WWW.IJRPR.COM

International Journal of Research Publication and Reviews

Journal homepage: www.ijrpr.com ISSN 2582-7421

Offline Exam Proctoring Using Pose Estimation Algorithm

[1]V. Anji Reddy, [2]K. N. B. Chandhrika, [3]K. Sai Praneeth, [4]L. Ganesh, [5]K. Bhanu Teja

[1]Assistant Professor, Department Of CSE, Lendi Institute of Engineering & Technology,
[2,3,4,5]Student – 8th Semester Department of Computer Science & Engineering
[2,3,4,5]Lendi Institute of Engineering and Technology, Jonnada, Vizianagaram
[1]chandhrikakarri@gmail.com, [2]chandhrikakarri@gmail.com, [3]praneethkontham@gmail.com, [4]ganeganesh375@gmail.com, [5]kollibhanut@gmail.com

ABSTRACT:

Examination is a key method of assessing any form of knowledge held by an individual. Examinees frequently cheat during exams by engaging in unethical activities. Examinees may display different expressions during misconduct than they might in other circumstances. In this project, we will use emotional analysis to identify the candidates who are malpractice in the examination hall. The system utilizes the webcam to collect the student's image, identify landmarks with MediaPipe, and determine the student's face angle with PnP. This data is then analyzed to spot any suspicious activity, like glancing away from the screen or reviewing notes. The system can also take screenshots to be reviewed later. Our project will help with more efficient monitoring of students compared to traditional proctoring methods. The proposed algorithm carries the best accuracy, specificity, and sensitivity of 83.8, 78.6, and 85.3 respectively compared to algorithms like YOLOv3, LSVM, and KNN.

Keywords: *Sentimental Analysis, Pose Estimation algorithm, MediaPipe, Face recognition*

1. INTRODUCTION

The Fourth Industrial Revolution, or Industry 4.0, emphasizes automation and computerization. These goals are achieved through the fusion of numerous physical and digital technologies, including sensors, embedded systems, Artificial Intelligence (AI), Cloud Computing, Big Data, and the Internet of Things. Digital technology has become increasingly interconnected.

Exams are now administered online using online proctoring tools as discussed in [11]. However, offline written exams are still held in many colleges where the presence of a teacher is required. Supervisors in examination rooms have a difficult time monitoring students' activity. It is an expensive and time-consuming method. Furthermore, the exams for various schools and colleges face the problem of manually proctoring the students, which fails to ensure the continuous integrity and security aspects of the examination, and a human cannot proctor at large scales effectively. Automatic exam activities recognition is therefore a necessitating and a demanding field of research as discussed in [12]. The proposed system automates the proctoring process at scale with the aid of computers and reduces the load on human proctors. In this work, we propose an integrated exam control solution, focused on surveillance, that offers an automatic categorization of the behavior and activities of students during exams. The suggested approach is related to offline exam activity recognition (EAR) which incorporates the body movements of the exam's participants across tests for the recognition of student acts. Six different actions regarding students' actions during exams are considered for classification. Such actions involve: (a) seeing back, (b) watching towards the front, (c) normal performing, (d) passing gestures to other fellows, (e) watching towards left or right, and other suspicious actions.

2. LITERATURE SURVEY

A. W. Muzaffar, M. Tahir, M. W. Anwar, Q. Chaudry, S. R. Mir and Y. Rasheed [1]in their work titled with "A Systematic Literature Review (SLR) of online examination in the context of e-learning" in higher education. The review selects and analyses 53 studies published in the last five years, identifying the leading features, development approaches, techniques/algorithms, datasets, online exams tools, and existing tools used in the selected studies. The article also in their work they investigates the participation of countries in online exam research and identifies key factors for the global adoption of online exams, which can help in selecting the right online exam system for a particular country based on existing e-learning infrastructure and overall cost. The findings of the study provide valuable insights for researchers and practitioners in selecting appropriate features, development approaches, tools, and techniques for implementing online exams solutions. Overall, this article serves as a literature survey of online examination in e-learning in higher education, summarizing and analysing the latest developments in the field.

F. Schroff, D. Kalenichenko and J. Philbin [2]in their work, titled with "FaceNet: A unified embedding for face recognition and clustering" a new system built for efficient face identification and verification at scale. FaceNet learns a mapping from face images to a compact Euclidean space in which distances correspond directly to a measure of facial similarity. This makes it simple to accomplish tasks like face recognition, verification, and clustering using standard techniques using FaceNet embeddings as feature vectors.

