (X-axis) relative to the video frames from frame 0 to frame 20, **Figure 21(a)** displays this landmark's location. This graph shows a change in the landmark's horizontal location from frame to frame. As the video progresses, the line's numerous peaks and valleys indicate that the landmark is hopping horizontally across the screen. **Figure 21(b)** follows the same landmark's vertical (Y-axis) location during the same time intervals. Again, this graph shows a lot of ups and downs, which could mean that the landmark is changing position. The candidate may be nodding, speaking, or shifting his stance, while the camera may be moving up and down. As a whole, these graphs illustrate the movement of the subject or object at this spot over time.

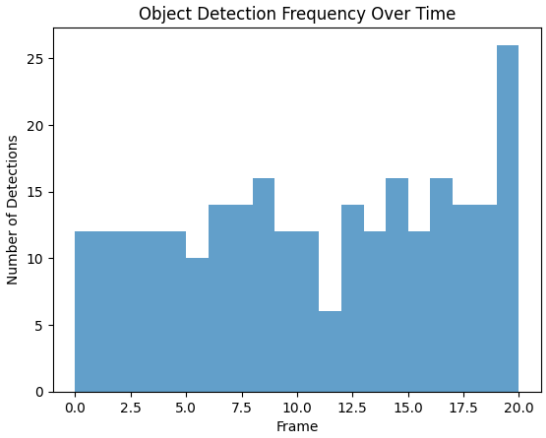


Figure 22. Object detection frequency analysis.

Figure 22 depicts the number of objects detected over time in a sequence of frames. The *'Frame'* label, running horizontally from zero to twenty, depicts a video's frame rate. With a range of 0 to somewhat more than 25, the *'Number of Detections'* labelled vertically displays the count of objects identified. As seen in the Figure, the number of detections changes from one frame to the next. More than ten detections occur in the beginning, from frames 0 to 2.5. Around frame 10, there's a clear decline in detections, followed by a substantial increase towards the end, reaching a peak around frame 20.

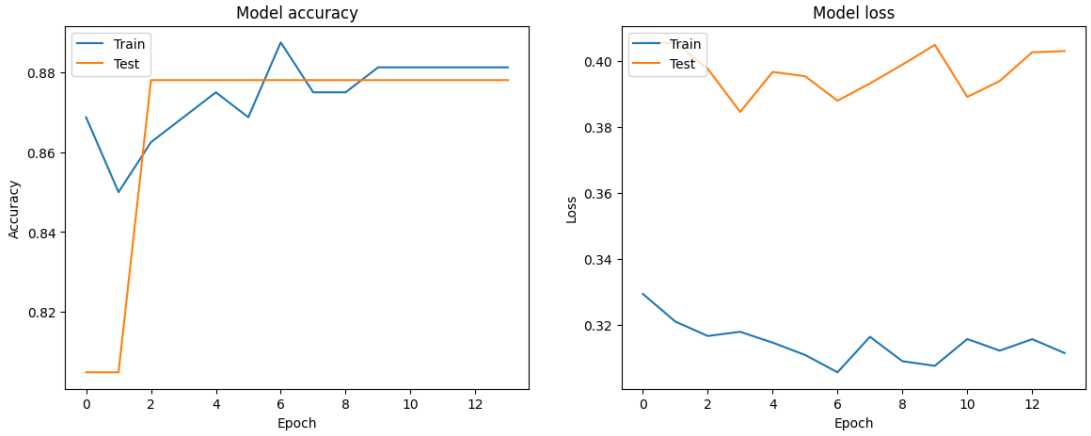


Figure 23. Model's accuracy and loss visualisation.

On the left-hand side of **Figure 23**, a 'Model Accuracy' graph is displayed, which shows the accuracy measure for both the training and testing data. The training accuracy demonstrates that the model consistently and effectively predicts the training data. The fact that the test accuracy is slightly lower indicates that the model is also good at generalising to new, unseen data. On the right, the '*Model Loss*' graph shows the error rate. A small decrease in the loss for the training data indicates that the model is improving at utilizing the data it has learned from, as it starts making fewer mistakes. On the other hand, the test loss is increasing, indicating that the gap between the model's predictions and the test data results is widening as training progresses. The statistical presentation of various model metrics is provided in **Table 9**.

