A Survey of Deep Learning Algorithms and YOLO Models for Online Examination Proctoring

[1]ASRA SARWATH, [2]NAAZNEEN TARANNUM, [3]WAJEEHA AREEN, [4]FAIZA TAHREEM, [5]SABIHA NUZHAT
Department of Computer Science and Engineering
Faculty of Engineering and Technology
Khaja Bandanawaz University, Kalaburagi, Karnataka

***Abstract*:** Proctored exams are timed exams that you take while proctoring software monitors your computer's desktop, webcam video and audio. Due to COVID 19 Pandemic the Online exams were the only option left for examination but monitoring of students during exams is primarily challenging due to lack of physical presence so that's why we need to develop methods that provide ways to detect unfair, unethical, and illegal behaviour during exams. The data recorded by the proctoring software is transferred to a proctoring service for review. Education Sector use Online Examination Proctoring to protect forbidden and ill-suited actions of student to serve fair justice among all. These prohibited issues are caused by various types of cheating such as disconnecting, impersonation, Advanced tech gadgets, a changing tabs, The `old school` tricks, and so on. We analyzed and reviewed the use of deep learning algorithms for Online Examination Proctoring in this survey. Because data are so important in Deep Learning (DL) methods, we describe some of the most commonly used network datasets in DL, discuss the challenges of using DL for Online Examination Proctoring, and make recommendations for future research.

***Keywords* – prohibited actions, online examination proctoring, deep Learning, machine learning, real-time datasets**

I. INTRODUCTION

According to a recent survey [1], more than 7.1 million students are taking, at least, one online course in 2013 in America. It also states that 70% of higher education institutions believe that online education is a critical component of their long-term strategy. Exams are a critical component of any educational program, and online educational programs are no exception. In any exam, there is a possibility of cheating, and therefore, its detection and prevention are important. Educational credentials must reflect actual learning in order to retain their value to society. The authors in state that the percentage of student exams committing academic cheating activity is on the rise. Nearly 74% of students in 2013 indicated that it would be somewhat easy to cheat in online exams.

They also found that in 2013, about 29% of the students admitted to cheating in online exams. When exams are administered in a conventional and proctored classroom environment, the students are monitored by a human proctor throughout the exam. In contrast, there is no convenient way to provide human proctors in online exams. As a consequence, there is no reliable way to ensure against cheating. Without the ability to proctor online exams in a convenient, inexpensive, and reliable manner, it is difficult for MOOC providers to offer reasonable assurance that the student has learned the material, which is one of the key outcomes of any educational program, including online education. A typical testing procedure for online learners is the following: students come to an on-campus or university-certified testing centre and take an exam under human proctoring.

In the last few years, a various method had been proposed to overcome the needed in online exam proctoring. The methods proposed as an effort of reducing cheating occurrence when an online exam being held and keep the academic integrity. Those methods can be classified into four categories:

a) no proctoring;
b) human online proctoring;
c) semiautomatic proctoring; and
d) fully automated online exam proctoring.

The COVID-19 pandemic has caused disruption in our lives. This situation has forced school, university, and other education and training around the world to move to online learning and hence online examination. However, many challenges such as unfaithful and ill-suited activities are prohibiting its wide adoption by the governments and public. So, there is a need to develop techniques to prevent such activities but current availability with this respect is limited with most of the software available from commercial entities that provide limited and "non-open" software tools. Many open-source tools and efforts are needed to bring innovation, variety, and richness to this online learning software systems domain. Artificial intelligence (AI) has revolutionized the world by providing many smart solutions to deal with many day-to-day problems and one of them is online examination proctoring.

This paper put forward the online examination proctoring system to cope up with the absence of physical proctor by making use of deep learning to continually proctor physical places. This system exploits biometric approaches including face recognition using the HOG face detector and the OpenCV face recognition algorithm. Moreover, to ensure fairness during online exams, the system is able to detect gadgets including mobile phones, laptops, iPads, and books and as well as the detection of 2nd person during session. And this also detects the change of tabs and pressing any key in the keyboard of laptops. This online examination proctoring system is implemented as a software system and evaluated using the FDDB (Face Detection Data Set and Benchmark) and LFW (Labelled Faces in the Wild) datasets.

