The Detection of Students' Abnormal Behavior in Online Exams Using Facial Landmarks in Conjunction with the YOLOv5 Models

Muhanad Abdul Elah Alkhalisy
Informatics Institute for Postgraduate Studies
Baghdad, Iraq
phd202020557@iips.icci.edu.iq

Saad Hameed Abid
Department of Computer Science
Al-Mansur University College
Baghdad, Iraq
Saad.hameed@muc.edu.iq

***Abstract* - The popularity of massive open online courses (MOOCs) and other forms of distance learning has increased recently. Schools and institutions are going online to serve their students better. Exam integrity depends on the effectiveness of proctoring remote online exams. Proctoring services powered by computer vision and artificial intelligence have also gained popularity. Such systems should employ methods to guarantee an impartial examination. This research demonstrates how to create a multi-model computer vision system to identify and prevent abnormal student behaviour during exams. The system uses You only look once (YOLO) models and Dlib facial landmarks to recognize faces, objects, eye, hand, and mouth opening movement, gaze sideways, and use a mobile phone. Our approach offered a model that analyzes student behaviour using a deep neural network model learned from our newly produced dataset" StudentBehavioralDS." On the generated dataset, the "Behavioral Detection Model" had a mean Average Precision (mAP) of 0.87, while the "Mouth Opening Detection Model" and "Person and Objects Detection Model" had accuracies of 0.95 and 0.96, respectively. This work demonstrates good detection accuracy. We conclude that using computer vision and deep learning models trained on a private dataset, our idea provides a range of techniques to spot odd student behaviour during online tests.**

Index items: Facial Landmarks, Behaviour Recognition, Dlib, Online Proctoring, Deep Learning.

I. INTRODUCTION

Information technology (IT) significantly influences people's lives as they assimilate more into society. E-learning has benefitted greatly during natural disasters, war, and pandemics [1]. The availability of online education increased. MOOCs, or massive open online courses, are gaining popularity [2]. Due to various technical advancements, E-learning may employ computer vision and machine learning [3]. Studies of online education have concentrated on the topic of course evaluation [4]. A problematic aspect of online course evaluation is the lack of direct student-teacher contact [5]. Most schools switched to an all-online curriculum after the COVID-19 epidemic [6]. Online courses and assessments are growing [4].

Artificial intelligence-powered proctoring solutions may be necessary for online assessments like MOOCs and recruiting exams [7]. A challenging exam must be passed to earn a top-notch online credential. As in classrooms and colleges, online examinations need to be proctored. Cheating is much easier to do on online exams. Therefore, it is necessary to have an AI-based system keep tabs on every student [8]. Deep learning's advent has aided the advancement of computer vision. Deep learning techniques may be used to complete everyday computer vision tasks, such as detecting anomalous behaviour in exams. Deep learning-based object identification algorithms have successfully succeeded in various fields [9]. Cameras and microphones might be necessary for such technologies to keep a tab on the students.

The AI-driven system would detect instances of fraud [10]. Systematic anti-cheating measures and responses It is possible to halt the examination or submit a report for staff review. Proctors in the form of humans may use monitoring software to keep tabs on the students. When cheating is discovered, a human proctor is contacted, and their questionable actions are recorded [11]. A human proctor may be ineffective if students attempt to take examinations from a place with inadequate internet access or electrical problems. Any problems with their live video might flag them for disqualification. Since the exam may be taken with the computer running, a proctoring system that operates automatically is best suited [12]. Detecting abnormal activity to avoid cheating is crucial for the quality of online assessments [13].

This study suggests a computer vision-based automated webcam-based proctoring system. The technology alerts teachers to unusual conduct in students, such as using a phone, talking to multiple people, glancing to one side, moving their eyes or hands, or opening their mouths. The proposed system combines vision-based capabilities by using three models, Dlib, YOLOv3, and YOLOv5, where the first model is used to extract facial landmarks to detect the state of the mouth opening, the second model is used to detect people and objects. The third model, which was trained using a newly developed dataset, analyzes student

behaviour. Every suggested model was created using a multithreading process.

The following sections make up this paper: Section II addresses the literature review; Section III outlines the suggested methods; Section IV includes the experiments and results; and Section V concludes with suggestions for further research.

violation risk [14]. Through many techniques, such as live video and audio streaming for the candidate and the candidate's surrounding environment, liveliness checks of the candidate, and facial comparisons with his or her photograph taken during the examination, the online examination system e-Parakh enables both supervised and unsupervised remote monitoring of the examination [15].