Abdul Wahid, Yasushi Sengoku, and Masahiro Mambo [3]in their work: With the rapid development of technology, secure online examination systems have become a hot topic in the educational world as discussed in These systems need to address computer and network security issues and prevent cheating by examinees. Various techniques have been proposed to enhance security, such as image processing, face recognition, keystroke dynamics, eye tracking, cryptography, and steganography. Researchers have developed different approaches to prevent cheating, such as randomizing questions and answers, using timers, monitoring examinee activities and restricting access to external devices and applications during the examination. Overall, secure online examination systems can provide a convenient and efficient way to conduct exams while maintaining high security and integrity. FaceNet' s fundamental novelty is that it directly optimizes the embedding itself using a deep convolutional network, rather than an intermediate bottleneck layer as in other deep learning systems. The system is trained using triplets of aligned matching/non-matching face patches generated by an innovative online triplet mining algorithm.

Y. Atoum, L. Chen, A. X. Liu, S. D. H. Hsu and X. Liu[4] and González-González, C.S.; Infante-Moro, A.; Infante-Moro, J.C.[5] in their work, titled with Implementation of E-Proctoring in Online Teaching offers a multimedia analytics system for automatic online exam proctoring that uses one webcam, one webcam, and a microphone to monitor the visual and audio surroundings of the testing venue. The system consists of six core components that estimate essential behavior indications in real time, including user authentication, text detection, voice detection, active window detection, gaze estimation, and phone detection. By combining these components and employing a temporal sliding window, the system can identify cheating at any point throughout the exam. The article emphasizes the significance of efficient online exam proctoring in the context of remote education's growing popularity and reach, particularly massive open online courses.

W. J. Bowers [6] looks at academic dishonesty in higher education institutions. The author emphasizes the importance of higher education institutions in educating citizens for a wide range of societal concerns, and academic dishonesty damages these institutions' legitimacy. Academic dishonesty has reached frightening levels in Ethiopia, according to the author, who explores how modern technology have made it simpler for students to cheat. The paper advocates investigating the difficulties of academic dishonesty in higher education and the far-reaching ramifications for corruption. The author contends that systematic and thorough measures are required to promote integrity and avoid dishonesty, particularly considering technological innovation.

M. Korman [9]in their work, titled with "Behavioral detection of cheating in online examination" seeks to investigate the feasibility of identifying online examination cheating by analysing the dynamics of human-computer interaction. With the increased usage of online or computer-based exams, there is growing worried about the possibility of cheating. This work presents a way of detecting computer-based examination cheating based on behavioral metrics and machine learning, which is related to the broader concept of academic dishonesty. This method's detection potential is primarily based on cue leakage theory, which may be investigated utilizing pattern recognition and anomaly detection theory using a behavioral biometrics approach. The study's goal is to create a systematic and all-encompassing method to promoting integrity and preventing dishonesty in online exams.

S. Dendir and R. S. Maxwell [13] in their work, titled with "Cheating in online courses: Evidence from online proctoring" using online proctoring, which entails using camera recording software, can successfully prevent academic dishonesty in online courses. The study employed a quasi-experimental approach, with two online courses chosen and online proctoring implemented for high-stakes tests in both courses. Before and after the introduction of online proctoring, the courses had the same material, organization, and examinations. Exam score analysis found that the implementation of online proctoring was associated with a drop in average performance in both courses. The findings also indicate that cheating occurred in online courses prior to the implementation of online proctoring. Additionally, the study found that the explanatory power of the regression models was higher for scores under proctoring, indicating that the use of online proctoring can help to identify and prevent cheating.

X. Li, K.-M. Chang, Y. Yuan, and A. Hauptmann [14] in their work, titled "Massive open online proctor: Protecting the credibility of MOOCs certificates" addresses the problem of cheating in MOOCs. To detect and prevent cheating in online assessments, the authors propose combining automatic and collaborative approaches. Online learning is gaining popularity in both formal educational settings and individual development, and online exams are increasingly being recognized as an efficient method of assessment. However, there is a lack of research on learners' perceptions of online exams in developing countries, which could contribute to the effective use of online exams in these contexts. Therefore, this study aimed to investigate students' perceptions of online exams at a state university in Turkey and another in Kyrgyzstan, and to compare the results.

G. A. Adanır, R. İsmailova, A. Omuraliev, and G. Muhametjanova [15]in their work, titled with "Learner's perceptions of online exams" employed a mixed method approach and included 370 undergraduate students enrolled in first-year online courses. A survey was used to collect quantitative data, while an open-ended question was used to acquire qualitative data. The findings revealed that learners' opinions differed depending on gender, major, and prior online course experience. Turkish students perceived online examinations to be less stressful, more dependable, and fairer than traditional paper-based exams, however Kyrgyz students did not. The study provides useful information for institutions in both nations as they plan for the efficient implementation of online tests.

