Detecting cheating in electronic exams using the artificial intelligence approach

Bashar H. Asker **Ahmad F. Al-allaf**

NTU, Technical Engineering College, Dept. of Computer Eng. Mosul, IRAQ

Abstract **-Today, e-learning has become a reality and a global trend imposed and accelerated by the Covid-19 pandemic. However, there are many risks and challenges related to the credibility of online exams which are of widespread concern to educational institutions around the world. In this paper, artificial intelligence techniques were used to design a system to monitor and prevent attempts to cheat in electronic exams. The proposed system relies on the webcam installed on the examinee's computer to capture videos of the examinee in real time and then process them using artificial intelligence and deep learning techniques. The proposed system performs a set of tasks, including detecting the identity of the examinee, the presence and absence of the examinee, the presence of more than one person next to the examinee, attempts to use a cell phone, and tracking the examinee's eye movement. A data set of (20) subjects representing real-world behaviors in the online exam was collected to test the system. A detection rate of close to (93.9%) was achieved for all types of cheating behavior with a fixed FAR rate of (5%).**

Keywords-Online Examination, face detection, phone cell detection, pupil tracking.

I. INTRODUCTION

E-learning is education provided on the Internet, using modern electronic technologies to access everything related to educational materials outside the boundaries of the educational classroom. One of the advantages of e-learning is that it reduces costs, is available to all individuals and different age groups, and is flexible because there are no time-related links. It also provides a neutral and structured education that tracks each student's achievements and logs their activities online and is considered environmentally friendly [1].

To raise e-learning quality standards, attention should be given to electronic exams. With the spread of the Corona pandemic, and educational institutions resorting to e-learning and electronic assessments, and due to the lack of direct supervision of students during exams; The phenomenon of electronic cheating spread among students, occupied academics, and negatively affected the credibility of e-learning and its results, and the scientific reputation of educational institutions.

Just as a physical proctor is necessary in a traditional classroom environment, the requirement for an electronic proctor is also essential in online electronic exams. So that this system can detect suspicious activities that may occur or reduce the chances of them happening.

The designed electronic exam monitoring system uses a webcam installed on a laptop computer connected to the Internet. The main features of the system are:

1-Providing a fully automated electronic monitoring environment for exams.

2 - Not to use any external devices that are expensive and not suitable for students.

3 - No need for a physical observer during the exam period.

It should be noted that sudden individual behavior for a limited period of time cannot be interpreted as suspicious behavior, as short and sudden movements should be neglected. While some repetitive head nods or some changes in eye movement to a specific position for a certain period of time may indicate an attempt to cheat. [2]

The aim of this paper is to design a system to monitor student's behavior and look for malpractices in online exam using a webcam. The system uses a set of libraries available in the Python language for the purpose of verifying the identity of the examinee student, discovering an attempt to use a cell phone, discovering the presence of more than one person in the exam room, and so on.

2. LITERATURE REVIEW

There are many researches dealing with the topic of preventing cheating in electronic exams, including:

In [3], Atoum et al. Introduced an online screening monitoring system using a multimedia analytics system. The system uses two cameras and a microphone and from the captured video and audio clips can extract a set of features such as user identification, text detection, speech detection, and pupils tracking. A dataset of (24) persons was collected. The researcher claims that an approximately (87%) detection rate was achieved for all types of cheating behavior with a flat rate of (2%).

In [4], Aiman Kiun uses three models, VGG16, Inception-v4, and Cell phoneNets to the video recordings of electronic examinations. These models achieved an accuracy of (96.8%) when using the traditional method of training the classier. While the use of the episode-constrained cross-validation achieved less accurate results up to (67.1%).

In [2], Prathish et al. used a webcam to take pictures, audio and video capture with the active window constitutes the input to a dataset-based inference system to determine if there are attempts to cheat during the electronic exam. In this paper, characteristic points are extracted from the examinee's face and then used to estimate head position. Suspicious behavior is also detected based on differences in yaw angle, presence of sound.

The paper presented by Tiong et al. [5] discusses the mechanisms to reduce cheating in electronic exams using artificial intelligence technology. Four deep learning algorithms, DNN, DenseLSTM, LSTM, and RNN, were used on two exam datasets for the purpose of monitoring online cheating. The researchers claim an overall accuracy of (95.32%) was achieved by DenseLSTM. Average accuracy rate (90%).