[16] The proposed technique helps examiners decide whether students pass online tests without misconduct. The system categorizes student visual focus of attention data using head position, eye gaze estimations, and machine learning (ML) algorithms.

[17] built an automated exam activity detection system that monitors students' body movements and utilizes deep learning to classify their activities into six categories. The activities include usual conduct, looking back, gazing forward, gestures, and glancing left or right. [18] Present a program that offers online student verification through biometrics (facial, voice, and typing) and a proctoring system. This paper describes a solution based on biometric authentication and an autonomous proctoring system. [19] offered an approach to create a complete AI-based system that can prevent test cheating. The system keeps an eye out for fraud and logs any evidence. This technology will be secure and reasonably priced.

However, employing YOLOv5 to automatically assess a student's behaviour during an online exam is not a strategy that has been studied. Additionally, our research utilized a newly created dataset called "StudentBehavioralDS" to learn the model for analyzing and detecting anomalous student behaviour, aiding in the fair administration of exams.

III. Proposed Method

In anomaly detection research, many different, complex cases are encountered, making it very challenging to find a solution. The main focus of our methodology is indoor examination rooms. The primary objective is automatically categorizing aberrant frames, which is subsequently communicated to a human reviewer in an executive summary. The system only needs the webcam or any other compatible camera. Fig. 1 shows the essential parts of the proposed system.

The following sections go into further detail on creating datasets, detection models, data cleaning and preparation, dataset augmentation, training of behavioural detection models, and result analysis.

II. Literature Review

Exams must be fair, and students must be shown to understand the material as more and more institutions convert to the digital world. This is made possible via fair ongoing assessments.

Response automates online exam proctoring. A webcam records the student during the exam, and an AI engine analyzes the video for anomalies. After these activities, the system creates a report that ranks proctoring results by exam

A. *Dataset Development*

This article manually created a dataset for the behaviour detection of online exam students because videos taken during actual online exams are not publicly available [20].

B. *Dataset Collection*

Twenty-four movies were taken using a webcam at a frame rate of 15 frames per second and a resolution of 1280 x 720. We have devised and created our data-gathering and labelling approach. With the help of hundreds of questions chosen from a question pool, we have used our custom-built online exam web application with video capture capabilities (online Exam simulator) [21] to assess students' competence levels.

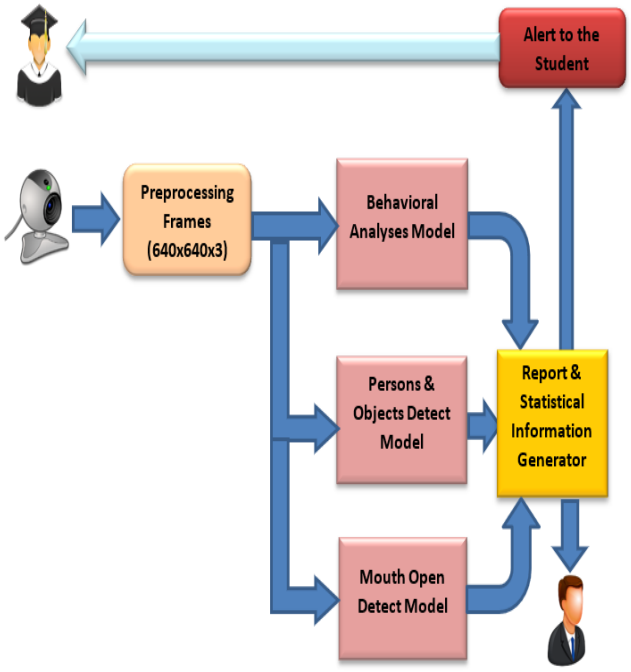


Fig. 1 illustrates the essential parts of the proposed system

The pool of questions includes multiple-choice options. During the acquisition of the dataset, the six participants were required to engage in various deliberate cheating scenarios in the context of online and closed-book exams throughout the videos; these motions included utilizing a phone, moving a hand or an eye, and shifting the head left or right. Participants joined the trials while being exposed to various test locations, camera settings, lighting conditions, etc. These changes make it challenging to catch cheating incidents. The videos mimic a student completing an online

test in front of a webcam. The overall running time of all films is roughly six hours, with each video lasting roughly fifteen minutes for each scenario.

C. *Videos to Frames Conversion*

Images were extracted from each recorded video at predetermined frame intervals. One frame was captured for every continuous ten frames to ensure scene variation. After converting, there are about 1,200 images for each video. Based on five scenarios, 7500 images were obtained. The total number of images captured from all videos was 37500 images.