3. METHODOLOGY

3.1 Introduction

Systems design is a critical stage in the development of any complex system, and it requires careful planning and coordination between various stakeholders. The goal of systems design is to create a system that can meet the specific needs and requirements of the business or organization while also being dependable, efficient, and cost-effective.

To achieve this goal, systems designers must consider a range of factors, including the technical requirements of the system, the organizational structure of the business or organization, the available resources, and the potential risks and challenges associated with the project. They must also be aware of emerging technologies and industry trends that may impact the design and development of the system.

The systems design process typically involves several stages, including requirements analysis, architectural design, module design, interface design, and data design. During each stage, designers must work closely with other members of the development team to ensure that the system is being designed in a way that meets the overall goals of the project.

One of the key challenges of systems design is managing complexity. As systems become more complex, it becomes increasingly difficult to ensure that all the components are working together effectively. This is where systems thinking and systems theory come into play, providing designers with the tools and frameworks needed to manage complexity and ensure that the system is working as intended.

Overall, systems design is a critical part of the development process for any complex system, and it requires a combination of technical expertise, organizational knowledge, and strategic thinking to ensure that the system is designed to meet the specific needs and requirements of the business or organization.

Software, right down to the data and how it travels and transforms throughout its travel through the system. Systems design then overlaps with systems analysis, systems engineering and systems architecture.

3.2. System Architecture

A system architecture is a conceptual model that defines the structure, behavior, and views of a system. It can consist of system components and the sub-systems developed, that will work together to implement the overall system. It primarily concentrates on the internal interfaces among the system's components or subsystems and on the interface(s) between the system and its external environment, especially the user.

System architecture consists of the following components:

• Video Sequence Input:

In this to input a video sequence in Python, we used the OpenCV library. We first open the video file using cv2.VideoCapture(). We then loop through each frame in the video using a while loop, reading each frame using cap.read(). This input is further used to process images of each student to estimate their cheating behaviour. The following step involves detecting multiple face locations in the frame.

• Detect multiple faces:

In this input the captured frame is used to detect multiple faces. Multiple faces are detected using face_recognition library in Python. It consists of face_ locations function which returns the face locations of each image found in the frame which can be used to extract each face and detect pose.

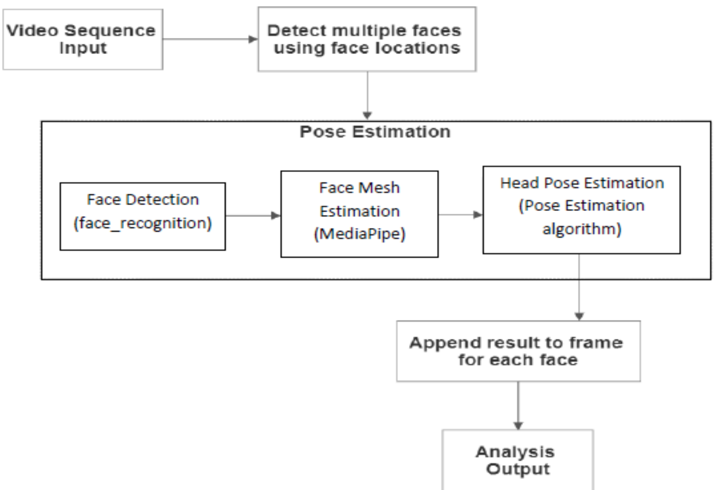


Fig 3.2.1 Architecture of proposed system

• Pose Estimation:

The best way the supervisors use to detect abnormal behavior is to check where the examinee is looking, we use the same concept in this project to detect the direction/angle of the head of the examinee. The major challenge is the computer vision algorithm used to detect the face using the images data as well as the direction of the head, such algorithms usually require large amounts of training data as well as discrete graphic hardware to work properly.

We tackle this problem by using third party open-source algorithms from MediaPipe library by google which is designed to work on low end devices with accuracy at par. The MediaPipe library provides accurate face landmarks, these face landmarks are then processed by the Perspective-n-Points algorithm implemented with help of an open computer vision library to provide the 3D orientation of the user's head.

• Cheating Behavior Detection:

This is the ultimate prediction algorithm that provides the result of the analysis. This is detected by applying a threshold over the head angle. If the user exceeds the angle, then the two flags are used to store the head pose which are named x-axis and y-axis. Then the flags of x-axis, y-axis and the previous output are fed into a probabilistic conditional algorithm to predict if the user is showing suspicious behavior in terms of percentage of surety. The detected pose of each face is appended to the output frame and displayed to the examiner.