Garg et al. [6] uses Convolutional Neural Network (CNN) to monitor the electronic exam. In this system, the camera on the student's computer used to track examinee face during the exam and you monitor his behavior to prevent any suspicious practices. The system tracks and identifies students' faces using the Haar Cascade Classifier and deep learning. The system can also detect multiple faces. This proposed system has recognition rate (93%).

Mathapati et al. [7] suggested a self-picture password scheme to prevent online exam cheating. In this paper, the self-picture is used as a password called a graphic password with personal physical codes in the form of digital pictures captured from a video camera. The visual features were extracted from the picture and used as a password.

Hu et al. [8] proposed a convolutional neural network to estimate head and mouth movement to detect abnormal behavior of a person during the electronic examination. The system has a false alarm rate of (5%) recognition rate of (90%).

3.SPECIFICATIONS AND ORGANIZATION OF THE PROPOSED SYSTEM

The test is created online, which can contain all kinds of questions such as short or multiple-choice questions, and so on. The system requires the presence of a computer installed in a suitable room for testing with a webcam. Before the exam begins, the identity of the examining student is identified by comparing the image of the student’s face with the images stored in the database in the exam center at the student’s college. The process of identifying the student’s identity is repeated in different periods throughout the exam period. In addition, a number of other requirements must be met for the system to function properly, and they are as follows:

1- There is no one inside the exam room other than the examinee student.
2- Do not turn off the camera during the exam.
3- Not to leave the exam room before submitting the exam.
4- Limiting the movement of the student during the exam. It is not allowed to look left or right for a period exceeding a predetermined time.
5- Not to use a cell phone or tablet during the exam.
6- The identification is made all the exam period and compared to the pictures in the dataset to ensure that the unaccredited examinee does not enter the exam.
7- Provides good lighting in the exam room and that the distance between his eyes and the webcam should be about (50) cm.

The system is able to run smoothly on a medium specification system. Where it can be implemented on computer specifications of 4th generation Intel I5 processor, (12) GB RAM, SSD hard drive and Intel HD Graphics (4600) internal

The video camera is set at (30) frames per second.

The proposed system consists of the following units:

- Identification of the examinee using face detection algorithms.
- Detection of the presence of one or more persons other than the examinee.
- Detecting attempts to look at the sides or above with the aim of cheating by observing the movement of the examinee’s pupils.
- Detection of an attempt to use a mobile phone or other tablet device.

The flowchart shown in Figure (1) shows the interconnection and functioning of the system units.

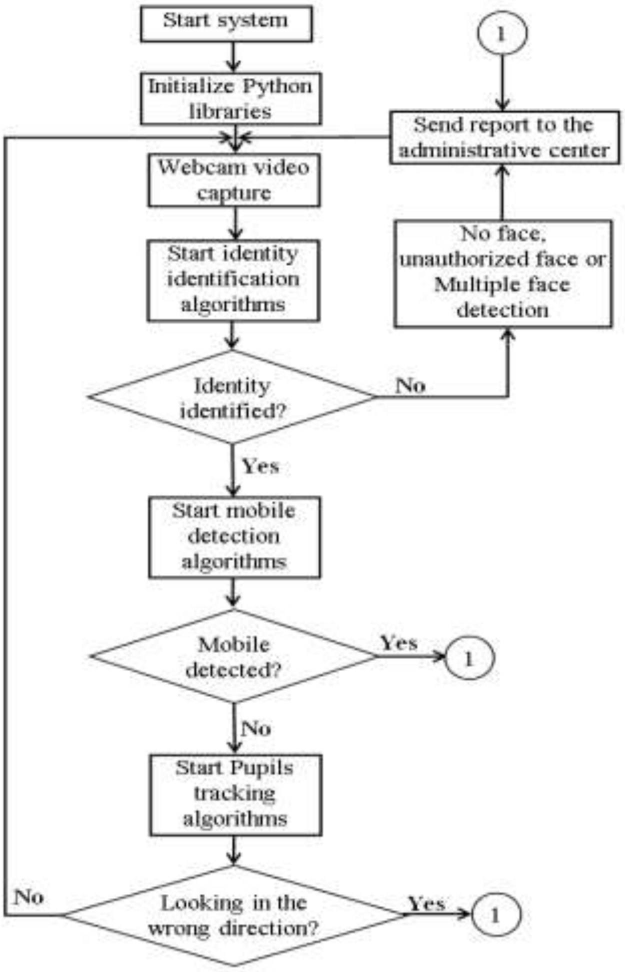
Start system
Initialize Python libraries
Webcam video capture
Start identity identification algorithms
Identity identified?
No
Yes
Start mobile detection algorithms
Mobile detected?
Yes
No
Start Pupils tracking algorithms
Looking in the wrong direction?
No
Yes
1
Send report to the administrative center
No face, unauthorized face or Multiple face detection