Table 9. Performance metrics of the model.

Metrics	**Value**
Accuracy	0.878
Precision	1
Recall	0.25
F1 Score	0.40
Mean Absolute Error	0.15

Figure 24 visualises accuracy, precision, recall, *F*1 score, and MAE. The model exhibits a combined accuracy score of 0.87. The *'precision'* score is at 1.00, meaning the model is nearly spot-on with its predictions. The *'Recall'* score is lower (0.25), indicating that the model fails to detect 25% of real occurrences. The *'F*1 *Score'*, which measures the model's accuracy and recall, is 0.40, indicating that it has memory issues because it is close to the recall value. The *'Mean Absolute Error'* of 0.15 indicates that the model's predictions are reasonably close to the real values. The model, while generally accurate, fails to capture all the events that it should. This is important for determining the model's strengths and areas for improvement.

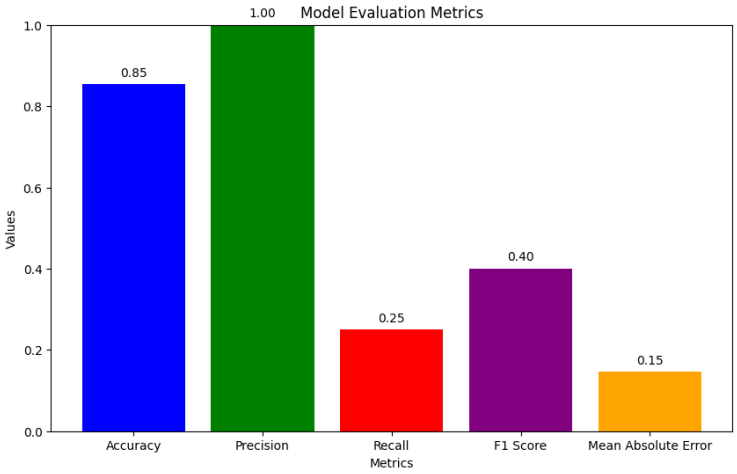


Figure 24. Model performance metrics visualisation.

5.3 Statistical Validation of Model Performance

The binomial significance test was used to statistically confirm the performance of the proposed model because it is an effective way to determine whether the observed accuracy improvement above the baseline level is caused by chance or actual superiority of the model. This test is suitable for binary classification projects, and therefore, it is the best to be used to assess the accuracy of the proposed proctoring system in terms of cheating detection. Its simplicity, interpretability and reliability guarantee a strong measure of statistical confidence of the results reported (https://files.eric.ed.gov/fulltext/ED460146.pdf).

5.3.1 Baseline Definition

Before performing the binomial significance test, it is required to define the baseline value of the metric under consideration. During this step, the original literature identified in Section 2.1 was considered to establish a baseline accuracy, demonstrating whether the performance enhancements were significant or not. According to Arianti et al. (2023), an object detection-based system has achieved an effectiveness of 73.1%, and Ngo et al. (2024) have reported an accuracy of 78.5% for their proposed model through the use of real-time behavior monitoring approaches. On the same note, Ahmad et al. (2021) have recorded a rate of 97.21 percent detection of cheating on face features; nonetheless, they have not ignored some of the limitations such as the fact that they have not tested it in real-time scenarios. The reason why we have opted to use these three studies is that their context perfectly fits our research objective. Through these three studies, it was found the mean accuracy of similar procedures in contemporary literature was found to be about 83%. This was coined as an appropriate level of accuracy baseline.

5.3.2 Binomial Test

The suggested hybrid proctoring system had an overall score of 88 percent on a test group of 350 students when it detected suspicious behaviours. To approve the fact that the perceived increase in the percentage was significant, the binomial significance test was conducted in order to examine whether the perceived surge in the percentage has significant differences with the baseline value of 83 percent. According to the basic assumption, the model identified 308 parallel to 350 test cases as correct and this is in line with the expected number of 287 cases to be correct. Two-sided binomial test had given a p-value of 0.01214 which corresponds to the fact that the performance that was observed was statistically significantly higher than the performance of 83% at the 95% confidence level. The given accuracy was observed with a 95%

confidence interval of [84.6, 91.4], which is not overlapping with the 83 percent baseline accuracy in the literature, which is an additional indication of the strength of improvement as well. In general, the result of this test confirms that the joint accuracy findings of the suggested system are statistically significant ($p \leq 0.05$). The overall results of the binomial test conducted using the MS Excel are indicated in **Table 10** below.