3.3. Face_recognition

Face recognition is a computer technology that identifies or verifies the identity of a person from a digital image or a video frame. The face recognition algorithm uses machine learning techniques to detect and recognize faces in an image or a video. One of the most popular face recognition algorithms is the face_recognition library, which is a Python package that allows developers to build applications that can recognize and manipulate faces from images and videos.

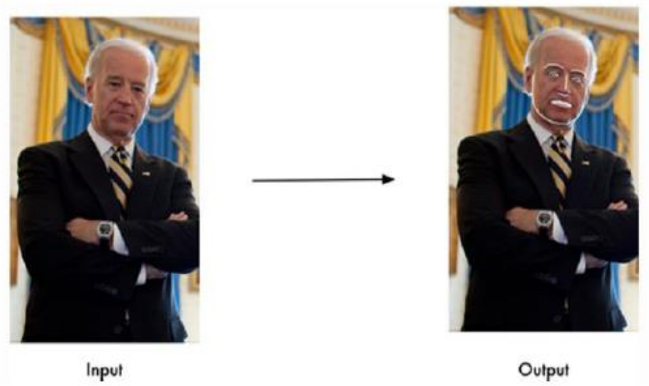


Fig. 3.3.1 Face Recognition

The face_recognition algorithm uses deep learning models to analyze facial features and match them to a database of known faces. It first detects faces in an image using a pre-trained model, and then encodes each face as a feature vector. This encoding captures the unique characteristics of the face, such as the location of key facial landmarks and the distribution of colors in the face.

Once the face is encoded, the algorithm compares it to a database of known faces to identify a match. If the face matches a known identity, the algorithm can provide information about the person, such as their name, age, or gender. The algorithm can also be trained to recognize multiple faces in a single image or video frame, and to track faces across multiple frames.

3.4. MediaPipe Face Mesh Estimation

MediaPipe is a cross-platform framework developed by Google for building multimodal machine learning pipelines. It provides pre-built components and tools for creating complex pipelines that can process data from various sources such as cameras, microphones, and other sensors. MediaPipe is designed to be used by developers and researchers who want to build real-time applications that can handle a wide range of input data types, including images, video, and audio.

One of the key features of MediaPipe is its ability to handle complex data processing tasks in real-time. It uses a combination of deep learning and traditional computer vision techniques to analyze input data and extract useful information. This makes it well-suited for applications such as augmented reality, virtual reality, and robotics, which require real-time processing of multiple data streams.

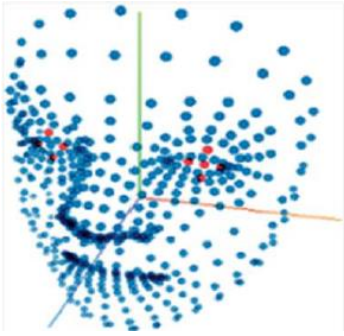


Fig. 3.4.1Face Mesh Estimation

Face mesh estimation is one of the pre-built machine learning models provided by MediaPipe. It uses deep learning techniques to estimate a 3D mesh of facial landmarks on a 2D image or video stream in real-time. The estimated mesh can be used for a variety of applications, including augmented reality, virtual try-on, and facial expression analysis. The estimated face mesh consists of a set of vertices representing the positions of key facial landmarks, such as the corners of the eyes, nose, and mouth. The mesh is dense and can accurately capture the shape and movements of the face in real-time.

Pseudocode for MediaPipe face landmark identification:

Input: A video stream that includes one or more faces.

Output: A video stream displaying the facial landmark points for each detected face on the screen.

1. Begin

2. Load the MediaPipe Face Mesh Estimation model and settings file to get started.

3. Set up the camera or video input stream.

4. Read the frame and pre-process it for input to the model

5. for each frame in the input stream do

6. Invoke the model with the pre-processed frame.

7. Retrieve the model's projected facial landmark coordinates.

8. for each face in the frame do

9. Draw the facial landmarks on the frame.

10. Show the rendered frame on screen or save it to a file.

11. When the input stream is terminated, all resources should be released.

12. End.

In addition to the face mesh estimation model, MediaPipe also provides pre-built models for face detection, face landmark detection, and face recognition. These models can be combined with the face mesh estimation model to build powerful facial analysis pipelines for a variety of applications.

In summary, MediaPipe is a powerful and flexible framework for building multimodal machine-learning pipelines that can process real-time data from a variety of sources. Its pre-built components and machine learning models make it easy to get started, while its flexibility and customizability make it suitable for a wide range of applications.