Figure 1. Organization of system modules

The proposed system uses a set of Python language libraries, which are:

- **OpenCV2 library** (Open Source Computer Vision Library) is a free and open source software library used in computer vision and machine learning applications [9, 10].
- **Face Recognition library which** used to recognize and process faces [11].
- **NumPy library (**Numerical Python), it is external library that comprises array-related methods in addition to a strong multi-dimensional array object [12].

4.THE DATASET

The aim of the current study is to monitor the student during the electronic exam by detecting and identifying the student's faces and behavior during the exam. Publicly available online data sets were not used because they may not meet all of the sitting positions and examinee movements required in this study, the images may be inappropriate, or some are low resolution even with images from the large data set. So our dataset was created, which contains a sufficient number of images representing the different situations (allowed and not allowed) of the student during the test for the purpose of testing the system that was designed. When taking pictures, the previously mentioned conditions, such as lighting and distance from the camera, were taken into account. The images were taken in the normal position as well as when moving the head in the four directions (up, down, left and right) with a deviation of an angle of (15 to 75) degrees. In addition to creating other images related to the student's attempts to use the mobile phone.

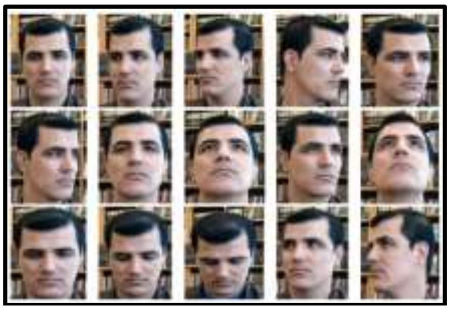

Figure 2: Sample of the created dataset for face recognition

The generated dataset contains about (200) images of twenty people (at least 7 images of each person in different poses). Figure 2 shows some of these images.

5. THE IMPLEMENTED MODULES OF THE PROPOSED ELECTRONIC MONITORING SYSTEM

5.1 Identity identification of the examining student

Python language (version 3.10.0) and libraries like CV2, SKlearn, Face recognition and NumPy with KNN (K nearest classifier) algorithm were used to detect and distinguish the examinee's face as well as detect the presence of more than one person inside the exam room. At system startup, the identity detection software will be executed and this task will be repeated throughout the test period, using a webcam with a resolution 2.1 MP (1920 * 1080 at 30 FPS). The software extracts a frame from the camera's video and begins to recognize faces in that frame by comparing it to the set of images in the previously prepared database. A cluster-based, K-nearest-neighbors-classified face recognition system and bootstrap assembly were used [13] as shown in the Figure 3.

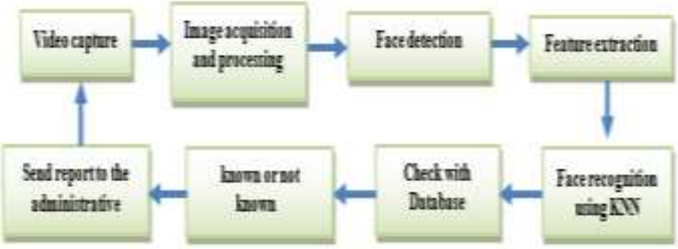
Video capture
Image acquisition and processing
Face detection
Feature extraction
Face recognition using KNN
Check with Database
known or not known
Send reports to the administrative

Figure 3. Steps for student identification

KNN is derived and based on the Near Classifier (NN) system. This classifier is based on a simple non-parametric decision. The distance between the features of each query image and the features of the other images in the training data set is analyzed. The image with the shortest distance from the query image in the feature space is the closest adjacent image. Euclidean distance used to calculate the distance between two features. [13]

First, the KNN classifier trains on a set of faces, then it can detect the face of the examinee in a live video by selecting K, that is, the images with the closest facial features at Euclidean distance. The Euclidean distance is the length of the line connecting two points. The Pythagorean theorem is used to determine it from the Cartesian coordinates of these two points [14]. The weighted voting used in KNN, therefore, the votes of the closest neighbors are more weighted, as in figure 4[15].