The rest of the paper is structured as follows. Section II discusses the LITERATURE SURVEY to online proctoring systems. Section III describes the RELATED WORKS. Section IV provides system evaluation. Section V concludes and discusses future work.

II. LITERATURE SURVEY

Asra et al. [1] The previous articles did not cover the wide range of cyber security datasets used, as well as the flaws in these deep learning techniques. As a result, the primary goal of this work was to present a bibliometric analysis of the deep learning approach used for detecting potential cyber security threats. A comparative analysis is then performed to review the various attacks encountered, the various platforms used, datasets, and learning models developed by various researchers in the field of cyber security using Deep Learning. This survey has also addressed existing research challenges, open issues, and future research directions.

Tayeb et al. [2] constituted two major steps. In the training phase the first step it employed was the dimension reduction and the second step was the Convolutional Neural Network hyperspectral image classification. And to evaluate the proposed CNN model, two well-known real world data sets were used in this experiment.

Asep Hadian S. G et al. [5] proposed a method in continuous user verification based on face verifications by implementing an incremental training process using images captured from m learning online lecture sessions as training data set in order to increase the robustness against variations of pose and lighting. The algorithm is trained each time a user finished his lecture session. The method proposed is expected to increase the verification accuracy without applying additional processing

Istiak et al. [9] proposes a novel online proctoring system that uses deep learning to continually proctor physical places without the need for the presence of a physical proctor. The system employs biometric approaches including face recognition using the HOG face detector and the OpenCV face recognition algorithm. The system is implemented as a software system.

Yousef Atoum et al. [13] introduced a multimedia analytics system to perform automatic and continuous online exam proctoring (OEP). this system monitors cues in the room where the test taker resides, using two cameras and a microphone the first camera is located above or integrated with the monitor facing the test taker. The other camera can be worn or attached to eyeglasses, capturing the field of view of the test taker. In this paper, these two cameras are referred to as the ″webcam″ and ″wearcam″ respectively. The webcam also has a built-in microphone to capture any sound in the room. they proposed a hybrid two-stage algorithm for our OEP system. The first stage focuses on extracting middle-level features from audio-visual streams

that are indicative of cheating. In the second stage, a joint decision across all components is carried out by extracting high-level temporal features from the OEP components at the first stage.

MIKEL et al. [14] presents a new system based on web applications which offer a continuous authentication identity service of online students through a constant biometric (face, voice, typing) recognition system (biometric traits cannot be lost, stolen, or recreated), as well as automatic continuous proctoring through automatic image and audio processing (device monitoring & lock-down and inappropriate behaviour detection) allowing online courses to gain value of what benefits both institutions and students. This solution is based on a high accuracy biometrics recognition and digital signal processing algorithms and it is complemented with human supervision for those situations in which the automatic algorithms are not able to determine reliable results.

Muhanad et al. [41] surveyed current proctoring systems based on artificial intelligence, machine learning, and deep learning is presented in this work. There were 41 publications listed from 2016 to 2022 after a comprehensive search on Web of Science, Scopus, and IEEE archives. We focused on three key study questions: current approaches for AI-based proctoring systems, techniques/algorithms to be employed, datasets used, and cheating detection methods suggested in such systems. Analysis of AI-based proctoring systems demonstrates a lack of training in using technologies, methodologies, and more.

Prathmesh et al. [25] Focused on the online examination system developed with the goal to make online examinations more accessible and reliable using deep learning models for the proctoring system. It also covers the various technologies and languages used in the development process, including but not limited to HTML3, CSS5, BOOTSTRAP5, Django, Python. The developed system is reliably able to detect and counter any attempts at cheating during the exam, and provides a user-friendly system interface with focus on ease of use and simplicity.

Alice et al. [30] This study involved a private university that had managed to carry out Emergency Remote Teaching during COVID-19 and still go ahead to conduct examinations. There was a methodical way of migrating the examinations that even afforded to survey the technologies that users had access to. Given that this this involved the use of technology but also challenged the organizational factors, it is apparent that the Online Examination and Proctoring Systems should be aligned to the University, and the University should be aligned to the Systems. Reorganization of the examination processes, organizational and cultural change management, ICT technical issues, extensive training, software selection and communication are the key requirements for successful online examinations and proctoring to take place.