Table 10. Binomial test.

Binomial test		
Sample size (n)	350	
Observed correct predictions (k)	308	
Baseline accuracy (p)	0.83	
Probability of exact k successes under Baseline	0.002212	
Probability of $\leq k$ successes under Baseline	0.996141	
Two-sided *p*-value	**0.01214**	(Statistically Significant)

The expected results, i.e. expected per base accuracy of 83 and mode of 291 correct predictions (averaged) are depicted by this binomial distribution graph labeled as **Figure 25**. The actual predicted result of 308 correct projections (88 percent) lies in the right tail, which means that the level of performance is more than the baseline.

In conclusion, the work of the suggested system not only had an empirically strong performance but also was statistically sound in contrast to the known AI-related proctoring solutions.

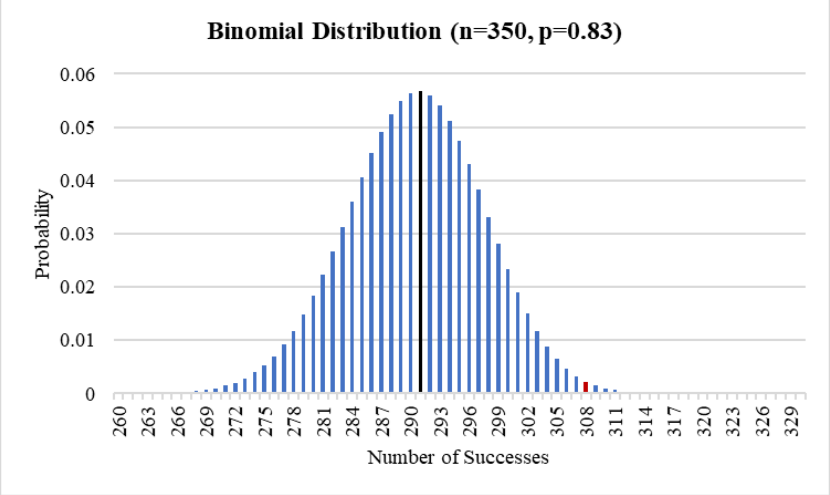


Figure 25. Binomial distribution.

5.4 System Performance

In order to discuss the computational performance of the system, this paper will discuss processing time/frame, CPU use as time moves on, and memory use as time moves on. The following graphs show the performance indicators of the system with respect to processing of the frames. The computing requirements of the processing, any bottlenecks, and optimisation can be more clearly understood with the insight identified in the images below. The system seems to reach a steady state once it has finished running with the initial set up or the initial few frames, as the CPU and memory consumption appears fairly constant.