For example, if k = 3, and the nearest three face images to the selected image in the training set of one person and two images of another person, the result will be the first person. To use KNN, a set of images is first prepared for the authorized people we want to identify. Organize the photos in one folder with a subfolder for each examinee. Then the "train" function is called using the appropriate parameters. 'model_save_path' has been saved, so the model can be reused without having to retrain it again. Finally, the "predict" function is called by the trained model passed to recognize the people in the video. The test results showed that the accuracy reached (90%).

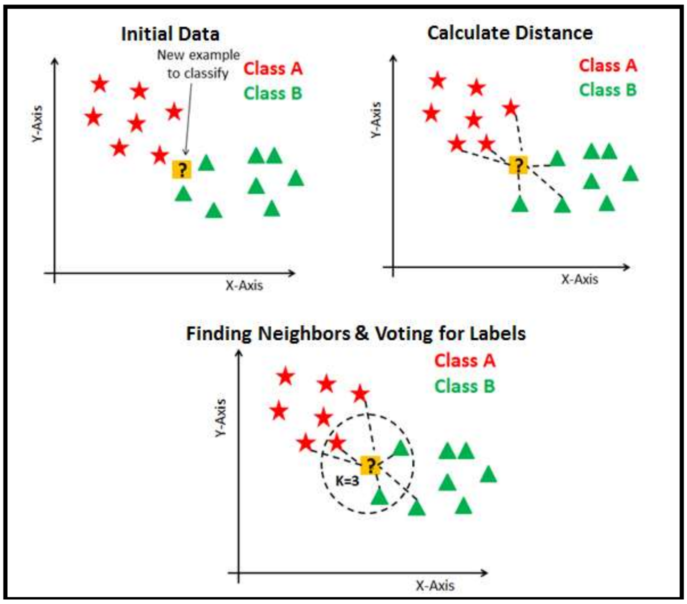

Figure 4. Example on KNN with K=3

5.2 Appearance/disappearance of the examinee's face

The process of recognizing the examinee's face is essential to ensure the reliability and integrity of the exam. As an example, in Figure 5, a 90-second video sample of the examinee's face was used at a rate of about (30) frames per second, so we will have (2700) frames in the (90) seconds. Since the x-axis represents the number of seconds, while the y-axis represents the appearance or non-appearance of the examinee's face and has two values, positive means the presence of the examinee's face and negative means the absence of the examinee's face from the screen. As it is clear from the figure, during the seconds (11 and 12) and (38 to 41), that is, within (6) seconds, or (6.66%) of the time, the examinee's face did not appear on the screen. While during the remaining time, that is (84) seconds, which is equivalent to (93.33%) of the time, it represents the appearance of the examinee's face on the screen. This information is sent to the exam administration center to make the right decision (the presence of cheating or not).

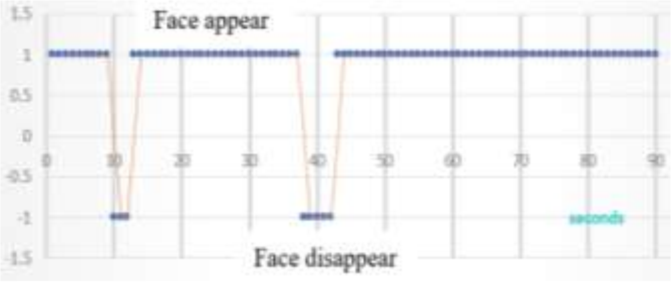

Figure 5 Appearance/disappearance of examinee's face in a video sample

5.3 Cell phone detection

The proposed system prohibits the use of any type of mobile phone. The use of a cell phone by the examinee is considered an attempt to cheat. There are many ways to cheat using a mobile phone such as reading notes, calling or texting people, searching the internet and taking a screenshot of the test to share with other examinees. There are some difficulties in detecting the phone due to the different sizes, colors and shapes of mobile phones. In this paper, TensorFlow library was used with YOLO-V3 to detect the presence of a cell phone. [3]

- **YOLO-V3**:

You only look once, or YOLO, is a fast algorithm to find out what's in a place. It is trained on the COCO dataset. Although it is not the most accurate object detection algorithm, it is very good when we need to detect objects in real time with good accuracy. Figure 6 shows architecture of YOLO-v3 [16].