Bakhitzhan et al. [31] devoted to the study of automated reading detection from the camera in an online exam. Two state of-the-art deep learning models, ConvLSTM and LRCN, were adapted to address the reading detection task. To train the models, they collected their own dataset of videos with a total length of thirty-seven minutes and 10% of the dataset was allocated to test the performance of the proposed models. For each model, the training was carried out in two different cases where in the first case no data augmentation is carried and in the second case, several video augmentation techniques were used with the aim to improve the performance. In all cases, the models performed excellently with LRCN reaching 100% accuracy for the augmented dataset in all four metrics. Since the performances of the models are satisfactory, they have not used other possible approaches. In particular, skeleton or biometric-based action recognition may be adapted for similar tasks. there model was trained for color videos with frames converted into RGB format.

Tanzila et al. [37] described the proposed model and the steps involved to perform EAR. The main steps of the proposed model include pre-training, Dataset augmentation, feature extraction from pre-trained L2-GraftNet, Feature subset selection from ASO based features subset selection, and classification. And provides a brief overview of the proposed model of EAR using the proposed CNN network.

III. RELATED WORKS

Methods had been proposed to overcome the needed in online exam proctoring. eight constraint procedures are proposed to minimize the cheating happens, while a method proposed by sending a real-time image of every classroom to streaming media server. Then a monitor data consists of video and audio streaming also screen snapshot capture continuously and the participant is verified with the data captured during sign in. a desktop robot with a 360o camera and movement sensor attached sends video recording to a monitoring center when compromising events detected. Then a fully automated online and continuous proctoring was proposed to detect some cheating behaviour of online exam participants, including impostoring, where the valid participant is being replaced by others to complete the exam.

According to [83], biometrics is a science that shows an individual's identity based on the attributes of a person (physical, chemical or behavioral). The needs for biometrics today are followed by the need for systems that depends on the accuracy of individual identity authentication. As for this work, the biometric to be used is face biometric captured continuously from the user available during the online lecture sessions and exam session.

In [84], the user image is captured by the camera for some time to build the user template, and then the system will detect if there is a person's face caught on camera to be verified using PCA and eigenface. The image size used for verification is 100x100 pixels. We should remind that not all data in the image are useful for face recognition, and more data will decrease the speed of the system and increase the use of storage media. One remaining problem stated in the previous researches is the system robustness for lighting and poses variations. Several methods had been proposed to overcome the lighting variations in the previous works. The typical methods used are histogram equalization, histogram specification, gradient, and gamma correction. As for the poses variations some methods had been proposed.

The online exam is facing immense challenges throughout the exam. Sarrayrih et al. [16] discussed the several challenges presented by the online exam, as well as providing a solution by grouping the hostnames or IPs of clients for a specific location and time, with a biometric solution like face recognition and fingerprints. In [17],a profile-based authentication framework is proposed for the online exam based on different challenging questions, including the favourite questions, personal questions, and an academic question. Fenu et al. [18] proposed a multi bio-metric continuous authentication system including face recognition, voice recognition, touch recognition, mouse, and keystroke in 2018. Selvi et al. [19] designs and implements a firewall security system using different firewall technologies, including Network Address Translation, Demilitarized Zone, and Virtual Protocol Network , which are used for intrusion detection. Wei et al. [20] proposed fingerprint-based solution. Garg et al. [21] proposed a face recognition and detection solution for the secured online exam using deep learning. Another online proctoring system was proposed by Atoum et al. [22], which continuously estimates six components, including voice, phone, text, and active window detection, estimation, and user verification. A fingerprint and eye tracker-based online test management system was proposed by Bawarith et al. [23]. Cheating and not cheating are used as student status to evaluate their proposed methodology.

Mettl created web-based online proctoring software that divides their system into four major components: candidate authentication using a picture, OTP and ID affirmation, human-based proctoring using real-time recording in the classroom, secure browser-based proctoring using disables the following features: opening new browsers and data transferring media, and AI-based proctoring using facial, mobile phone, candidate distraction, and multiple person detection. To use this software, the examiners have to pay.