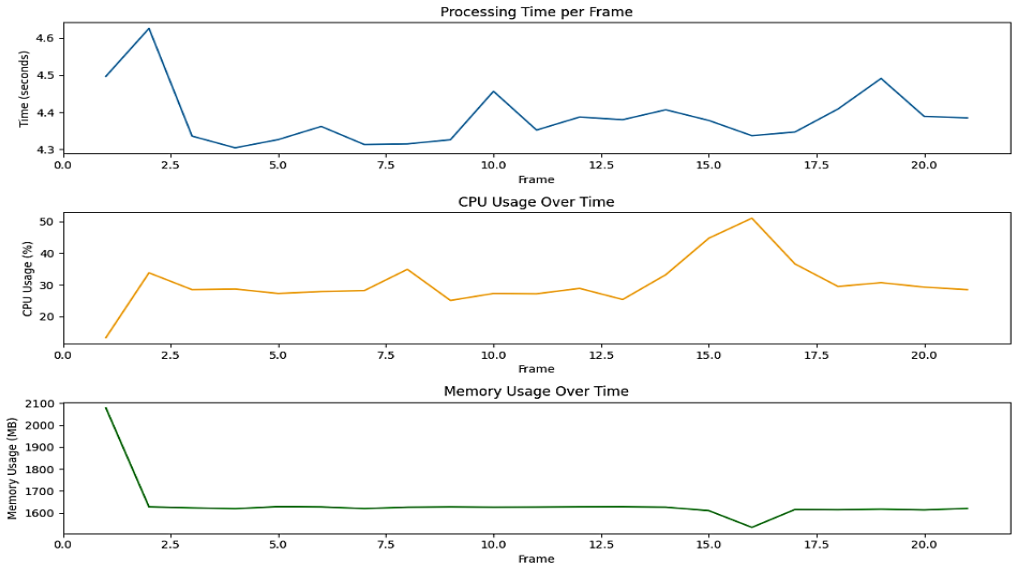


Figure 26. (a) Processing time/frame, (b) CPU usage, (c) Memory usage.

Figure 26 shows a number of performance metrics of a computer system. In our work, these graphs are plotted against the total number of frames which are processed.

Figure 26(a) presents the processing time of each frame and displayed in seconds. The processing time seems to be a bit different but it usually ranges between 4.3 and 4.6 seconds. This variation may have been occasioned by the complexity of either frame or functional computing activities. The graph in **Figure 26(b)** shows the rate at which the CPU was used when the frames were being processed. It is variable and maxima points to the fact that there are frames that will be more taxing to the CPU than others. Utilization is generally constant, except for a noticeable surge at around 45%, which may be the effect of a very strenuous frame, or the initiation of some sort of background activity. The memory utilization in Megabytes (MB) is shown in **Figure 26(c)**. At the beginning, it experiences a sharp reduction of more than 2000 MB to some 1600 MB and then it levels off.

5.5 Comparative Analysis with State-of-the-art Techniques

Table 11 shows a comparison of the proposed methodology with the previous studies on the basis of various essential detection techniques.

Table 11. Essential parameters of the study comparison.

Studies	Object detection	Head pose detection	Eye direction	Talking anomalies detection
Arianti et al. (2023)	P	O	O	O
Ngo et al. (2024)	O	P	P	O
Alguacil et al. (2024)	O	O	O	O
Noorbehbahani et al. (2022)	O	O	O	O
Holden (2021)	O	O	O	O
Ege and Ceyhan (2023)	P	O	O	P
Ahmad et al. (2021)	P	O	P	O
Dadak et al. (2022)	P	O	O	P
Bilen & Matros (2021)	O	O	O	O

6. Discussion

Besides the accuracy and efficiency indicators presented in the previous section, practical significance of the suggested hybrid proctoring system is in its flexibility in operating operations under a number of circumstances in different online learning. In the case of higher educational institutions, the system can be readily integrated within the existing Learning Management Systems (LMS) system to offer another wave of academic integrity guarantee in the process of administering remote academic exams. High stakes testing that involves final exams, entrance tests and graduate level testing can also be implemented using the proposed AI-based proctoring system as a means to identify who is taking the test and avoiding cheating in online classes. It also enables distance learning in a merit-based manner since it would limit the logistical drawbacks and increase sustainability through less student movement and Campus congestion. Also, this system can be applied by the professional certification agencies who can administer secured exams to the candidates throughout the world.

The two-stage design is particularly useful when the course is a massive open online course (MOOC), with thousands of students enrolled at once: since there are anomalies in their performance, the Fold-1 can quickly detect suspicious users, which is more affordable in terms of computational expenses and allows scaling the monitoring up, respectively, and followed by the more costly CNN analysis. This is in order to make sure that it is able to track large groups of people without stretching as far as institutional capabilities allow. Applications of Fold-2 in the real world would be to introduce additional value like anomalies in a video-based representation, like that of mobile phones, earphones, or off-screen gazes. This will help teachers to make their judgments better, as they will have visible, interpretable behavioral data concerning dishonesty, rather than relying solely on score-based anomalies. The synergistic nature of the fast initial screening (Fold-1) and fine-grained behavioural analysis (Fold-2) system makes the system more dependable and credible in real practice deployments.