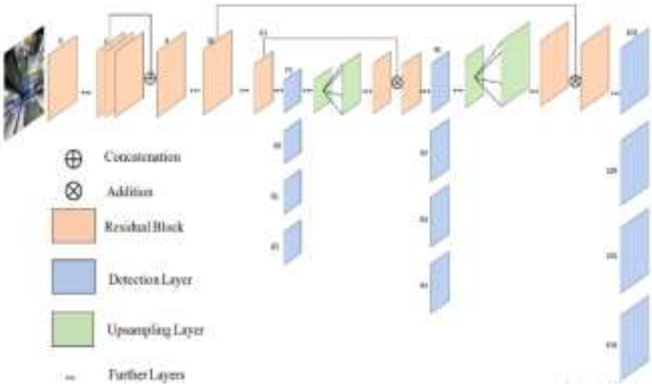

Figure 6. YOLO-V3 architecture

It contains (53) detection convolutional layers, each layer followed by batch normalization and ReLU activation, we will be 106 layers in the YOLO-V3 architecture [17]. YOLOv3 can detect and classify (80) objects (such as a person, bicycle, car, motorcycle, plane, bus, train, etc.)

The rectified linear activation function (ReLU) is a piecewise linear function that its output will be the direct input if its value is positive, otherwise its output will be zero (see Figure 7). To perform the detection, a 1 x 1 detection kernel is applied to feature maps of three different sizes at three different places in the network. The shape of the detection kernel is 1 x 1 x (b x (5 + c)). For a feature map prediction by variable B the number of bounding squares for a cell is represented on the feature map. where 5 represents the four bounding box attributes, and C represents the number of classes. In YOLO v3 trained on COCO, B = 3 and C = 80, so the size of the kernel is 1 x 1 x 255.

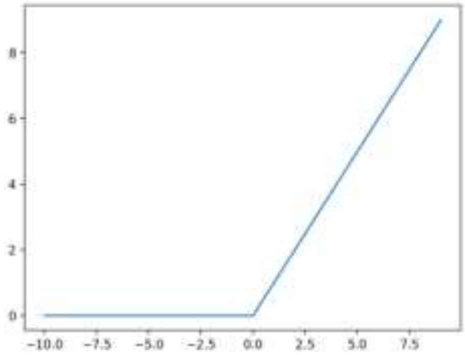

Figure 7. ReLU activation function

The first inspection is done through layer No. (82). For the first 81 layers, the network is sampled from the image down, so that this layer (81) contains a step (32). For example, for an image with a resolution of (416 x 416), the resulting feature map will be a size of (13 x 13). One detection here uses a 1 x 1 detection kernel, which gives us a 13 x 13 x 255 detection feature map. That is, a feature map from Layer 79 will expose a batch of convolutional layers before sampling them 2x to 26 x 26 dimensions. The resulting feature map is then deeply bound to a feature map from Layer 61. Once again, the aggregated feature maps undergo a few 1 x 1 convolutional layers to combine features. from the previous layer. The second detection is done by layer No. 94 to generate a 26 x 26 x 255 detection feature map.

Repeat the same process again. Where the feature map in layer 91 is subjected to small convolutional layers and then the depth is set to the feature map from layer 36. As in the previous case, some 1×1 convolutional layers follow the merging of information from the previous layer we make the final of the 3 in layer 106, to generate a feature map of size 52 x 52 x 255. YOLO v3 predicts squares at three different scales [17]. We used phone class which is number (67) in the arrangement of classes and (yolov3.weights) which is the original weights file of YOLOv3 that pretrained using COCO dataset. Detection speed of YOLO-V3 is (29) ms/frame when input dimensions was (416×416). Figure 8 states a comparison according to mean Average Precision and detection speed time for object detection algorithms. The test shows that the accuracy is (41 to 91) % for mobile detection.