IV. Deep Learning for Online Proctoring System

A. The Proposed Framework

The proposed web-based online proctoring system is distributed into two modules. Firstly, the online registration part, and secondly, the online proctoring part describes the proposed architecture for the online proctoring system.

1) Online Registration: For registering students' faces, we accessed the student's web-camera through HTTPS protocol during registration and captured the students' faces, storing

the face information in the database. W++++++++e used a flask microframework for web development.
The primary challenge in this module was accessing the client-side camera to capture the students' faces. To get around this, we utilized the HTTPS (HyperText Transfer Protocol Secure) protocol to access the students' webcams, which encrypts all conversations between the browser and the server. When using the HTTPS protocol to host a website, SSL certificates are also required. We used a self-signed SSL certificate to run the server.
2) Online Proctoring: During online exam sessions, there are some challenges to conducting the exam. Challenges are:
• An unauthorized student may participate in the exam.
• Multiple students may participate together for the exam.
• The student may use his still picture for face recognition.
• The student may use a device such as a mobile,
laptop, or iPad to run a video for face recognition.
• The student may use books during the exam.
Our main goal is to mitigate those challenges. In the online proctoring module, we use biometric methods like face detection and recognition with eye-blinking detection. The suggested system's algorithm is detailed in Algorithm 1. In the face-recognition part, we detect and recognize students' faces and detect multiple faces in front of the camera. There is a chance that students can use their still pictures in front of the web-camera. As a result, the face-recognition algorithm recognizes the student as a real face. To avoid the recognition of still pictures, we use eye-blinking methods. If the number of eye-blinking is not more than 30, then we can confirm that the picture in front of the camera is still. There is another possibility that students can hold a device in front of the web camera by playing his face video. In this case, face recognition and eye-blinking algorithms will detect the image as real and authenticate. So, we use object detection methods like YOLOv3, which also serve to prevent cheating in the exam using devices. We also detected the book using the same YOLOv3 model as the fare and secure exam.

Algorithm 1 Master Algorithm

```
procedure onlineProctor
    while True do
        frame ← captureFrameFromWebCamera
        faceDetect ← faceDetection(method = ”HOG”)
        faceCount ← detectFaceCountForCurrentFrame
        if faceCount == 0 then
            Exam cancel: no face found
        else if facecount == 1 then
            Exam continue
        else if facecount > 1 then
            Exam cancel: multiple face found
        end if
        faceMatch ← faceRecognition()
        if faceMatch == False then
            Exam cancel: Unauthorized face found
        end if
        blinkingRatio ← eyeBlinkingDetection()
        if blinkingRatio > 4.5 then
            noBlinking ← noBlinking + 1
        end if
        if noBlinking > 30 then
            Exam cancel: still image found
        end if
        object ← objectDetection()
        if object == list of target objects then
            Exam cancel: object found
        end if
    end while
end procedure
```

B. Datasets

We have used FDDB data sets for face detection which contain 5171 face annotation from 2845 images collected from Faces in the wild data sets. We divide the data sets into two parts, including faces and without faces, and implement face detection algorithms to evaluate our proposed system. The resolution of each image in the data set is 86 x 86 pixels. To evaluate face recognition algorithms, we used LFW dataset which contain 5749 people's 13233 images, where 1680 people's had two or more images. As our face recognition algorithm needs a single image for face recognition, we divided the data sets into ten sections based on the number of images available of the people in each data set. In the data set, every 3180 people have one image, every 775 and 290 people have two and three images consecutively, and so on. Each of the image resolutions is 250 x 250 pixels.

C. Face Detection

In our proposed system, we used the HOG (Histograms of Oriented Gradients) method to detect the faces that were proposed by Navneet and Dalal [29]. In the initial stage of face detection, we convert our input image into grayscale because we don't need an RGB image to find faces. After that, we process every single pixel and the directly surrounding pixels of the image at a moment. We would like to determine the darkness of the current pixel is in contrast to the pixels around it. To show in which direction the image is becoming darker, we draw an arrow. If we replicate such a method for each and every pixel in the image, then we discover that every pixel is followed by an arrow. Gradients are the arrows, which determine the overall image's movement from brightness to darkness. Following that, we can see the image's fundamental pattern. To conduct the function, we divided all of the images into 16x16 pixel squares. Then we count gradient points in each major direction of each square and replace the square image with the strongest single gradient direction. The process's output will convert the original picture into the face's fundamental structure, which seems to be the most similar to the HOG pattern derived from training images. We used the HOG frontal face detector using dlib and the OpenCV library for face detection.

