A Cheating Detection System in Online Examinations Based on the Analysis of Eye-Gaze and Head-Pose

Ambi Singh, Smita Das

{ambisaniya10@gmail.com, smitadas.nita@gmail.com}

NIT Agartala, India

Abstract. The devastating effect of COVID-19 ensued the closing of educational institutes throughout the world and led to a shift in the education mode from offline to online. Online exams became the new way of testing the academic knowledge of a student. The major concern which arises in an online exam is the apparent risk of malpractices emerging due to the remote invigilation. Therefore, cheating in online exams is widespread regardless of the levels of development. In this paper, our main goal is to propose a smart system which can automatically detect cheating in online exam. For this, the students' webcam is used to perceive his/her eye-gaze and head-pose. Based on the analysis of eye-gaze and head-pose, we have followed the students' intension to indulge in cheating.

Keywords: Online Exams, Machine Learning, Computer Vision, Eye-Gaze, Head-Poses, OpenCV, Python, Yolov3

1. Introduction

Exams are the rudimentary ways [1] of assessing students' learning and performance with respect to particular subjects. Exam results demonstrate [2] which part of the lesson each student remembers or takes a keen interest in. They allow higher education establishments to assess if students are able to deal with the demands at their future workplaces [3]. The traditional way of appearing for exams is to reach physically in an examination hall as per its schedule and attempt the answer in pen-and-paper mode in the presence of an invigilator [4]. Exams are usually conducted in a reasonable way where the invigilator can scold the student, warn and even punish him/her if found to be engaged in malpractices during the exam duration.

In recent times, information and communication technologies have witnessed rapid developments [5] and thus have directly affected human life. When the world was hit by COVID-19, social distancing came out to be the only precaution to this pandemic. Hence, the education sector [6] too rehabilitated drastically with immediate closing of schools, colleges and workplaces as well. A rise in e-learning took place along with virtual lectures, assessments and tutorials for students across the globe. Prior to COVID, online exams weren't much preferred in most places. Nevertheless, the sudden switching [7] from books to computers, pencils to keyboards, black-boards to PowerPoint presentations, question papers to online exams befitted the need of the hour. Sitting at home, completely safe from the infection risk of virus, attempting exams online is itself a boon [8] of technology. Students can

THEETAS 2022, April 16-17, Jabalpur, India

DOI 10.4108/eai.16-4-2022.2318165

use online exams to save time to pursue other interests alongside studies. The need for writing on paper is unnecessary in this online mode of exam and thus can save cost on paper and it is environmentally friendly [9] too.

1.1 Challenges with Online Exams

The major concern that arises in an online exam is the apparent risk of malpractices [10] that emerges due to the remote invigilation. The teacher can only have a frontal view of the student via her screen. The student can use a mobile phone to look out for answers [11] on the internet, use a book or her notes to refer to the solution [12] to a problem, making sure that her unethical activities are not captured on screen. Also, the student might let someone else answer the test [13] on their behalf or refer others for solutions.

Major reasons that motivate students to cheat include [14] the desire for better grades, fear of failure in examinations, lack of interest in studies and a very easy access to online information during the online exams. Most of them believe [15] that they would not be caught cheating, or the punishment would not be much severe if caught. Pressure from parents to do well in exams is also a major reason students prefer to cheat instead of honestly attempting their exam [16].

A student might also fear missing out or losing in the test due to common thinking that her fellow students would definitely ace the same test by cheating. As a result, cheating detection is necessary to maintain authenticity and integrity of the exam.

In Section 2, we have depicted the related works. Section 3 shows the proposed method followed by result analysis in section 4. Finally, the paper is concluded in section 5 with future direction of research work.

2. Related Works

In [17], a novel online proctoring system is proposed that uses HOG face detector and OpenCV face recognition algorithm. It is implemented as a software system using the FDDB and LFW datasets, thus achieving up to 97 % accuracy for face detection and 99.3% accuracy for face recognition. This system can detect gadgets like mobile phones too in an online exam.

In [18], a smart system has been proposed to detect cheating activities of a student attempting the exam in physical mode while in a real examination hall. With webcam installed in smart devices, placed on the student's desk, his suspicious behaviours are monitored. A real time video is captured and is then analysed to build a knowledge base for the system. This system can detect examinee's eye gaze and head poses to detect cheating.