The proposed framework will not have to substitute the role of human invigilators: rather, it will be one of the instruments of monitoring used alongside other methods in a hybrid proctoring environment, e.g., in a course blending learning or a professional certification test. It is possible that the frame-by-frame analysis of Fold-2 will be helpful to human proctors because it emphasizes some abnormalities automatically and can help to decrease the number of cognitive labor involved in manual surveillance and to avoid neglecting anomalies. Not only can it guarantee a better odds of identifying fake actions, but it also enhances trust between the students and the teachers since the quantitative tool of performance trends and the qualitative tool of behavioral hints are employed to make decisions. It is worth noting that the model is far-widened in references to considering the objectives of education since it intensifies impartiality, veracity, and reliability in the online learning environments. Making the system efficient and ethically right, i.e., with regard to protecting personal privacy, enhancing transparency in detecting anomalies, and lowering false accuses will contribute to the practical value and encourage the use of valid digital assessments as the SDG 4 goals should be promoted.

Even so, there are the ethical and pedagogical benefits of adopting the proposed model, though its technical performance is rather significant. The system minimizes on the intrusion of privacy and discomfort on the students by a two stage design, which eliminates the unnecessary full video monitoring of students. This will be applied to only the students identified as flagged by Fold-1 to provide detailed behavioural analysis which will be provided by Fold-2. The approach fosters equity because it is not based on advanced apparatus or extremely solid Internet connectivity, which leads to the equal opportunities of access to education. Also, the built-in list of anomalies (e.g., the off-screen gaze, head tracking, phone detection, talking) enhances the transparency and the trust between the students and the instructors. These interpretable outputs may be linked to the training of invigilators, who can get more prepared to address alerts, be able to distinguish

between real cheating and false positives, and make students feel confident in the process. In conclusion, our system shows how technology may be merged with ethical responsibility and then be used in a sustainable application in real world learning settings.

7. Conclusion and Future Work

This paper introduces an AI-assisted hybrid proctoring system, which comprises an LSTM-based performance inspection as well as a CNN-based behavioral analysis for the enhancement of the integrity of the online testing. Since students should be focusing on their tests most of the time, the system is configured to detect a standard gaze, which is straight ahead. We refer to this phenomenon as *'Anomaly'* when students frequently look away, either to the right, left, or down, for a considerable amount of time. The system has identified different frames showing a student looking off to the side or using a cell phone, indicating that someone is cheating. To prevent students from cheating during remote examinations, educational institutions could implement this technology. As it is extremely difficult to oversee exams in an online setting, this proposed technique provides an additional level of confidence to the online education systems. If teachers can detect when their students' eyes dart away from the test, they will have more evidence to back up their suspicions of cheating. In addition to eye-tracking, the combination of multiple items, such as gaze direction, object recognition and head movement, gives a lengthy view of the behaviour during the examination process with help of which examiners may ensure that learners are behaving accordingly. It is also paramount to bear fairness and to find a proper balance in this whole process, and these technologies can also frighten innocent candidates. The outstanding aspect of fold 1 lessens the amount of video proctoring required.

The system proposed had a state of the art features such as the ability to track eye movement, face features and identification of objects to keep a check on the people taking the test. The findings have been very intriguing, since it is revealed that different types of abnormalities can be detected by the system such as off screen gazes, cell phone use and abnormal face movement. This system is evaluated using a sample of 350 students whose results about the exam are stored. In order to test the proposed model in several conditions, the second fold is analysed with some extra pieces of footage. Results indicate that the system detects correctly the desired anomalies such as the eye movement tracking, speech, and cell phone detection. The system was found to have an accuracy of 87.8%, as well as exhibited efficiency, computers load reduction, and scalability, thus contributing to the work on remote education sustainability through less infrastructure and travel needs. The results of the binomial significance test demonstrate that the accuracy of the proposed hybrid proctoring has been statistically significant ($p \leq 0.05$) in combination. It is worth noting that the recommended system helps in the SDG 4 inclusive objectives through fairness and justice in online education opportunities. The system was fairly calculated, had a processing time of about 4.4ms per frame, a CPU usage of 30 percent to 40 percent frame-throughput, and a memory usage of approximately 2000MB for the first 20 frame, and constant thereafter.

There are hard policy implications to the study of online education that are well applied and particular to universities, MOOCs, and professional certification agencies that are looking to find reliable but ethical online assessment tools. By increasing the accuracy, transparency, and interpretability of the high-stakes tests when revealing discrepancies, the system would increase the credibility and responsibility of the system.