D. Face Recognition

Face recognition is the most popular biometric solution for the online authentication system. OpenCV is a famous computer vision library that was started by Intel in 1999. OpenCV implements three face recognition algorithms, including Eigenface, Fisherface, and LBPH (Local Binary Patterns Histograms) face recognition. To detect faces, these algorithms employ the Haar cascade classifier technique, introduced by Paul and Michael [30].In our proposed methodology, we snap a picture of a student as input and use HOG techniques to recognize faces in the image. Then, for the identified picture, estimate the 68 landmarks. Faces that are oriented differently and seem differently to a computer may all belong to the same person, and these signs can be used to easily identify them. Finally, the identified photos are directly compared to previously learnt and saved faces in our database. The pseudo code for facial recognition is shown in the Algorithm 2. We match a known face from our database to unknown faces using a deep neural network. We train a classifier to determine which known student is the closest match based on measures from a new test image. The classifier's output would be the name of a student. The number of faces in the photograph is also counted.

Algorithm 2 Face Recognition

1: procedure Face Recognition (studentId, studentName)
2: while True do
3: Grab current frame from student's Webcam
4: frame ← currentFrame
5: faceLocations ← get all faces on frame
6: faceEncoding ← get all faceEncodings on frame
7: faceMatch ← compare studentFace with all faces
8: if faceMatch == True then
9: Face Recognized
10: else
11: Face Unrecognized
12: end if
13: end while
14: end procedure

1) Facial Landmark Estimation: To a computer, the split faces rotated in various orientations appear to be different. To address this issue, we apply the face landmark estimation which aids in the localization and representation of important facial features including the right and left eye, nose, jawline, mouth, and right and left eyebrow. The HELEN dataset is being utilized to find 194 landmarks on the face from a single image in a millisecond using this approach, which gives an ensemble of randomized regression trees. The method below can help determine whether two faces facing different directions and appearing differently from a computer's perspective are actually the same person. Based on the fundamental concept of 68 distinct places on an image, we will train the system to recognize any 68 specific landmarks from the target image. We can center the eyes and lips no matter how the faces are rotated after using
this method

2) Encode the Faces: The most basic concept in facial recognition is matching a recognized face to an unknown one. We identify a previously tagged face that appears to be frighteningly similar to an unknown face as belonging to the same individual. If there are thousands of students, it will take a long time to recognize everyone. As a result, we will need a technique for extracting a few basic measures from each face so that we can measure our unknown face and find the closest known face. We may, for example, measure the distance between the eyes and eyebrows, the length of the nose and mouth, and the size of each ear.

E. Eye Blinking Detection

To identify a still image, the eye blinking method is utilized. Each eye is represented by 6 (x, y)-coordinates, which begin in the upper left corner and work clockwise around the rest of the area.

Algorithm 3 Eye Blinking Detection

1: procedure Blinking Detection(studentId, studentName)
2: eyeModel ←load shape-predictor-68-face-landmarks
3: while True do
4: frame ← Grab current frame from Webcam
5: # get Blinking Ratio
6: function bRatio(EyePoint, landmark)
7: leftPoint ← left eye point
8: rightPoint ← right eye point
9: centerTop ← center top eye point
10: centerBottom ← center bottom eye point
11: horLineLenght ← horizontal line of eye point
12: verLineLenght ← vertical line of eye point
13: ratio ← horLineLenght / verLineLenght
14: return ratio
15: end function
16: faces ←dlibFrontalFaceDetector (Frame)
17: for face ← faces do
18: # facial landmark (lm)
19: lm ← eyeModel(grayFrame, faces)
20: # Left Eye Ratio
21: ler ← bRatio ([37,38,39,40, 41,42], lm)
22: # Right Eye Ratio
23: rer ← bRatio([43,44,45,46, 47,48], lm)
24: bliningRatio ← (ler + rer)/2
25: end for
26: end while
27: end procedure

V. YOLO Models for Online Examination Proctoring

YOLO is an algorithm that uses neural networks to provide real-time object detection. This algorithm is popular because of its speed and accuracy. It has been used in various applications to detect traffic signals, people, parking meters, and animals.