In [19], the authors introduce the concepts of cheating in online exams and methods to control the same. Techniques have been proposed to detect and prevent the student from cheating. This is done through continuous authentication using an online proctor. Using visual C# and SQL server database, the system proposed authenticates the test-taker via a Fingerprint Reader, Eye Tribe Tracker. A test-taker is considered to be cheating or not cheating, considering two parameters based on screen time. The number of times, the test-taker moves out of the screen, i.e. moves away from the screen, measuring the total amount of time she does this.

In [20], to ensure a transparent, fair examination system, the authors developed a system based on students' eye movement and generating an alert message if cheating is detected. From the input video acquired, human faces are detected followed by analysis of eye pupils to find eye-movements. From the low-resolution images, eyes are detected using information from edges, pixel intensity. Viola Jones algorithm is used for human face detection, Canny Edge Algorithm for eye detection and Kalman Filtration algorithm for pupil detection.

In [21], the framework proposed for electronic-based invigilation of a computer-based exam performs authentication of the candidate using her fingerprints. Student's iris directional focus is monitored. A prescribed threshold is considered for the system. If it is exceeded by the gazing angle or the voice level, it will be presumed that the candidate is cheating, that she is communicating with someone. Implementation is done using Python, Java languages for programming. To communicate with the resources, JDBC is used, at the database level. MySQL provides the platform to create and manage the examination along with the authentication of databases.

In [22], the test-taker must have 2 cameras and a microphone that are used to capture videos as he is attempting the exam. Low-level features are extracted using 6 components which are processed in a temporal window to acquire high level features. This is an easy to use and a cheaper model for cheating detection in online exams. The test-taker is verified. Text, speech, phone and active-windows are detected along with eye-gazes. From 24 people who underwent the exam, a database is collected containing audio and visual data. From the different types of cheating behaviours analysed, Segment based detection rate was 87% at a fixed FAR of 2%.

3. Proposed Method

In this research, a pre-trained machine learning model of OpenCV, known as the Caffe Model is used to prevent cheating activities of a test-taker by detecting her face and facial landmarks, analysing her head poses along with eye tracking. For this, OpenCV will use input from the live webcam feed of the student's computer, converting that to several images. Logs are created after identifying various features and activities of the test-taker.

3.1 Detecting the Candidate's Face

One of the most fundamental aspects of Computer Vision is Face detection. From the captured image, face and key-points are detected, followed by extraction of major features. Bounding boxes on faces are usually drawn using pre-trained models. To detect the student's face, Open-CV's DNN type Model in this research. It is based on a Single Shot Multi-box detector (SSD). ResNet-10 architecture is the backbone. This is known as the Caffe Model, trained by using images from the web. It was included in OpenCV deep neural network module, post version 3.3. A quantized Tensor-flow version (8-bit) also exists but the floating point 16 version of the original Caffe implementation is used. It gives the fastest frames per second. Along with frontal faces, this can identify side faces also, works well with occlusion and quick head movements.

For finding faces in an image, a function is defined to which the model is given as input along with image in the form of a numpy array. An array is produced as the output that consists four coordinates of the face each corresponding to the four face corners. Using OpenCV, the image given as input is converted into blob format. Then, the presence of face is determined by a confidence/ probability value. A probability value>0.5 is considered as detection of face. The four coordinates around the face are estimated, which fed to a defined function outputs a rectangle.

3.2 Detecting the Candidate's Facial Landmarks

Facial landmarks will be used to localize and represent salient regions of the student's face, like her eyebrows, eyes, jawline, nose and mouth. To detect them, Dlib's 68 keypoints landmark predictor is commonly used as it produces great results in real-time scenarios.

A pre-trained model for facial detection provided by Yin Guobing, a Tensor-flow CNN model. It gives 68 landmarks that can be utilized in defining a face object. It is trained on 5 datasets.

Talking about TensorFlow, it is a python library which contains the tools to create or train robust machine learning models. It is end-to-end as well as open-source. Available since late 2015, this software has developed thousands of models for machine learning.

Hence in this model provided by Yin, landmark points are drawn across the student's facial landmarks. For this, the input is an array consisting facial coordinates, the image, landmark model of face given to a defined function. A numpy array of the coordinates of various facial landmarks is the output. This produces great results even if the person is wearing specs.