After manual filtering was performed on the collected images, excluding any frames that do not have anomalous behaviour and blurred images, our dataset has a total of 8,520 images.

D. Ground Truth Data Labeling

The challenge of assigning a class subject to each frame was challenging. Image annotation assigns labels to images taken from a dataset that can be used to train a model. They provide details about the image, including its location and shape. LabelImg and MakeSense, an open-source application for annotating digital pictures and movies, are two of the most utilized technologies in computer vision annotation. The browser-based program MakeSense enables a range of work situations. For instance, massive picture libraries with ground truth labels are needed to train deep learning algorithms for object detection and identification. In this study, each instance of cheating is labelled using both tools. Figure 2 shows example instances of cheating and how each picture was manually labelled using the LabelImg program.

Fig.2 The Manual Ground Truth Labeling

We create the ground truth dataset by manually annotating each frame with a label to assign each activity to a particular behaviour. The participants' heads and torsos were usually visible in the webcam's field of view. The ground truth data was collected by manually labelling the head position, eye, mouth, and pose.

Because defining what is considered normal and abnormal behaviour is subjective, the behaviour ground truth was annotated separately. Based on specific behaviours, the dataset contains five classes: mobile using, hand moving, eye moving, mouth open, and looking side. Images of students holding their phones or moving their hands were labelled with the words "mobile using" and "hand move," respectively. Images of students moving their heads were tagged as looking side, while those of students moving their eyes or mouth were tagged as eye movements and mouth movement. As shown in Fig. 3, The annotation was in normal YOLO annotation format and was based on the suggested model utilized in this study; a txt file with the same name is produced for each picture file in the same directory. The annotations for each associated picture file are stored in a Txt file and include the image file's 'object class', 'object coordinates', 'height, and 'width', as in (1).

<'Object-Class'><X><Y><Width><Height> (1)

The annotations are easier to deal with even after resizing or stretching photos because they are normalized to lie within the range [0, 1]. A new line is drawn for each object in the picture. The picture below shows a YOLO annotation with two separate things in it.

```
1 1 0.408 0.30266666666666664 0.104 0.15733333333333333
2 1 0.245 0.424 0.046 0.08
```

Fig. 3: Text File Including Information About Annotations

Sample dataset snaps are displayed in Fig. 4. StudentBehavioralDS is the dataset's name.

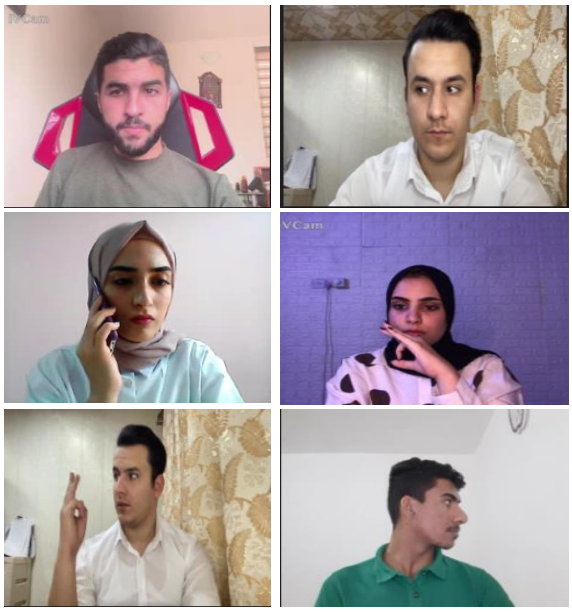


Fig. 4: Image Snippets from a Sample Dataset

Table 1 provides details regarding the developed dataset.

TABLE I: Developed Dataset Details

Index	Name	No. of Images	Description
0	Normal	1500	No Abnormal Behavioral
1	Mobile_Using	1720	Students Using Mobile Phones
2	Hand_Move	1700	Student Moving his Hand

3	Eye_Move	1700	Student Moving his Eye Left or Right
4	Looking_Side	1400	Student Looking Left or Right
	Total	8520	

E. Detection Models

Object identification can help identify objects in images. Classifying objects involves location, picture categorization, and object detection.