Object detection is a phenomenon in computer vision that involves the detection of various objects in digital images or videos. Some of the objects detected include people, cars, chairs, stones, buildings, and animals.

This phenomenon seeks to answer two basic questions:

- What is the object? This question seeks to identify the object in a specific image.
- Where is it? This question seeks to establish the exact location of the object within the image.

Object detection consists of various approaches such as fast R-CNN, Retina-Net, and Single-Shot MultiBox Detector (SSD). Although these approaches have solved the challenges of data limitation and modelling in object detection, they are not able to detect objects in a single algorithm run. YOLO algorithm has gained popularity because of its superior performance over the aforementioned object detection techniques.

YOLO is an abbreviation for the term 'You Only Look Once'. This is an algorithm that detects and recognizes various objects in a picture (in real-time). Object detection in YOLO is done as a regression problem and provides the class probabilities of the detected images.

YOLO algorithm employs convolutional neural networks (CNN) to detect objects in real-time. As the name suggests, the algorithm requires only a single forward propagation through a neural network to detect objects. This means that prediction in the entire image is done in a single algorithm run. The CNN is used to predict various class probabilities and bounding boxes simultaneously.

The YOLO algorithm consists of various variants. Some of the common ones include tiny YOLO and YOLOv3.

YOLO algorithm is important because of the following reasons:

- **Speed:** This algorithm improves the speed of detection because it can predict objects in real-time.
- **High accuracy:** YOLO is a predictive technique that provides accurate results with minimal background errors.
- **Learning capabilities:** The algorithm has excellent learning capabilities that enable it to learn the representations of objects and apply them in object detection.

YOLO algorithm works using the following three techniques:

- Residual blocks
- Bounding box regression
- Intersection Over Union (IOU)

Residual blocks

First, the image is divided into various grids. Each grid has a dimension of S x S. The following image shows how an input image is divided into grids.

Fig. Grids

In the image above, there are many grid cells of equal dimension. Every grid cell will detect objects that appear within them. For example, if an object center appears within a certain grid cell, then this cell will be responsible for detecting it.

Bounding box regression

A bounding box is an outline that highlights an object in an image. Every bounding box in the image consists of the following attributes:

- Width (bw)
- Height (bh)
- Class (for example, person, car, traffic light, etc.)- This is represented by the letter c.
- Bounding box center (bx,by)

The following image shows an example of a bounding box. The bounding box has been represented by a yellow outline.