Fig 1: Detecting landmark points on a face with specs

The coordinates of the face act as the regions of interest and hence need to be extracted from the image. Square images of size 128*128 containing faces are given to the model. It returns the 68 key points which are converted to the original image dimensions. This Tensor flow model gives ~7.2 FPS. 0.05 seconds are taken by the landmark prediction step.

3.3 Analysing Candidate's Head Poses

Head pose estimation is basically figuring out the direction of the students's head. Four directions are taken into account – up, down, left, right. To analyse the head poses of a test-taker, her face is located in the frame and then her facial landmarks. For this, the python files created for face and facial landmark detection are imported.

Now, recognizing is easier when the test-taker is facing the camera. If the face is at an angle or some facial landmarks are not visible due to the test-taker's head-movements, then it is a problem. So, a function is needed to create the 3D coordinates. Six points of the face are accounted for the same i.e. the chin, nose tip, extreme left and right points of lips, and the left corner of the left eye and right corner of the right eye. After obtaining the required vectors, the image is those 3D points projected on a 2D surface. This is achieved using the OpenCV's project point function which gives a NumPy array of 2D coordinates as the output.

Next is creating an annotation box on the student's face so as to estimate the head poses. For this, the image, rotation and translation vector along with the camera matrix are given as the input. The output produced is lines that are drawn using OpenCV's polylines and line function. We find angle with the x-axis for recording the up and down movement of head and angle made with the y-axis to find the angle of movement. Combining these two, we get the direction to which the test-taker's head is facing.

The OpenCV's Video-Capture function triggers the webcam to start. Webcam feed is processed frame by frame to obtain an image from it. After detection of face in this image, specific facial landmarks identified are marked. A line is drawn from the face landmark coordinate of nose using functions and the angle is also calculated. The direction of the head is estimated to be left, right, up or down as per the angle value. The output is simultaneously displayed on the live feed and recorded in the log file.

3.4 Tracking the Candidate's Eye-Gaze

Analysing a test-taker's eye movements, such as she looking away from the screen or frequently changing her angle of focus, can help in detecting her cheating activities. To detect real-time gaze of the test-taker via webcam, the first step is finding her face and facial landmarks. Using the landmark points of eyes, test-taker's eyes are tracked. The estimated angles from the x-axis describe the movements on the left or right and those with y-axis tell about the upward and downward movement of eyes.

The OpenCV's Video-Capture function triggers the webcam to start. Webcam feed is processed frame by frame to obtain an image from it. After detection of face in this image, specific facial landmarks identified are marked. Functions are defined for finding the contour, masking the eye ball coordinates and identifying the direction of the eye ball. The image is further processed by implementing them. The direction of the eye ball is displayed as output on the live webcam feed as well as recorded in the log file.

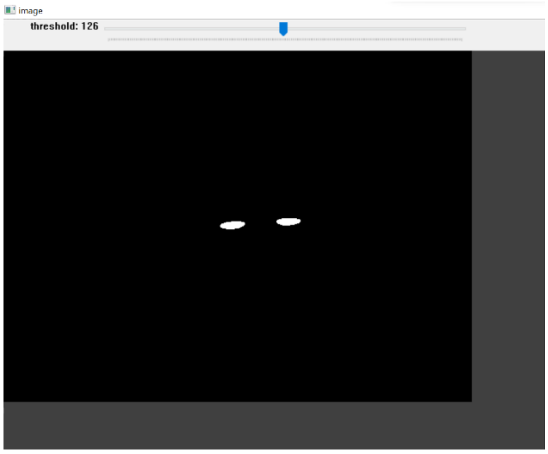

Fig 2: Mask created to locate eyes

To implement this, the python files created for face and facial landmark detection are imported. They contain functions which are used to obtain facial landmark coordinates. The movement of coordinates are hence tracked and recorded. Using the OpenCV's fillConvexPoly function, a mask of the size of the eye is created. The coordinates of the extreme points are identified by taking an array and a numpy array of facial landmarks of the eyes with a blank mask as input. From the processed image, an array of end points and the eye points' actual value is calculated and passed into a function which will find and return the eye ball positions. To find position of the eye, OpenCVs find Contours and moments functions are used.