1) Abnormal Behavioral Detection

We must train a convolutional neural network model to identify unusual student behaviour on our recently constructed dataset. These abnormal behaviours are related to our dataset's favoured classes (subjects) (Mobile Using, Hand Move, Eye Move, Looking Side). In this work, YOLOv5s was used and trained on our private dataset [22]. The structural elements of the YOLOv5 framework are the backbone, neck, and predict head. The neck creates feature maps at three distinct sizes while the backbone extracts feature data from incoming images. Utilizing these feature maps, the prediction head merges extracted information to provide richer target characteristics [22]. In this model, the non-maximum suppression (NMS) method [23] is used. The accessible network topology of YOLOv5 is depicted in Fig. 5. To locate the best anchor frame, YOLOv5 adjusts clustering to a variety of training datasets [24]. YOLOv5 tried to activate the sigmoid, leakyReLU, and SiLU functions [23].

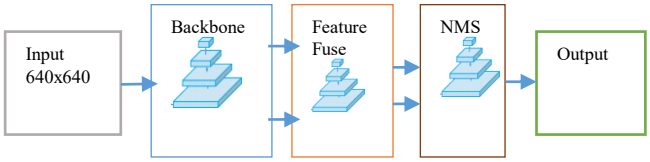


Fig.5 Yolov5 Main Architecture

2) Mouth Open Detection

The Dlib was utilized to perform face, mouth, nose, and eye detection. Dlib is a library for data analysis and machine learning that may be used to make practical applications [16]. Python bindings make it simple to utilize this library. This collection may be used to pinpoint 68 different facial landmarks, such as the chin, jawline, eyebrows, nose, eyes, and lips. In addition to rough face detection, exact facial areas may be extracted from landmark points. The result of eliminating data noise will considerably enhance face-recognition models. Dlib is used by two shape prediction algorithms that are based on the BUG300W face landmark dataset. These algorithms detect sixty-eight and five landmark points in a picture, respectively [16]. The 68 face landmarks used in this work are represented by their indexes in Fig. 5. Dlib employs a Histogram of Oriented Gradients-based face detector (HOG). Due to the fact that these indices, which are shown in Figure 6, are the same ones that Dlib employs, we are able to rapidly establish which areas on the face correspond to which component.

Based on the output of Dlib facial landmarks, the open mouth detection technique simply calculates the distance between the points around the mouth's corners to determine whether or not the mouth is open. The model detects mouth opening state if the distance for at least three outer pairings and two inside pairs is more significant than their corresponding threshold values. Outer and inner thresholds were 4 and 3.

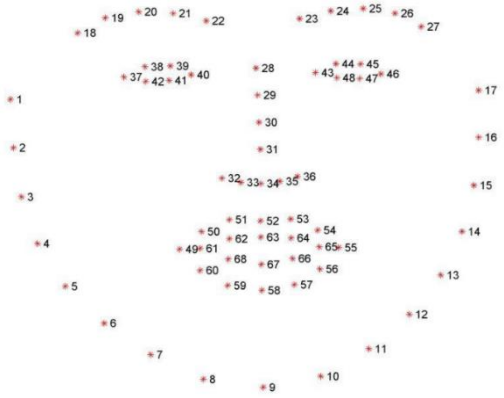


Fig 6. The 68 Indices of Face landmark [16]

3) Persons and Objects Detection

In order to recognize persons, groups of persons, and other objects in the live webcam feed, we utilized the pre-trained model of YOLOv3. The COCO dataset, with its 80 labels, was used to train this model [9]. The YOLOv3 algorithm has made several improvements over YOLOv1 and YOLOv2, considerably boosting detection accuracy and detection speed. Its central tenet is the actuality of using CNN "end-to-end" to complete the entire object detection procedure [9]. The number of persons that were counted was determined by looking at footage from a camera. In the event that the number is either equal to or larger than one, an alarm will go off. Because the COCO dataset includes 1, 67, and 74 person, book, and laptop indices, respectively, before we can send out an alert, we need to check to see whether there are any similar class indices.

C. Data Preprocessing

The dataset consists of video frames from various subjects recorded at various periods. The videos that were used produced 1280x720 RGB-formatted frames. The collected frames were resized to 640x640 resolution before being used to train the Yolov5 model.

1) Data Cleaning

After the manual labelling process was finished, data cleaning was carried out. We cleaned up the data using the following techniques in addition to hand curation:

- Get rid of duplicate image files.
- Get rid of duplicated annotation files.
- Eliminate any photos that lack an associated annotation file.
- Eliminate annotation files that lack associated image files.