3.5. PnP (Perspective-n-Point) Algorithm

It is a computer vision algorithm used to solve the problem of finding the position and orientation of a camera relative to a 3D object in space. This problem is also known as the camera pose estimation problem.

The PnP algorithm uses a set of 3D points in the object's coordinate system and their corresponding 2D projections in the camera's image plane to estimate the camera pose. The algorithm assumes that the camera's intrinsic parameters (focal length, principal point, and distortion coefficients) are known beforehand.

There are several variants of the PnP algorithm, but the most widely used one is the PnP problem with known correspondences, which assumes that the 2D-3D point correspondences are known beforehand. This problem can be solved using various methods, including the Direct Linear Transformation (DLT) algorithm, Levenberg-Marquardt optimization, and the RANSAC (Random Sample Consensus) algorithm.

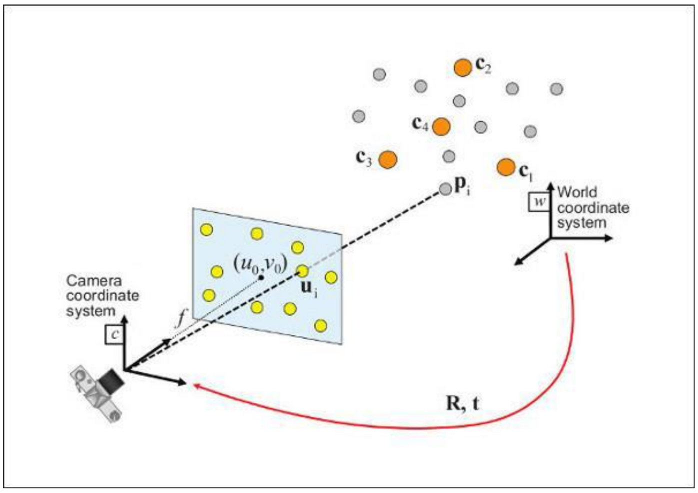


Fig. 3.5.1 PnP (Perspective-n-Point) pose computation.

The PNP problem is a widely studied problem in computer vision, and there are various approaches and algorithms available for solving it, such as the Direct Linear Transformation (DLT) method, the Perspective-n-Point with RANSAC (Random Sample Consensus) method, and different variants of the Iterative Closest Point (ICP) algorithm. These methods can be used to estimate the face angle or pose accurately and robustly, and can have applications in facial recognition, human-computer interaction, virtual reality, and augmented reality, among others.

Algorithm for estimating face angle using the Perspective-n-Point (PNP) problem:

Input: Images with 3D and 2D coordinates of the same landmarks along with intrinsic camera parameters.

Output: Estimated 3D rotation and translation of the face with respect to the camera.

1. Begin:

2. Convert 2D landmarks to homogeneous coordinates: For each 2D landmark (x, y), convert it to a 3D homogeneous coordinate X_homo = [x, y, 1].

3. Estimate pose: Use a PnP algorithm to estimate the rotation and translation of the face. This can be done using the solvePnP function from a library like OpenCV. If prior knowledge of the face pose is available, use it as an initial guess for the rotation and translation vectors. Otherwise, run the algorithm with just the 3D and 2D landmarks, and the camera intrinsic parameters. The output of this step is the rotation vector and translation vector.

4. Convert rotation vector to rotation matrix: Use the Rodrigues function from OpenCV to convert the rotation vector (rvec) to a rotation matrix (R).

5. Convert rotation matrix to Euler angles: Use the atan2 function to compute the three Euler angles theta_x, theta_y, and theta_z from the rotation matrix (R). The order of rotations should be specified beforehand, for example, if we want to rotate first around the X axis, then around the Y axis, and finally around the Z axis.

6. Return results: Return the three Euler angles theta_x, theta_y, theta_z and the translation vector as the estimated 3D rotation and translation of the face with respect to the camera.

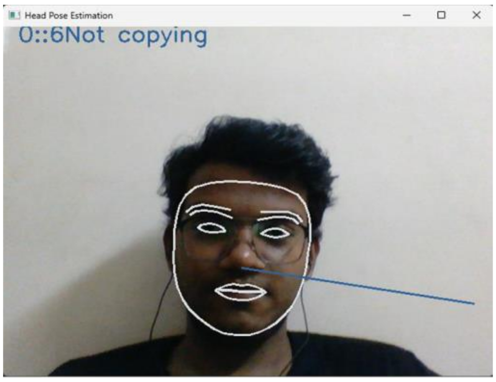
7. End

The PnP algorithm is widely used in applications such as robotics, augmented reality, and object tracking, where it is necessary to estimate the pose of an object in real time.