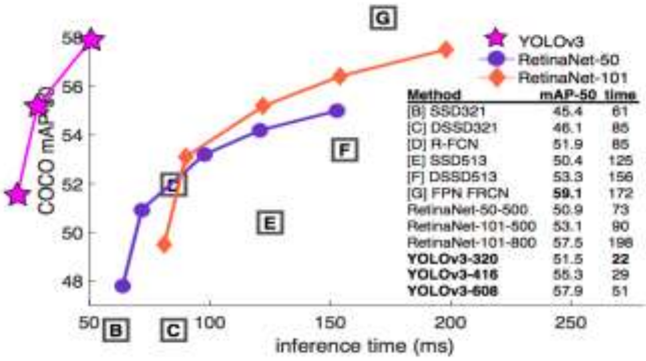

Figure 8. Average Precision comparison of object detection algorithms [18]

5.4 pupils tracking

MediaPipe Iris is a machine learning algorithm for iris estimation that tracks iris landmarks, pupil and eye contours in real time using the camera and without specialized hardware. It can determine the metric distance between object and camera with less than (10%) error, as shown in Figure 9 [19].

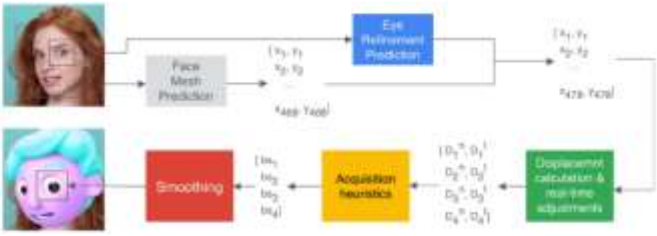
Face Mesh Prediction
Eye Refinement Prediction
Displacement calculation & real-time adjustments
Acquisition heuristics
Smoothing

Figure 9. Overview of the pupil blend shapes acquisition

Using libraries such as (Mediapipe, CV2, math and NumPy) and Euclidean distance with (16) points (362, 382, 381, 380, 374, 373, 390, 249, 263, 466, 388, 387, 386, 385,384, 398) for left eye and 16 points (33, 7, 163, 144, 145, 153, 154, 155, 133, 173, 157, 158, 159, 160, 161, 246) for right eye, as in Figure 10[20].

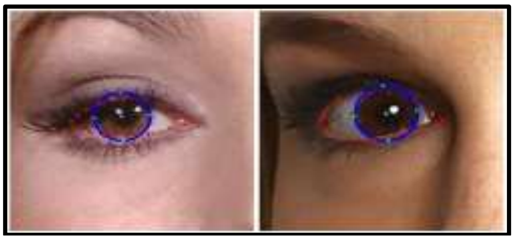

Figure 10: Eye region (red points) and iris region (blue points)

For a black mask, the eye area is converted to white while the pupil image is black. The threshold used to create a binary mask and its value will be different according to the lighting, (60) chosen as the default for webcam and room lighting. Now the object to be found should be white and the background black. Then find the position of the pupils (right or left). As shown in the Figure 11.

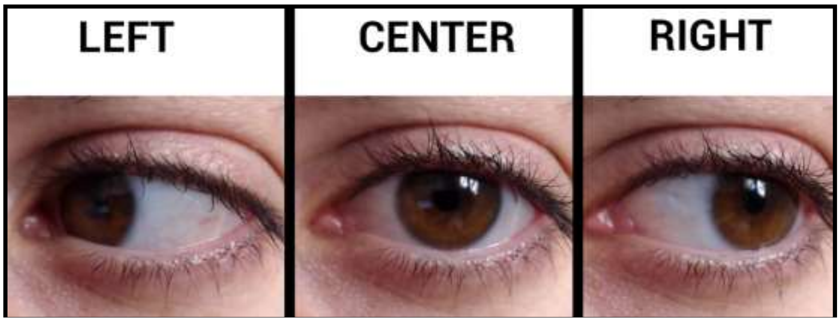

Figure 11. Pupil position in left and right directions

The results were good and mean average accuracy is (92.5%) for (20) persons, as shown in Figure 12, where the X-axis represents number of persons, and the Y-axis represents pupil track accuracy. We note that the accuracy changes between (75 to 100)% due to sampling in different places, different lighting levels, and different people. During the exam period, the images showing cheating will be stored in separate folders and sent to the exam adminstration center.