2) Dataset Sipliting

Before beginning the training procedure, the dataset must first be organized and then divided into training and validation segments using a manner that is logically suited

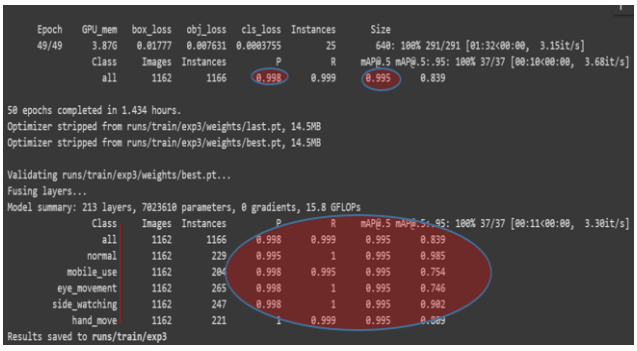
for YOLO network formats. The dataset is then broken down as follows: 10% of the dataset should be spent on testing, 20% on validating, and 70% on training.

A. *Online and Offline Augmentation*

Deep learning requires many data during the training phase. The process of augmentation is utilized to expand the dataset. In our work, we employ both offline and online augmentation. Unbalanced training data will force the network model to focus on more objects during training, which is a surefire way to make the model overfit. Although sample collecting is difficult in real life, there are comparatively few eye movement samples in the dataset of this study. We used offline data augmentation to increase the variety of eye movement poses and enhance network performance. In this work, data augmentation was done on several images of the student moving his eye. The chosen photos were given a horizontal flip, brightness enhancement, and Gaussian noise addition. Several online augmentations to the dataset are also carried out by the Yolov5 algorithm preprocessing, including HSV H (0.014), HSV S (0.6) and V (0.30), Flip right (0.5), Copy paste (0.1), and Scale (0.5). These increase the size of the dataset.

B. *Behavioral Detection Model Training*

A successful deep-learning model must be created by carefully adjusting the hyperparameters. The initial learning rate, anchor-multiple thresholds, Stochastic Gradient Descent (SGD) momentum/Adame, and batch size are examples of common hyperparameters in the yolv5 model. We begin the training process using 1162 photo patches of entities from six different classes, starting learning rate of 0.01, anchor-multiple threshold of 5.0, Stochastic Gradient Descent momentum of 0.936, epochs of 50, and batch size of 16. As a hardware accelerator, we trained our model in Google Colab using the GPU. The model converges during the course of training with satisfactory results for mAp=0.995 and accuracy=0.95. Figure 7 displays the outcomes of the behavioural detection model training.



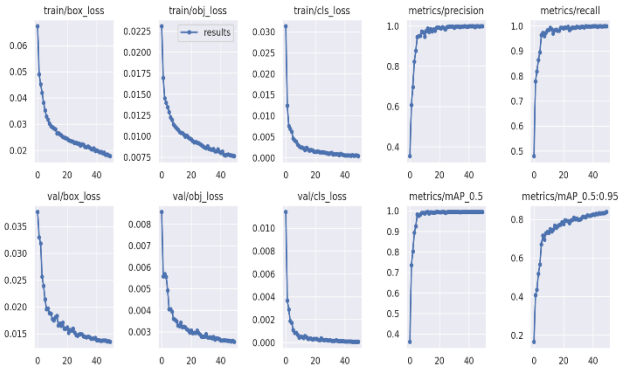


Fig. 7 Results of Training the Behavioral Detection Model

IV. Experimental and Results

Our method is implemented using Dlib, YOLO techniques, and OpenCV. The webcam is activated using OpenCV's Video Capture. The camera feed is captured and analyzed frame by frame. We established three threads—Behavioral Analysis, Person-Objects, and Mouth Opening—so that each model ran concurrently to speed up the process. The person and object detection model was used to analyze this frame, and the output included the object boxes, prediction score, and number of objects of each category. Results include more than one person being printed if the class 'person' has more than one occurrence and a book or laptop being printed if the class "book" or "laptop" is found. After receiving a copy of the input frame and processing it using the Yolov5 model, the behavioural detector model creates a prediction score for each class. If the class "eye motion" is recognized, for instance, alarm messages are printed as output, and the detection time is logged for later use in statistical analysis and report preparation.

68 (x, y)-coordinates for facial structures make up the model's outputs. You can locate faces by using simple Python indexing with coordinates like [48, 68] for the mouth and [27, 35] for the nose. For statistical analysis and report production, all model outputs are recorded.

V. Performance EvaluationA series of experiments have explored models and system effectiveness. The findings are presented here. It is crucial to understand terms like "TN," "TP," "FP," "FN," "precision," "accuracy," and "recall [25].

True Positive (TP): Good samples with accurate labelling. True Negative (TN): How many samples have accurate negative labels? False Positive (FP): Negative samples are mistakenly categorized as positive. False Negative (FN): Negative samples with incorrect labels.