3.5 Detect Cellphone or Another Person

To find the no. of people and use of cell-phone in a live feed, we use the YOLOv3 model along with OpenCV. The YOLO algorithm which uses the following three techniques:

a) Residual blocks: Any given image is divided into various grids each having a S x S dimension. The grid cell responsible for the object detection will be the one in which it appears [23].

b) Bounding box regression: A bounding box is an outline whose job is to highlight any object in an image [24]. Every bounding box in an image has the following attributes: Width (bw), Height (bh), Class (for example, person, car, animal, etc.), represented by the letter c and Bounding box center (bx, by).

c) Intersection over union (IOU) : Intersection over union (IOU) describes how boxes overlap. YOLO uses IOU to provide an output box for surrounding the objects perfectly. Each grid cell predicts the bounding boxes and their confidence scores. If the predicted bounding box is the same as the real box, the IOU is equal to 1. This mechanism eliminates bounding boxes that are not equal to the real box [23].

Using YOLOv3 weights and DenseNet Configurations, the YOLOv3 model is built [25]. The presence of a person or mobile phone on screen can be detected. After capturing webcam feed, the obtained image is resized to a size supported by the YOLOv3 model. A defined function processes the image, identifying objects in it. Different sized anchor boxes are used by the model. If the class ’person’ count is more than one, or if the class ‘cell-phone’ is detected, the same is shown on the output screen and the log file.

4. Results Analysis

Facial landmarks represent salient features of a person’s face, like her eyebrows, eyes, jawline, nose and mouth, as shown in Fig 3. To detect them, Dlib’s 68 keypoints landmark predictor is commonly used as it produces great results in real-time scenarios.

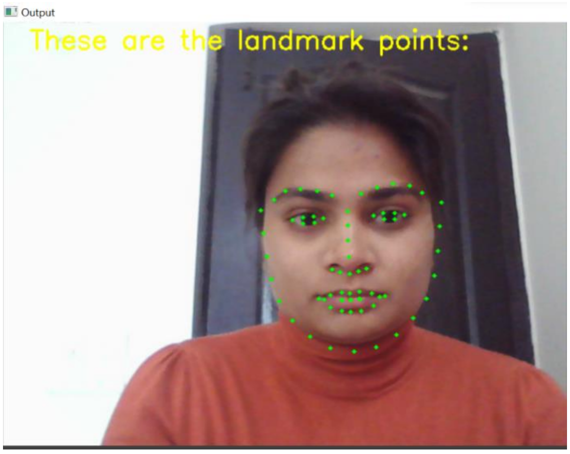

Fig 3: Identifying Facial Landmarks

To detect real-time gaze of the test-taker via webcam, test-taker's eyes are tracked using the landmark points of eyes. The estimated angles from the x-axis describe the movements on the left or right and those with y-axis tell about the upward and downward movement of eyes. In the figure 4, the test-taker is looking upwards which is shown on screen and the log file.

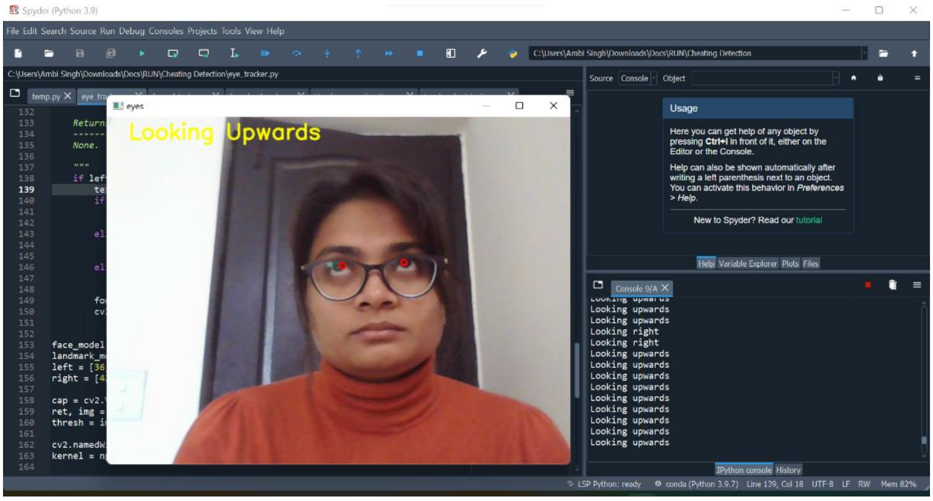

Fig 4: Output for Eye Tracking : Test-Taker looking Upwards

The direction of the eye ball is displayed as output on the live webcam feed as well as recorded in the log file. In Fig 5, the test-taker is looking leftwards, In Fig 6, she is looking to her right, which is simultaneously shown on screen as well as the log file.