The small database and availability, the lack of connectivity in the rural regions, the gap in the realm of technological resources, and incompatibility on cross-platform platforms are the limitations of this study. Such issues provide an incentive to consider sustainable solutions, which, again, are aligned with SDG 9 (Industry, Innovation, and Infrastructure) and SDG 10 (Reduced Inequalities) to increase the availability of

digital and provide equal educational opportunities to everyone. In future, the research is planned to be validated by in-depth trials in various contestants and institutions, associate it with current educational web resources, enhance detection accuracy, make it supportive of the emotional wellbeing of students, and instead of detecting the behavior change and consider a pattern of honesty in the education field. Moreover, the adoption of bias reduction in anomaly detection, improved fairness and privacy protection, adaptive assessment, behavioral pattern detection, and blockchain-based credentialing should be considered as the focus of the future work. Such practices will make sure that AI-based proctoring does not only maintain the academic integrity but is helping to achieve the educational goals of equity, transparency, affordability, scalability, and sustainability.

Conflict of Interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgments

This research did not receive any specific grant from funding agencies in the public, commercial, or not-for-profit sectors. The authors would like to thank the Editor-in-Chief, section editors, and anonymous reviewers for their comments and suggestions that helped to improve the quality of this work. Informed consent was obtained from all parents and/or legal guardians (if participants are under 18) involved in the study. Participants were informed and provided their consent prior to participating in this study. The dataset includes candidates' marks from a custom dataset of 350 students and video recordings. No personal data was collected, and no identifiable information is included or published. All results are fully anonymised, and no data can be traced back to any individual participant. Confidentiality and anonymity have been strictly maintained throughout the study to protect participant privacy. The datasets which are generated during and/or analysed during the current study are all available from the corresponding author on the reasonable request.

AI Disclosure

The author(s) declare that no assistance is taken from generative AI to write this article.

References

Adoga, P.I. (2023). Framework for design of security systems for monitoring examinees and proctors during external offline examinations in Nigeria. *FUDMA Journal of Sciences*, *7*(6), 12-17. https://doi.org/10.33003/fjs-2023-0706-2007.

Ahmad, I., AlQurashi, F., Abozinadah, E., & Mehmood, R. (2021). A novel deep learning-based online proctoring system using face recognition, eye blinking, and object detection techniques. *International Journal of Advanced Computer Science and Applications*, *12*(10), 847-854.

Al-Airaji, R.M., Aljazaery, I.A., Alrikabi, H.T.S., & Alaidi, A.H.M. (2022). Automated cheating detection based on video surveillance in the examination classes. *iJIM*, *16*(08), 124-137. https://doi.org/10.3991/ijim.v16i08.30157.

Alguacil, M., Herranz-Zarzoso, N., Pernías, J.C., & Sabater-Grande, G. (2024). Academic dishonesty and monitoring in online exams: a randomized field experiment. *Journal of Computing in Higher Education*, *36*(3), 835-851. https://doi.org/10.1007/s12528-023-09378-x.

Allen, I.E., & Seaman, J. (2015). *Grade level: tracking online education in the United States*. Babson survey Research Group. Babson College, 231 Forest Street, Babson Park, MA 02457.

Anohina-Naumeca, A., Birzniece, I., & Odiņeca, T. (2020). Students' awareness of the academic integrity policy at a Latvian university. *International Journal for Educational Integrity*, *16*(1), 12. https://doi.org/10.1007/s40979-020-00064-4.

Arianti, A.S., Putra, M.A.S., & Nugraha, E. (2023). Designing an online examination system with object detection-based proctoring. *Computing and Education Technology Journal*, *3*(2), 1-12.

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.

Aurelia, S., Thanuja, R., Chowdhury, S., & Hu, Y.C. (2024). Retracted article: AI-based online proctoring: a review of the state-of-the-art techniques and open challenges. *Multimedia Tools and Applications*, *83*(11), 31805-31827.

Beck, V. (2014). Testing a model to predict online cheating—Much ado about nothing. *Active Learning in Higher Education*, *15*(1), 65-75.

Berkey, D., & Halfond, J. (2015). *Cheating, student authentication and proctoring in online programs*. New England Journal of Higher Education. Boston, MA.