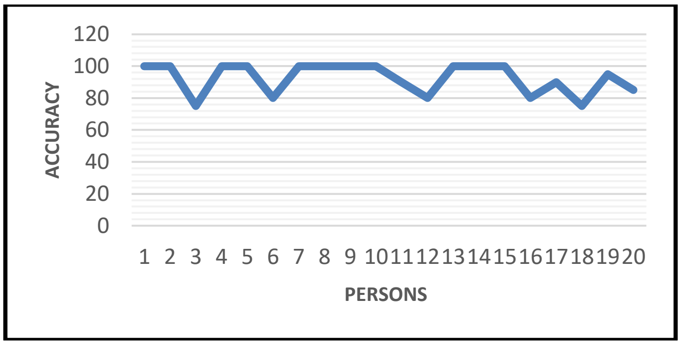

Figure 12. Pupil track accuracy

6. RESULTS AND DISCUSSIONS

Although the classifiers like CNN (Convolutional Neural Network) will offer us significantly greater accuracy, especially for non-frontal facing faces and partly occluded faces. However; the CNN was not used in the proposed system because the system is designed to detect specific positions of the examinee, and good lighting must be provided, as stated in (Specifications and Organization of the Proposed System) (clause 7) in this paper. So, KNN was used in face recognition. For accurate results, photos must be well-lit and high-resolution. The tests of applying the system on the webcam images of the cases studied in this research showed that the accuracy varies according to the type of the object, and this is due to the number of features that can be detected by the object, as well as to the type of algorithms used in the detection and resolution of the webcam used in our system.

For example, the accuracy of detecting the absence of a face was very high, reaching (100%). While the accuracy of facial recognition was up to (90%). As for the accuracy of detecting the presence of more than one person, it was (100%). As for detecting the use of mobile phones, the detection accuracy has reached (81%), which is not ideal because the shapes of mobile phones are many and differ from each other in terms of size, shape and color. Finally, for the iris tracking, the detection accuracy reached (92.5%) for the left and right directions, and it is good result because the distance between the examiner and the webcam was (50) cm and good lighting, as mentioned in the system design assumptions. The average accuracy for all systems was (93.9%) as shown in Figure 13.

7.CONCLUSION

Recently, interest in e-learning has increased, and one of the most important parts of e-learning is electronic exams. To give credibility to these tests, there must be monitoring and supervision of these tests to prevent cheating attempts. In this research, an electronic system based on deep learning mechanisms was designed to monitor the student during the electronic test. A set of requirements have been set for the system to function properly, including providing good lighting, as well as not allowing the student to make some movements, which are considered an attempt to cheat.

It only requires a laptop and an external webcam with a resolution of at least 2.1 MP (1920 * 1080 at 30 FPS) to get good results. During the work of the system, the real-time video is captured from the webcam, then the video is processed to extract six features: no face detection, face detection, more than one face detection, and recognition of the examinee's face, Tracking the head movement of the examinee through tracking the movement of the examinee Pupil, as well as detecting trying to use a cell phone.

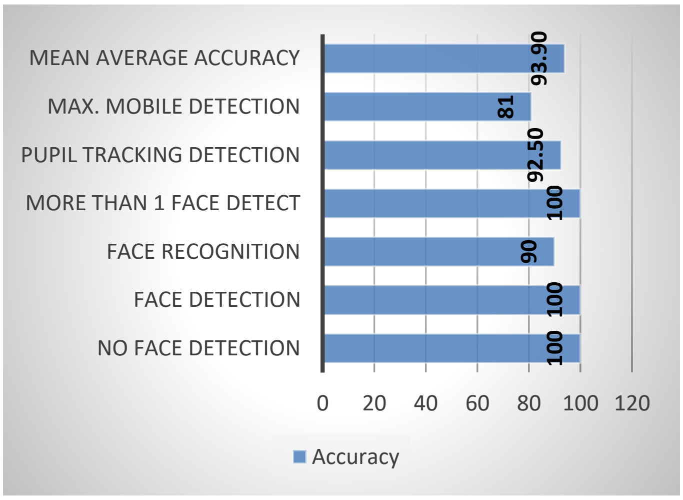

Figure 13. Item accuracy and mean average system accuracy

These features are handled by using different algorithms to detect cheating cases and send a report directly to the exam supervision center. The results obtained indicate that the proposed system was effective in supervising the exam and contributed significantly to reducing cheating attempts during the electronic exam. The average accuracy of the proposed system's detection of the six advantages mentioned above was approximately (93.9%).