4. RESULTS

The project aimed to create an offline exam proctoring system, which involved conducting experiments with different facial recognition and angle detection methods. Initially, the dlib library was used for 2D facial landmark recognition but failed to predict the angle of the face accurately. Consequently, MediaPipe was adopted for face recognition. The pnp method was chosen to determine the angle required for proctoring, followed by additional mathematical steps. However, this approach resulted in false positive detections when students turned their heads to take something. To address this issue, a delay was introduced in the system's response. Overall, the project involved refining the system's facial recognition and angle detection capabilities to create an effective offline exam proctoring solution.

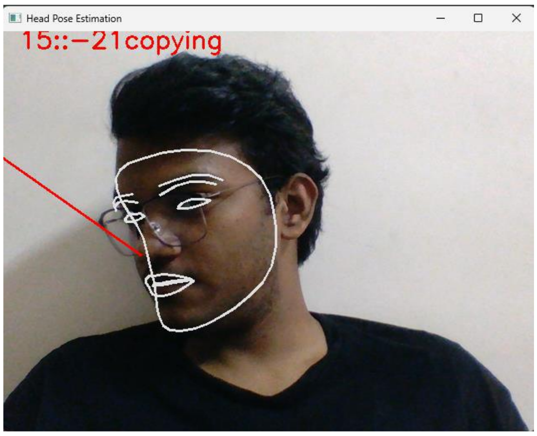
Case 1:



4.1.1 Single face recognition (Not copying)

The above Figure 4.1.1 shows the prediction of a person as not copying when he is looking straight at the screen. The blue text above the person's face indicates that the system has successfully recognized and identified the person's face. The angle of rotation horizontally is within the range of -10 and +10 degrees, and the vertical angle is less than 10 degrees, which means that the person is looking straight at the screen.

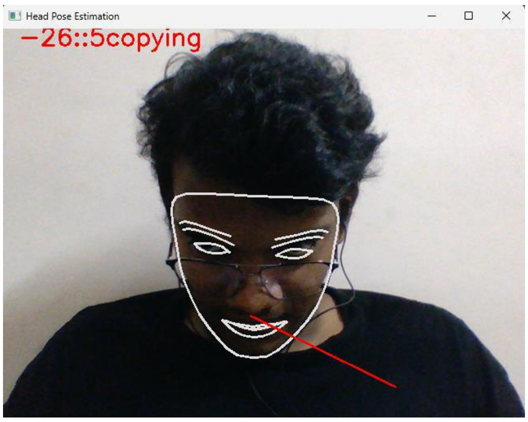
Case 2:



4.1.2 Looking Left (Copying)

The above Figure 4.1.2 shows the prediction of a person as copying when he is looking to the left. The red text above the person's face indicates that the system has identified the person's face and recognized that they have turned their head to the left by an angle of more than 10 degrees. The delay in their action is also more than 15 seconds, which indicates that they might be trying to copy information from the screen. Therefore, this behaviour is flagged as copying by the system.

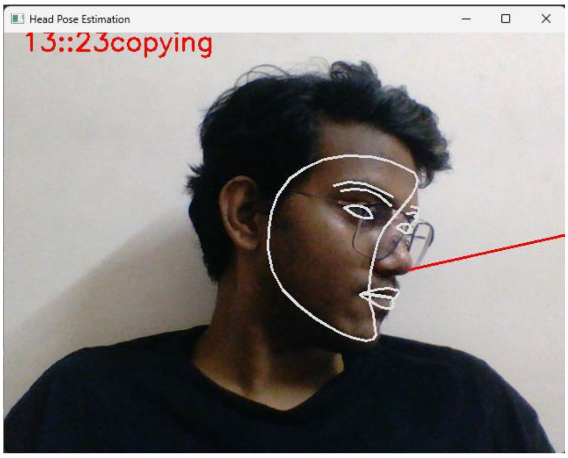
Case 3:



4.1.3 Looking Down (Copying)

The above Figure 4.1.3 shows the prediction of a person as copying when they are looking down. The red text above the person's face indicates that the system has identified the person's face and recognized that they are looking down with an angle of head rotation greater than 10 degrees. The delay in their action is also more than 15 seconds, indicating that they may be trying to copy information from the screen. Therefore, this behaviour is flagged as copying by the system.