Bilen, E., & Matros, A. (2021). Online cheating amid COVID-19. *Journal of Economic Behavior & Organization*, *182*, 196-211.

Bora, J., Dehingia, S., Boruah, A., Chetia, A.A., & Gogoi, D. (2023). Real-time assamese sign language recognition using mediapipe and deep learning. *Procedia Computer Science*, *218*, 1384-1393. https://doi.org/10.1016/j.procs.2023.01.117.

Chen, C., Long, J., Liu, J., Wang, Z., Wang, L., & Zhang, J. (2020). Online academic dishonesty of college students: a review. In *2020 International Conference on Advanced Education, Management and Social Science* (pp. 156-161). Atlantis Press. Hangzhou, China.

Cluskey Jr, G.R., Ehlen, C.R., & Raiborn, M.H. (2011). Thwarting online exam cheating without proctor supervision. *Journal of Academic and Business Ethics*, *4*, 1-7.

Corrigan-Gibbs, H., Gupta, N., Northcutt, C., Cutrell, E., & Thies, W. (2015). Deterring cheating in online environments. *ACM Transactions on Computer-Human Interaction*, *22*(6), 1-23.

Dadak, A., Uyan, U., & Öztürk, M.U. (2022). Real time cheating detection pipeline for online exams. In *2022 30th Signal Processing and Communications Applications Conference* (pp. 1-4). IEEE. Safranbolu, Turkey.

Dombrowski, Q., Gniady, T., & Kloster, D. (2023). Introdução ao Jupyter notebook. *The Programming Historian em Português*, *3*. https://doi.org/10.46430/phpt0043.

Ege, M., & Ceyhan, M. (2023). Talent-interview: web-client cheating detection for online exams. *arXiv preprint arXiv:2312.00795*.

Etter, S., Cramer, J.J., & Finn, S. (2007). Origins of academic dishonesty: ethical orientations and personality factors associated with attitudes about cheating with information technology. *Journal of Research on Technology in Education*, *39*(2), 133-155.

Foody, G.M. (2023). Challenges in the real world use of classification accuracy metrics: from recall and precision to the Matthews correlation coefficient. *Plos One*, *18*(10), e0291908. https://doi.org/10.1371/journal.pone.0291908.

Foody, G.M. (2024). Ground truth in classification accuracy assessment: myth and reality. *Geomatics*, *4*(1), 81-90. https://doi.org/10.3390/geomatics4010005.

Grijalva, T.C., Kerkvliet, J., & Nowell, C. (2006). Academic honesty and online courses. *College Student Journal*, *40*(1), 180-185.

Guo, P., Feng Yu, H., & Yao, Q. (2008). The research and application of online examination and monitoring system. In *2008 IEEE International Symposium on IT in Medicine and Education* (pp. 497-502). IEEE. Xiamen.

Holden, O.L., Norris, M.E., & Kuhlmeier, V.A. (2021). Academic integrity in online assessment: a research review. In *Frontiers in Education* (Vol. 6, p. 639814). Frontiers Media SA. https://doi.org/10.3389/feduc.2021.639814.

Huang, S.C., & Le, T.H. (2021). Advanced TensorBoard. In *Principles and labs for deep learning* (pp. 169-200). Academic Press, London. https://doi.org/10.1016/B978-0-323-90198-7.00002-1.

Imah, E.M., Puspitasari, R.D.I., Annisa, F.Q., & Al Habib, H. (2023). A comparative study of deep transfer learning algorithm for cheating detection in the exam based on surveillance camera recording. In *Proceedings of the 8th International Conference on Sustainable Information Engineering and Technology* (pp. 259-265). *Badung, Bali, Indonesia.* https://doi.org/10.1145/3626641.36269.

Jierula, A., Wang, S., Oh, T.M., & Wang, P. (2021). Study on accuracy metrics for evaluating the predictions of damage locations in deep piles using artificial neural networks with acoustic emission data. *Applied Sciences*, *11*(5), 2314. https://doi.org/10.3390/app11052314.

Joseph, F.J.J., Nonsiri, S., Monsakul, A. (2021). Keras and TensorFlow: a hands-on experience. In: Prakash, K.B., Kannan, R., Alexander, S., Kanagachidambaresan, G.R. (eds) *Advanced Deep Learning for Engineers and Scientists*. EAI/Springer Innovations in Communication and Computing. Springer, Cham. https://doi.org/10.1007/978-3-030-66519-7_4.