Accuracy: the percentage of classrooms where the predictions were accurate, and it is represented by Eq. (2):

$$Accuracy = \frac{\mathrm{TP} + \mathrm{TN}}{\mathrm{TP} + \mathrm{TN} + \mathrm{FP} + \mathrm{FN}} \times 100 \qquad (2)$$

Precision: the proportion of really positive classifications that are correctly predicted and it is represented by the Eq. (3):

$$Precision = \frac{TP}{TP + FP} \times 100 \quad (3)$$

Recall: All positive courses accurately predict the proportion of classes, and it is represented by the Eq. (4):

$$Recall = \frac{TP}{TP + FN} \times 100 \quad (4)$$

Models for object detection, such as YOLO, are evaluated using the mean average precision (mAP). The mAP calculates a score by comparing the detected box to the ground-truth bounding box. The model's detections are more precise the higher the score.

Average Precision (AP): The weighted average of the precisions at each threshold is calculated; it considers the increase in recall from the previous threshold and it is represented by the Eq. (5):

$$AP = \sum_{k=0}^{k=n-1} [\text{Recall}(k) - \text{Recall}(k+1)] * \text{Precisions}(k) \quad (5)$$

Mean Average Precision (mAP): The average AP for each class is known as the Mean Average Precision and it is represented by the Eq. (6):

$$\text{mAP} = 1/n \sum_{k=1}^{k=n} APk \quad (6)$$

A. Behavioral Detector Model Results

The analysis revealed that YOLOv5 has a significant reduction in inference time and a high mean average accuracy (mAP). Our results show that our experiments had an accuracy rate of 0.87 per cent. The results of 250 pictures tested on the trained models are shown in Table 2.

According to our data, the accuracy rate of Fig. 8 depicts a few of our findings.

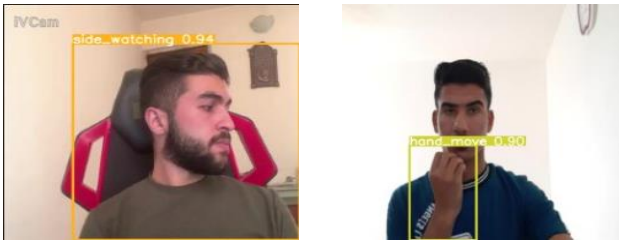


Fig. 8 illustrates some model detection results.

B. Mouth Opening Detector Results

The output of the mouth opening detect model, along with facial landmarks and the mouth opening, is shown in Fig. 9.

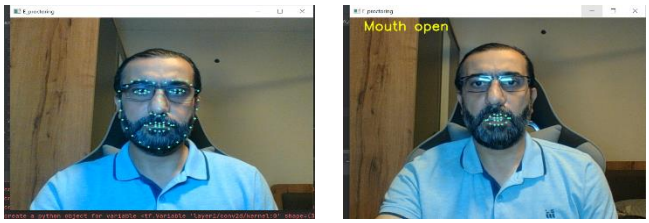


Fig. 9 Model Results for Mouth and Facie landmark

C. Person and Objects Detector

The individual and other things visible in the camera's field of view, such as the book or notepad, are detected during the object detection model's assessment. Figure 10 shows the model's results. Human detection had a 99.91 per cent accuracy rate, and object detection had a 97.08 per cent accuracy rate.

Fig. 10 Book and Person Detection

The overall assessment of the suggested student behaviour analysis is shown in Table 3. Table 3's findings demonstrate that the suggested methods for this research are accurate and reliable in spotting students' abnormal conduct during online tests.

TABLE 2: display the outcomes of 250 photos being tested on

Class Name	**Precision (%)**	**Recall (%)**	**mAP (%)**
Normal	**0.86**	**0.86**	**0.86**
Mobile_Use	**0.94**	**0.97**	**0.97**
Hand_Move	**0.94**	**0.92**	**0.92**
Eye_Move	**0.85**	**0.87**	**0.86**
Looking_Side	**0.92**	**0.90**	**0.90**

TABLE 3: Final evaluation findings for the suggested models

Action	Accuracy (%)	Recall (%)	Precision (%)
Behavioral Analysis	0.824	0.910	0.876
Person detection	0.933	0.965	0.964
Objects detection	0.946	0.823	0.933
Mouth Open	0.957	0.921	0.939
Multi-person	0.943	0.924	0.932

Table 4 compares the work provided with past efforts in terms of both the approach used and the precision of the results.