REFERENCES

[1] Sander Tamm, "10 Major Advantages of E-Learning" e-student.org, Nov 13, 2021, https://e-student.org/advantages-of-e-learning/ .

[2] Swathi Prathish, Athi Narayanan and Kamal Bijlani, "*An Intelligent System for Online Exam Monitoring* ",International Conference on Information Science (ICIS), 2016.

[3] Yousef Atoum, Liping Chen, Alex X., Stephen Hsu and Xiaoming Liu, "Automated Online Exam Proctoring", pp.1-15, December 2018.

[4] Aiman Kuin, "Fraud detection in video recordings of exams using Convolutional Neural Networks", M.S. thesis, university of Amsterdam, Amsterdam, 2018.

[5] Leslie Ching and HeeJeong Jasmine, "E-cheating Prevention Measures: Detection of Cheating at Online Examinations Using Deep Learning Approach - A Case Study ", *LTEAX CLASS FILES*, pp.1-9, Jan. 2021.

[6] Kavish Garg, Kunal Verma, Kunal Patidar, Nitesh Tejra and Kunal Patidar, "Convolutional Neural Network based Virtual Exam Controller", *International Conference on Intelligent Computing and Control Systems (ICICCS 2020).*

[7] M. Mathapati, T. Senthil, A. Krishna and S. Vinoth, "Secure Online Examination by using Graphical Own Image Password Scheme", *IEEE International Conference on Smart Technologies and Management for Computing, Communication, Controls, Energy and Materials (ICSTM)*, 2017.

[8] Senbo Hu, Xiao Jia and Yingliang Fu, "Research on Abnormal Behavior Detection of Online Examination Based on Image Information", *International Conference on Intelligent Human-Machine Systems and Cybernetics*, 2018.

[9] openCV team, "about openCV", https://opencv.org/about/.

[10] Natali Almeida, "Facial Recognition System applied to Multipurpose Assistance robot for Social Human-robot Interaction (MASHI)", M.S. thesis, Escuela Técnica Superior de Ingeniería Industrial de Barcelona, Barcelona, 2017.

[11] Adam Geitgey, "face recognition",https://face-recognition.readthedocs.io/en/latest/readme.html. GitHub, Inc.

[12] TechVidvan team, "Python NumPy Tutorial for Data Science", https://techvidvan.com/tutorials/python-numpy-tutorial/.

[13] Ebrahimpour, H.; Kouzani, A.," Face Recognition Using Bagging KNN", Proceedings of the International Conference on Signal Processing and Communication Systems, 17–19 2007

[14] David Eppstein, "Euclidean distance", https://en.m.wikipedia.org/wiki/Euclidean_distance.

[15] João Leal "GPU based Dynamic k-Nearest Neighbours", https://github.com/artifabrian/ dynamic-knn-gpu , 2018 GitHub, Inc.

[16] Yuan Dai, Weiming Liu, Haiyu Li, and Lan Liu “Efficient Foreign Object Detection Between PSDs and Metro Doors via Deep Neural Networks”, IEEE Access PP(99):1-1, March 2020.

[17] Ayoosh Kathuria, “What’s new in YOLO v3?”, https://towardsdatascience.com/yolo-v3-object-detection-53fb7d3bfe6b, 2018.

[18] Yolo V3, 2019 : https://supervise.ly/explore/models/yolo-v-3-coco-1849/overview.

[19] Artsiom Ablavatski, Andrey Vakunov, Ivan Grishchenko, Karthik Raveendran and Matsvei Zhdanovich, “Real-time Pupil Tracking from Monocular Video for Digital Puppetry” Proceedings of the CVPR Workshop on Computer Vision for AR/VR 2020.

[20] https://medium.com/axinc-ai/mediapipe-iris-detecting-key-points-in-the-eye-637f5c1e728e Apr 14, 2021.