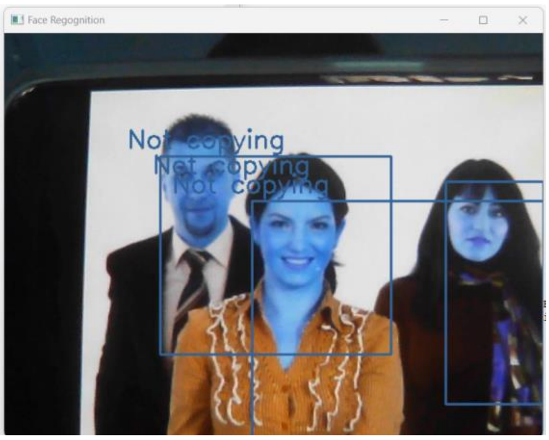
Case 4:



4.1.4 Looking Right (Copying)

The above Figure 4.1.4 shows the prediction of a person as copying when they are looking right. The red text above the person's face indicates that the system has identified the person's face and recognized that they have turned their head to the right by an angle greater than 10 degrees. Additionally, the delay in their action was more than 15 seconds, leading the system to flag this as a potential instance of copying. However, it is important to note that this is just one of several criteria used by the system to identify potential instances of copying, and other factors such as the person's gaze direction and head pose are also considered.

Case 5:



4.1.5 Multiple face recognition

In Figure 4.1.5, the results of multiple face recognition tests are presented, and it can be observed that in all cases where people are looking straight ahead, the model predicts that they are not copying. The green text displayed above the individuals' faces confirms that the system has successfully identified them, and their head poses are within the acceptable range, with horizontal angles between -10 and +10 degrees and vertical angles less than 10 degrees. This suggests that when individuals are facing the screen with their heads straight, they are engaged in legitimate exam-taking behaviour and not attempting to cheat by copying information from the screen.

5. PERFORMANCE EVALUATION

Experiment 1:

Suppose we have conducted an experiment that involves testing a binary classification model on 100 test cases. In this experiment, the model was evaluated using the TP, TN, FP, and FN metrics. The following are the performance metrics for the model:

• Accuracy: The model correctly classified 76.9% (77 out of 100) of all test cases.

• Sensitivity: The model correctly identified 73.3% (73 out of 100) of all positive cases.

• Specificity: The model correctly identified 81.8% (82 out of 100) of all negative cases.

Experiment 2:

Suppose we have conducted an experiment that involves testing a binary classification model on 200 test cases. In this experiment, the model was evaluated using the TP, TN, FP, and FN metrics. Following are the performance metrics for the model:

• Accuracy: The model correctly classified 83.769% (168 out of 200) of all test cases.

• Sensitivity: The model correctly identified 78.57% (148 out of 200) of all positive cases.

• Specificity: The model correctly identified 85.33% (170 out of 200) of all negative cases.

• Precision: The precision of model is 78.57%

Algorithms	Accuracy (%)	Precision (%)	Specificity (%)
YOLOv3	76.50	50.70	72.80
LSVM	58.15	24.27	57.07
KNN	81.20	48.63	78.93
PnP (proposed)	83.76	78.57	85.33

Table 5.1.1. Performance Evaluation between algorithms YOLOv3, LSVM, KNN and PnP respectively.

The below figures are the graphical representations of the above performance evaluation showing Accuracy, Precision, and Specificity comparison between algorithms YOLOv3, LSVM, KNN, and PnP respectively.

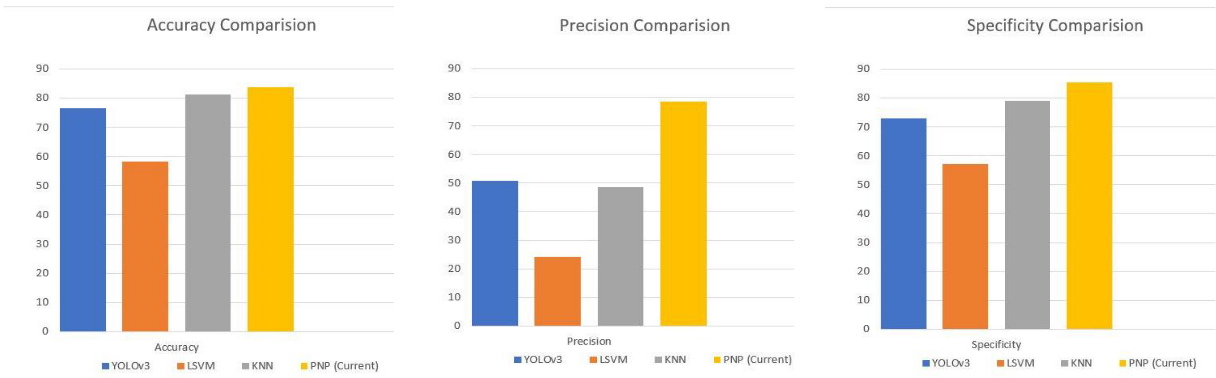


Fig.5.1.1. Precision Comparison between algorithms YOLOv3, LSVM, KNN, PnP respectively.