TABLE 4: Comparison of the Proposed Work with Related Works

Works	Object Person & Detection	Face Detector Model	Facial Landmarks Model	Behavioral Detector Model	Dataset Used	Over All Accuracy
Our Work	YOlOv3	(HOG) Dlib	Dlib Facial Landmarks	Yolov5	Our Private DS	0.95
[19]	YOlOv3	CNN +Haar Cascade	Haar-Cascade Classifier	None	None	0.92
[17]	None	L2-GraftNet CNN	None	L2-GraftNet	CUI-EXAM	0.88
[18]	None	FaceBoxe method	FaceBoxeme thod	M3L	FDDB	0.89

VI. Conclusion And Future Work

This paper uses computer vision and deep neural networks to provide a multi-model approach to avoid and examine unusual student behaviour during online assessments. Our approach includes the following: object identification, mouth open-close detection utilizing facial landmarks, face detection, eye movement, head movement, hand movement, multi-person, and mobile use. In order to track student behaviour during tests, our proposed solution makes use of a pre-trained deep neural network model that was found using our recently created dataset. With the use of a multithreading technique, which included launching three different processes, models were run simultaneously to speed up the system. Our study and evaluation foundation were three metrics: recall, precision and mean_average_precision. Consequently, we received a mean average precision score of 0.87 for behavioural analyses, 0.97 for the person and object detection, and 0.95 for mouth-opening detection. Finally, after overall work evaluation, our proposal still has limitations like the web camera's position (according to display) is fixed, the user's head is elevated similarly to the same level as the camera, and the users are required to remove the spectacles eyeglasses.

As a potential future phase, adding more dataset classes, audio synthesis, and biometric authentication for applicant authorization might be added. Additionally, we would prefer a single Yolo model for both the behavioural and object detectors rather than two Yolo models.

References

[1] S. Aisyah, Y. Bandung, and L. B. Subekti, “Development of Continuous Authentication System on Android-Based Online Exam Application,” *2018 Int. Conf. Inf. Technol. Syst. Innov. ICITSI 2018 - Proc.*, pp. 171–176, Jul. 2018, doi: 10.1109/ICITSI.2018.8695954.

[2] H. Alessio and K. Maurer, “The Impact of Video Proctoring in Online Courses.,” *J. Excell. Coll. Teach.*, vol. 29, pp. 183–192, 2018.

[3] 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,” *IEEE Access*, vol. 9, pp. 32689–32712, 2021, doi: 10.1109/ACCESS.2021.3060192.

[4] S. Dendir and R. S. Maxwell, “Cheating in online courses: Evidence from online proctoring,” *Comput. Hum. Behav. Reports*, vol. 2, no. October, p. 100033, 2020, doi: 10.1016/j.chbr.2020.100033.

[5] S. Arnò, A. Galassi, M. Tommasi, A. Saggino, and P. Vittorini, “State-of-the-art of commercial proctoring systems and their use in academic online exams,” *Int. J. Distance Educ. Technol.*, vol. 19, no. 2, pp. 41–62, 2021, doi: 10.4018/IJDET.20210401.oa3.

[6] P. A. Novick, J. Lee, S. Wei, E. C. Mundorff, J. R. Santangelo, and T. M. Sonbuchner, “Maximizing Academic Integrity While Minimizing Stress in the Virtual Classroom,” *J. Microbiol. Biol. Educ.*, vol. 23, no. 1, pp. 0–10, 2022, doi: 10.1128/jmbe.00292-21.

[7] S. Vincent-Lancrin and R. van der Vlies, “Trustworthy artificial intelligence (AI) in education : Promises and challenges,” *OECD Educ. Work. Pap. No. 218*, no. 218, p. 17, 2020, [Online]. Available: https://www.oecd-ilibrary.org/education/trustworthy-artificial-intelligence-ai-in-education_a6c90fa9-en

[8] A. Nigam, R. Pasricha, T. Singh, and P. Churi, “A Systematic Review on AI-based Proctoring Systems: Past, Present and Future,” *Educ. Inf. Technol.*, vol. 26, no. 5, pp. 6421–6445, 2021, doi: 10.1007/s10639-021-10597-x.

[9] M. T. Fang, K. Przystupa, Z. J. Chen, T. Li, M. Majka, and O. Kochan, "Examination of abnormal behaviour detection based on improved YOLOv3," *Electron.*, vol. 10, no. 2, pp. 1–17, 2021, doi: 10.3390/electronics10020197.

[10] E. A. Hall, M. B. Roberts, K. A. Taylor, and D. E. Havrda, “Changes in Academic Performance after Transitioning to Remote Proctoring: A Before-After Evaluation,” *Pharmacy*, vol. 10, no. 4, p. 92, 2022, doi: 10.3390/pharmacy10040092.