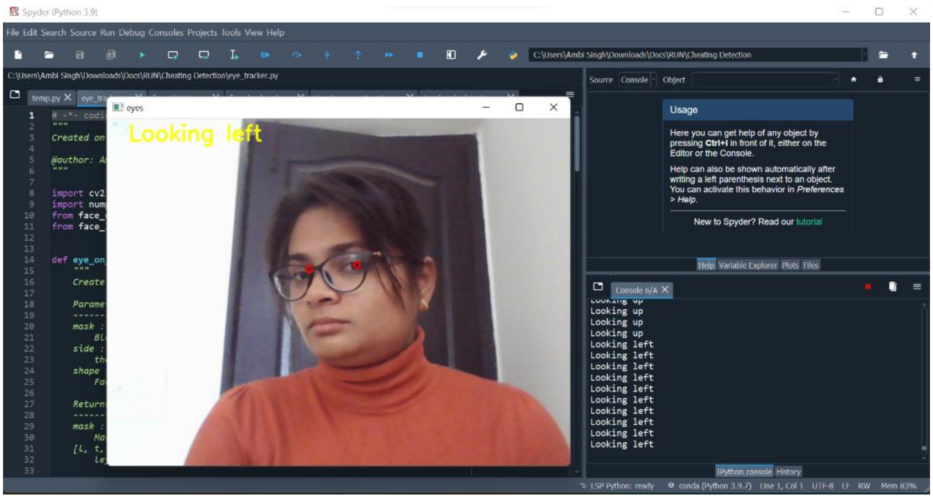

Fig 5: Output for Eye Tracking : Test-Taker looking leftwards

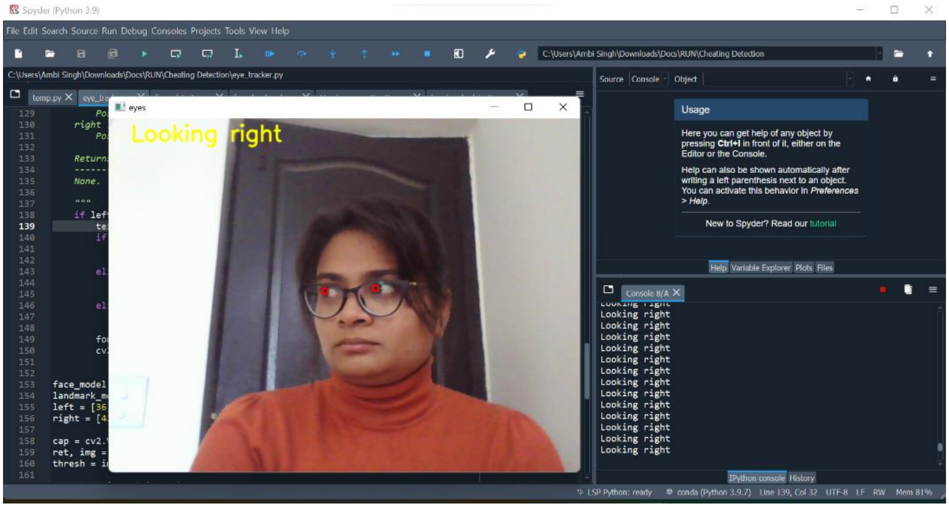

Fig 6: Output for Eye Tracking : Test-Taker looking rightwards

Head pose estimation is basically figuring out the direction of the test-taker's head. The direction of the head is estimated to be left, right, up or down as per the angle value. The yellow line shows angle of the head with the x-axis, and the blue line shows angle of the head with the y-axis, combining them direction of head movement of the test-taker. In Fig 7, the test-taker has moved her head to the upward direction, in Fig 8, her head is towards the right and that is simultaneously shown on the screen as well as the log file.