6. CONCLUSION

In this project has demonstrated the potential of using facial recognition technology to identify suspicious behaviour in examination halls. This technology can help to monitor students and detect any malpractice more efficiently and accurately. Additionally, it can provide valuable evidence that can be used in disciplinary proceedings. With further development, this technology could be further utilized to detect and prevent cheating in examinations, ensuring that assessment measures are fair for all examinees.

7. REFERENCES

[1] A. W. Muzaffar, M. Tahir, M. W. Anwar, Q. Chaudry, S. R. Mir and Y. Rasheed, "A Systematic Review of Online Exams Solutions in E-Learning: Techniques, Tools, and Global Adoption," in IEEE Access, vol. 9, pp. 32689-32712, 2021, doi: 10.1109/ACCESS.2021.3060192.

[2] F. Schroff, D. Kalenichenko and J. Philbin, "FaceNet: A unified embedding for face recognition and clustering," 2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR),2015, pp. 815-823, doi:10.1109/CVPR.2015.7298682.

[3] Abdul Wahid, Yasushi Sengoku, and Masahiro Mambo. 2015. Toward constructing a secure online examination system. In Proceedings of the 9th International Conference on Ubiquitous Information Management and Communication (IMCOM '15). Association for Computing Machinery, New York, NY, USA, Article 95, 1–8. https://doi.org/10.1145/2701126.2701203

[4] Y. Atoum, L. Chen, A. X. Liu, S. D. H. Hsu, and X. Liu, "Automated Online Exam Proctoring," in IEEE Transactions on Multimedia, vol. 19, no. 7, pp. 1609-1624, July 2017, doi:10.1109/TMM.2017.2656064.

[5] González-González, C.S.; Infante-Moro, A.; Infante-Moro, J.C. Implementation of E-Proctoring in Online Teaching: A Study about Motivational Factors. Sustainability 2020, 12, 3488.https://doi.org/10.3390/su12083488

[6] W. J. Bowers, Student Dishonesty, and Its Control in College. Bureau of Applied Social Research, Columbia University, 1964.

[7] C. M. Ghanem and N. A. Mozahem, ''A study of cheating beliefs, engagement, and perception—The case of business and engineering students,'' J. Academic Ethics, vol. 17, no. 3, pp. 291–312, Sep. 2019.

[8] S. W. Tabsh, H. A. El Kadi, and A. S. Abdelfatah, ''Faculty perception of engineering student cheating and effective measures to curb it,'' in Proc. IEEE Global Eng. Educ. Conf. (EDUCON), Apr. 2019, pp. 806–810.

[9] M. Korman, ''Behavioral detection of cheating in online examination,'' M.S. thesis, Lulea Univ. Technol., 2010.

[10] C. Voss and N. J. Haber, ''Systems and methods for detection of behavior correlated with outside distractions in examinations,'' Google Patent 9 892 315 B2, Feb. 13, 2018.

[11] K. Jalali and F. Noorbehbahani, ''An automatic method for cheating detection in online exams by processing the student's webcam images,'' in Proc. 3rd Conf. Elect. Comput. Eng. Technol. (E-Tech), Tehran, Iran, 2017, pp. 1–6.

[12] K. Butler-Henderson and J. Crawford, ''A systematic review of online examinations: A pedagogical innovation for scalable authentication and integrity,'' Comput. Educ., vol. 159, Dec. 2020, Art. no. 104024.

[13] S. Dendir and R. S. Maxwell, ''Cheating in online courses: Evidence from online proctoring,'' Comput. Hum. Behav. Rep., vol. 2, Aug. 2020, Art. no. 100033.

[14] X. Li, K.-M. Chang, Y. Yuan, and A. Hauptmann, ''Massive open online proctor: Protecting the credibility of MOOCs certificates,'' in Proc. 18th ACM Conf. Comput. Supported Cooperat. Work Social Comput., Feb. 2015, pp. 1129–1137.

[15] G. A. Adanır, R. İsmailova, A. Omuraliev, and G. Muhametjanova, ''Learners' perceptions of online exams: A comparative study in Turkey and Kyrgyzstan,'' Int. Rev. Res. Open Distrib. Learn., vol. 21, no. 3, pp. 1–17, 2020.