[11] S. Kaddoura, D. E. Popescu, and J. D. Hemanth, “A systematic review on machine learning models for online learning and examination systems,” no. Ml, pp. 1–32, 2022, doi: 10.7717/peerj-cs.986.

[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, no. September, p. 104024, 2020, doi: 10.1016/j.compedu.2020.104024.

[13] F. F. Kharbat and A. S. Abu Daabes, “E-proctored exams during the COVID-19 pandemic: A close understanding,” *Educ. Inf.*

Technol., vol. 26, no. 6, pp. 6589–6605, 2021, doi: 10.1007/s10639-021-10458-7.

[14] M. J. Hussein, J. Yusuf, A. S. Deb, L. Fong, and S. Naidu, "An Evaluation of Online Proctoring Tools," *Open Prax.*, vol. 12, no. 4, p. 509, 2020, doi: 10.5944/openpraxis.12.4.1113.

[15] A. K. Pandey, S. Kumar, B. Rajendran, and B. B S, "E-parakh: Unsupervised online examination system," *IEEE Reg. 10 Annu. Int. Conf. Proceedings/TENCON*, vol. 2020-Novem, pp. 667–671, 2020, doi: 10.1109/TENCON50793.2020.9293792.

[16] C. S. Indi, K. V. Pritham, V. Acharya, and K. Prakasha, "Detection of Malpractice in E-exams by Head Pose and Gaze Estimation," *Int. J. Emerg. Technol. Learn.*, vol. 16, no. 8, pp. 47–60, 2021, doi: 10.3991/ijet.v16i08.15995.

[17] T. Saba, A. Rehman, N. S. M. Jamail, S. L. Marie-Sainte, M. Raza, and M. Sharif, "Categorizing the Students' Activities for Automated Exam Proctoring Using Proposed Deep L2-GraftNet CNN Network and ASO Based Feature Selection Approach," *IEEE Access*, vol. 9, pp. 47639–47656, 2021, doi: 10.1109/ACCESS.2021.3068223.

[18] M. Labayen, R. Vea, J. Florez, N. Aginako, and B. Sierra, "Online Student Authentication and Proctoring System Based on Multimodal Biometrics Technology," *IEEE Access*, vol. 9, pp. 72398–72411, 2021, doi: 10.1109/ACCESS.2021.3079375.

[19] A. Sridhar and J. S. Rajshekhar, "AI-Integrated Proctoring System for Online Exams," *J. Artif. Intell. Capsul. Networks*, vol. 4, no. 2, pp. 139–148, 2022, doi: 10.36548/jaicn.2022.2.006.

[20] L. Tang, T. Xie, Y. Yang, and H. Wang, "Classroom Behavior Detection Based on Improved YOLOv5 Algorithm Combining Multi-Scale Feature Fusion and Attention Mechanism," *Appl. Sci.*, vol. 12, no. 13, 2022, doi: 10.3390/app12136790.

[21] M. A. E. Alkhalisy, "online-exam-emulator-iips," *IIPS*, 2022. https://online-exam-emulator-iips.web.app/

[22] Z. Ying, Z. Lin, Z. Wu, K. Liang, and X. D. Hu, "A modified-YOLOv5s model for detection of wire braided hose defects," *Meas. J. Int. Meas. Confed.*, vol. 190, no. September 2021, p. 110683, 2022, doi: 10.1016/j.measurement.2021.110683.

[23] R. Li and Y. Wu, "Improved YOLO v5 Wheat Ear Detection Algorithm Based on Attention Mechanism," *Electron.*, vol. 11, no. 11, 2022, doi: 10.3390/electronics11111673.

[24] L. Zhu, X. Geng, Z. Li, and C. Liu, "Improving yolov5 with attention mechanism for detecting boulders from planetary images," *Remote Sens.*, vol. 13, no. 18, pp. 1–19, 2021, doi: 10.3390/rs13183776.

[25] F. Mahmood *et al.*, "Implementation of an Intelligent Exam Supervision System Using Deep Learning Algorithms," *Sensors*, vol. 22, no. 17, 2022, doi 10.3390/s22176389.

[26] M. Cao, H. Fu, J. Zhu, and C. Cai, "Lightweight tea bud recognition network integrating GhostNet and YOLOv5," *Math. Biosci. Eng.*, vol. 19, no. 12, pp. 12866–12896, 2022, doi: 10.3934/mbe.2022602.