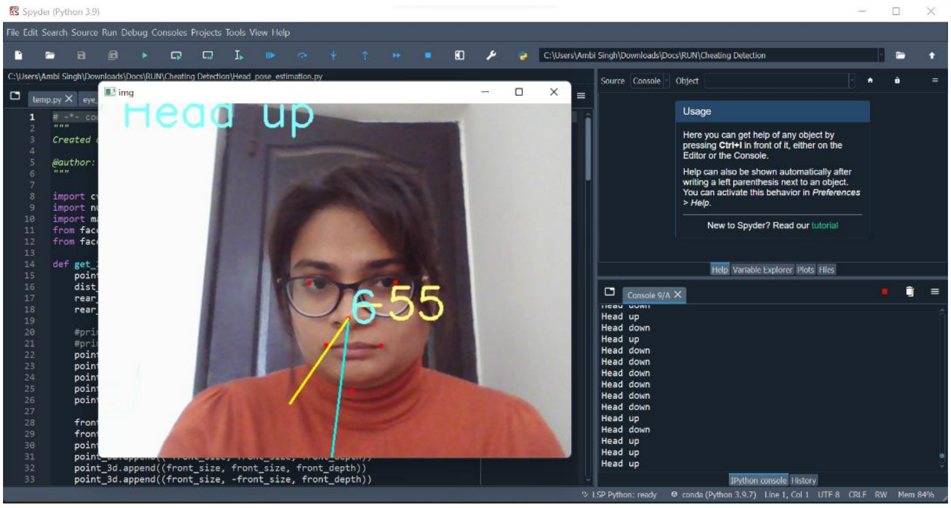

Fig 7: Output for Head pose Tracking : Upward Movement

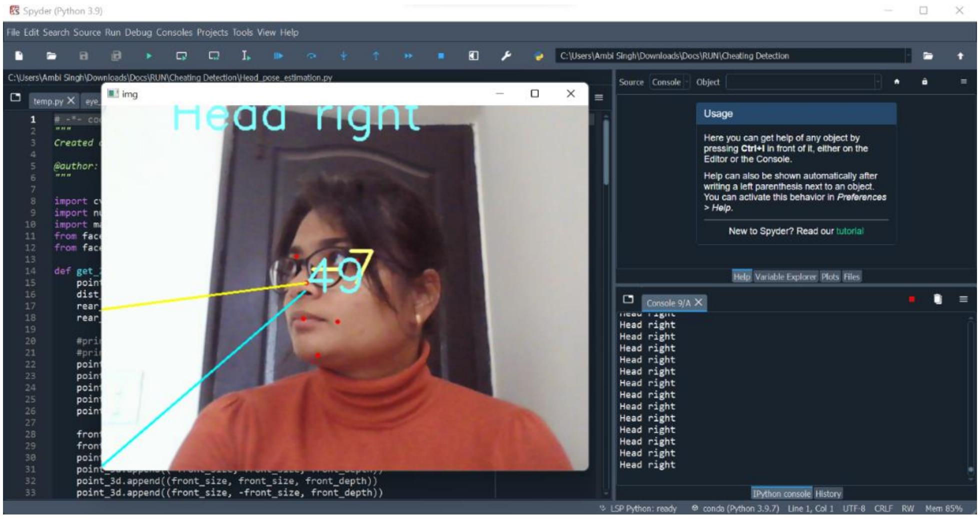

Fig 8: Output for Head pose Tracking : Movement towards right

The output is simultaneously displayed on the live feed and recorded in the log file. In the Fig 9, direction of the head downwards, In Fig 10, the test-taker has her head in the left direction.