Karthika, R., Vijayakumar, P., & Bharat, S.R. (2019). Secure online examination system for e-learning. In *2019 IEEE Canadian Conference of Electrical and Computer Engineering* (pp. 1-4). IEEE. Edmonton, AB, Canada.

King, D.L., & Case, C.J. (2014). E-cheating: incidence and trends among college students. *Issues in Information Systems*, *15*(1), 20-27.

Kulkarni, A.V., Rathore, B.P., Singh, S.K., & Bahuguna, I.M. (2011). Understanding changes in the Himalayan cryosphere using remote sensing techniques. *International Journal of Remote Sensing*, *32*(3), 601-615.

Mohammadkarimi, E. (2023). Teachers' reflections on academic dishonesty in EFL students' writings in the era of artificial intelligence. *Journal of Applied Learning and Teaching*, *6*(2), 105-113. https://doi.org/10.37074/jalt.2023.6.2.10.

Ngo, D.A., Nguyen, T.D., Dang, T.L.C., Le, H.H., Ho, T.B., Nguyen, V.T.K., & Nguyen, T.T.H. (2024). Examining monitoring system: detecting abnormal behavior in online examinations. *arXiv preprint arXiv:2402.12179*.

Nigam, A., Pasricha, R., Singh, T., & Churi, P. (2021). A systematic review on AI-based proctoring systems: past, present and future. *Education and Information Technologies*, *26*(5), 6421-6445. https://doi.org/10.1007/s10639-021-10597-x.

Noorbehbahani, F., Mohammadi, A., & Aminazadeh, M. (2022). A systematic review of research on cheating in online exams from 2010 to 2021. *Education and Information Technologies*, *27*(6), 8413-8460. https://doi.org/10.1007/s10639-022-10927-7.

Nurpeisova, A., Shaushenova, A., Mutalova, Z., Ongarbayeva, M., Niyazbekova, S., Bekenova, A., & Zhumasseitova, S. (2023). Research on the development of a proctoring system for conducting online exams in Kazakhstan. *Computation*, *11*(6), 120. https://doi.org/10.3390/computation11060120.

Park, C. (2017). In other (people's) words: plagiarism by university students—literature and lessons. In: Barrow, R. (ed) *Academic Ethics* (pp. 525-542). Routledge, London. https://doi.org/10.4324/9781315263465.

Rane, N.L., Paramesha, M., & Desai, P. (2024). Artificial intelligence, ChatGPT, and the new cheating dilemma: strategies for academic integrity. *Artificial Intelligence and Industry in Society*, *5*, 2-2.

Rosen, W.A., & Carr, M.E. (2013). An autonomous articulating desktop robot for proctoring remote online examinations. In *2013 IEEE Frontiers in Education Conference* (pp. 1935-1939). IEEE. Oklahoma City, USA.

Surahman, E., & Wang, T.H. (2022). Academic dishonesty and trustworthy assessment in online learning: a systematic literature review. *Journal of Computer Assisted Learning*, *38*(6), 1535-1553.

Wan, Z., Li, X., Xia, B., & Luo, Z. (2021). Recognition of cheating behavior in examination room based on deep learning. In *2021 International Conference on Computer Engineering and Application* (pp. 204-208). IEEE. Kunming, China.

Yang, Y., & Fan, F. (2023). Ancient thangka Buddha face recognition based on the Dlib machine learning library and comparison with secular aesthetics. *Heritage Science*, *11*(1), 137. https://doi.org/10.1186/s40494-023-00983-8.

Yarlagadda, S.S., Tule, S.H., & Myada, K. (2024). F1 score based weighted asynchronous federated learning. *International Journal for Research in Applied Science and Engineering Technology*, *12*(2), 947-953. https://doi.org/10.22214/ijraset.2024.58487.

Zelinsky, A. (2009). Learning openCV---computer vision with the OpenCV library (Bradski, GR et al.; 2008)[On the Shelf]. *IEEE Robotics & Automation Magazine*, *16*(3), 100-100. https://doi.org/10.1109/MRA.2009.933612.