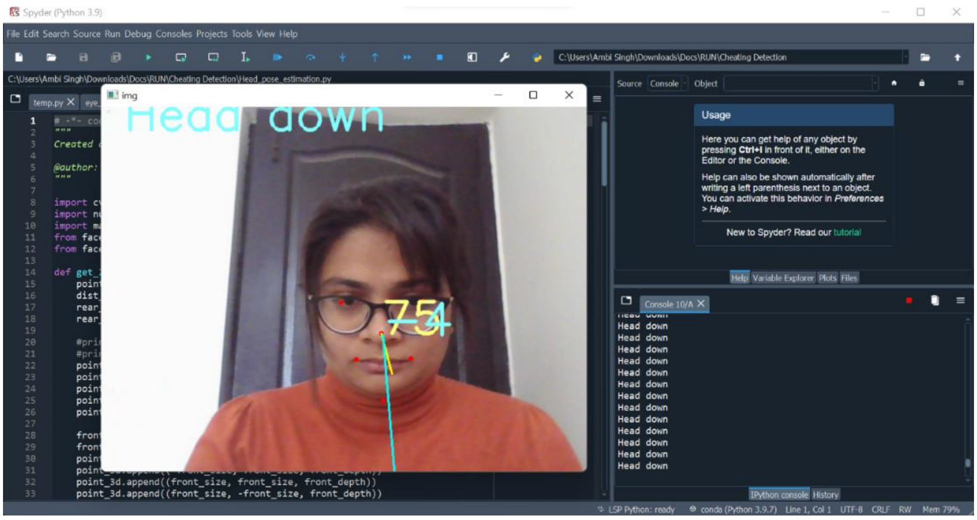

Fig 9: Output for Head pose Tracking : Downward Movement

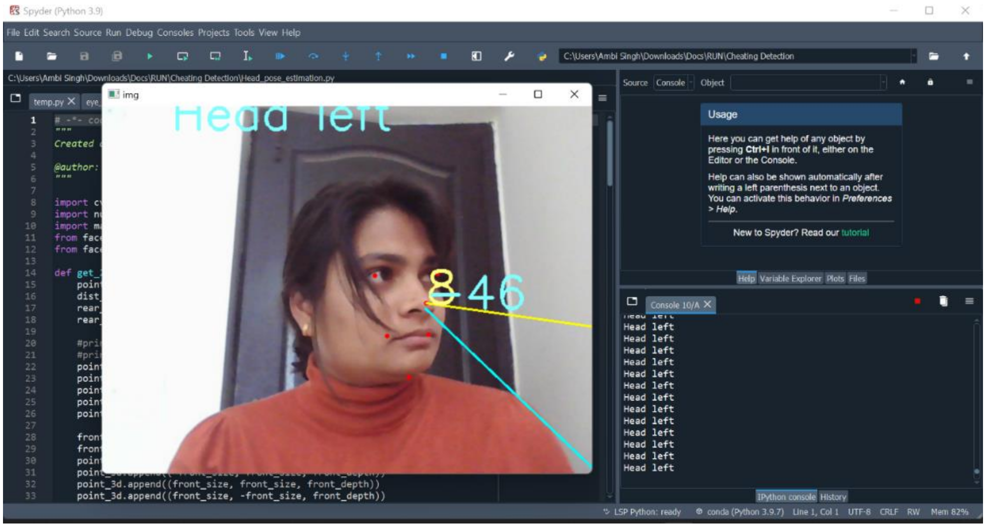

Fig 10: Output for Head pose Tracking : Movement towards left

5. Conclusion & Future Work

This main contribution in this research work is to present a smart system which can autonomously detect and track student's eye gaze and study the head orientation/poses to robustly detect his/ her cheating activities in an online exam. This work can be further enhanced by using external devices to capture back-view as well as side-view of the student/ test-taker. Spoofing should be detected to check if the test-taker is actually a real person taking the test. Fingerprint or voice authentication measures can be added to improve this system as a whole.

References

1. Butler-Henderson, K., & Crawford, J. (2020). A systematic review of online examinations: A pedagogical innovation for scalable authentication and integrity. *Computers & Education*, *159*, 104024.

2. Arora, S., Chaudhary, P., & Singh, R. K. (2021). Impact of coronavirus and online exam anxiety on self-efficacy: the moderating role of coping strategy. *Interactive Technology and Smart Education*.

3. Böhmer, C., Feldmann, N., & Ibsen, M. (2018, April). E-exams in engineering education—online testing of engineering competencies: Experiences and lessons learned. In *2018 IEEE global engineering education conference (EDUCON)* (pp. 571-576). IEEE.

4. Dendir, S., & Maxwell, R. S. (2020). Cheating in online courses: Evidence from online proctoring. *Computers in Human Behavior Reports*, *2*, 100033.

5. Shraim, K. (2019). Online examination practices in higher education institutions: learners' perspectives. *Turkish Online Journal of Distance Education*, *20*(4), 185-196.

6. Chen, B., Bastedo, K., & Howard, W. (2018). Exploring Design Elements for Online STEM Courses: Active Learning, Engagement & Assessment Design. *Online Learning*, *22*(2), 59-75.

7. Fayyoumi, A., & Zarrad, A. (2014). Novel solution based on face recognition to address identity theft and cheating in online examination systems. *Advances in Internet of Things*, *2014*.

8. Hu, S., Jia, X., & Fu, Y. (2018, August). Research on abnormal behavior detection of online examination based on image information. In *2018 10th International Conference on Intelligent Human-Machine Systems and Cybernetics (IHMSC)* (Vol. 2, pp. 88-91). IEEE.

9. Tiong, L. C. O., & Lee, H. J. (2021). E-cheating Prevention Measures: Detection of Cheating at Online Examinations Using Deep Learning Approach--A Case Study. *arXiv preprint arXiv:2101.09841*.

10. 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*, 1-48.

11. Fan, Z., Xu, J., Liu, W., & Cheng, W. (2016, August). Gesture based misbehavior detection in online examination. In *2016 11th International Conference on Computer Science & Education (ICCSE)* (pp. 234-238). IEEE.

12. Yongcun, W., & Jianqiu, D. (2021, March). Online Examination Behavior Detection System for Preschool Education Professional Skills Competition Based on MTCNN. In *2021 IEEE 2nd International Conference on Big Data, Artificial Intelligence and Internet of Things Engineering (ICBAIE)* (pp. 999-1005). IEEE.

13. Zhang, L., Zhong, X. L. B. H. B., & Cui, H. (2016). Research on Human Skeleton Modeling and Anthropometric Parameters Modeling for Online Examination. *Journal of Computers*, *27*(3), 15-20.

14. Kulkarni, M. D., & Alfatmi, K. (2021, June). New Approach for Online Examination Conduction System Using Smart Contract. In *2021 10th IEEE International Conference on Communication Systems and Network Technologies (CSNT)* (pp. 848-852). IEEE.

15. Bosch, N., D'Mello, S., Baker, R., Ocumpaugh, J., & Shute, V. (2015, June). Temporal generalizability of face-based affect detection in noisy classroom environments. In *International Conference on Artificial Intelligence in Education* (pp. 44-53). Springer, Cham.

16. B. Keresztury and L. Cser, "New cheating methods in the electronic teaching era ", ProcediaSocial and Behavioral Sciences, vol .93, pp.1516-1520, 2013.

17. Istiak Ahmad , Fahad AlQurashi , Ehab Abozinadah , Rashid Mehmood, "A Novel Deep Learning-based Online Proctoring System using Face Recognition, Eye Blinking, and Object Detection Techniques", (IJACSA) International Journal of Advanced Computer Science and Applications, Vol. 12, No. 10, 2021

18. Partha Pratim Debnath, Md. Golam Rashed, Dipankar Das, "Detection and Controlling of Suspicious Behaviour in the Examination Hall", International Journal of Scientific & Engineering Research Volume 9, Issue 7, July-2018

19. Razan Bawarith, Dr. Abdullah Basuhail, Dr. Anas Fattouh and Prof. Dr. Shehab GamalelDin "E-exam Cheating Detection System" In (IJACSA) International Journal of Advanced Computer Science and Applications, Vol. 8, No. 4, 2017 116.

20. Ali Javed, Zeeshan Aslam "An Intelligent Alarm Based Visual Eye Tracking Algorithm for Cheating Free Examination System" In I.J. Intelligent Systems and Applications, 2013, 10, 86-92

21. Gabriel Babatunde Iwasokun, Oluwole Charles Akinyokun, Taiwo Gabriel Omomule "Design of E-Invigilation Framework Using Multi Modal Biometrics" In 15th International Conference on Electronics Computer and Computation (ICECCO 2019)

22. Yousef Atoum, Liping Chen, Alex X. Liu, Stephen D. H. Hsu, and Xiaoming Liu "Automated Online Exam Proctoring" In IEEE TRANSACTIONS ON MULTIMEDIA, VOL. 19, NO. 7, JULY 2017

23. Weijun Chen, Hongbo Huang, Shuai Peng1, Changsheng Zhou, Cuiping Zhang1, "YOLOface: a real-time face detector" In Springer-Verlag GmbH Germany, part of Springer Nature 2020.

24. Meng-ting Fang , Zhong-ju Chen ,Krzysztof Przystupa , Tao Li , Michal Majka and Orest Kochan , "Examination of Abnormal Behaviour Detection Based on Improved YOLOv3", Article in Electronics, January 2021

25. S. Harish, D. Rajalakshmi, T. Ramesh, S. Ganesh Ram, M. Dharmendra, "New Features for Webcam Proctoring Using Python and OpenCV", ISSN: 2237-0722 Vol. 11 No. 2 (2021)