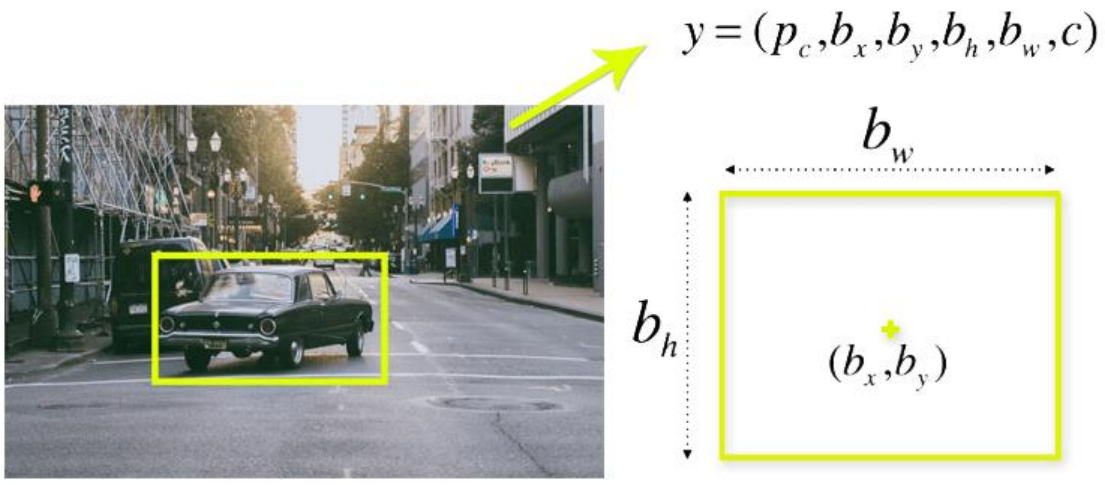


Fig. Bounding Box

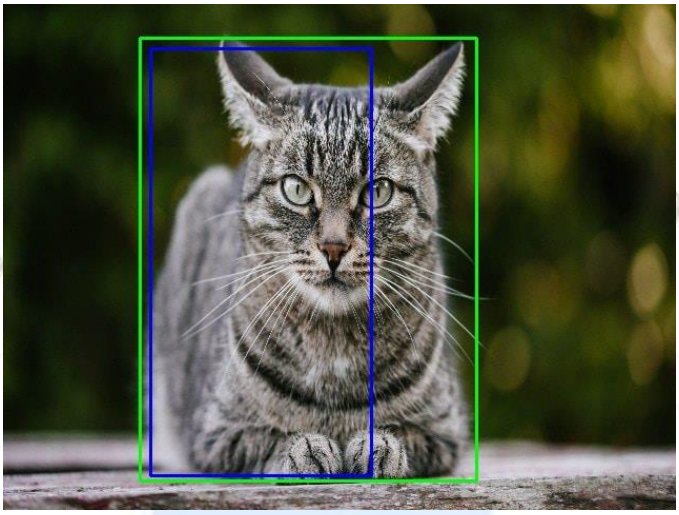
YOLO uses a single bounding box regression to predict the height, width, center, and class of objects. In the image above, represents the probability of an object appearing in the bounding box.

Intersection over union (IOU)

Intersection over union (IOU) is a phenomenon in object detection that describes how boxes overlap. YOLO uses IOU to provide an output box that surrounds the objects perfectly.

Each grid cell is responsible for predicting the bounding boxes and their confidence scores. The IOU is equal to 1 if the predicted bounding box is the same as the real box. This mechanism eliminates bounding boxes that are not equal to the real box.

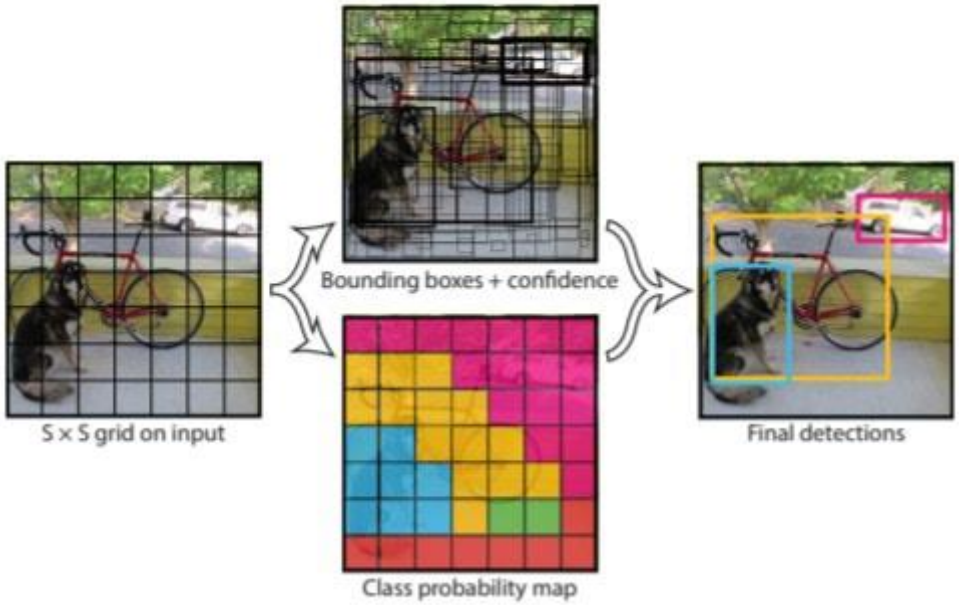
The following image provides a simple example of how IOU works.



In the image above, there are two bounding boxes, one in green and the other one in blue. The blue box is the predicted box while the green box is the real box. YOLO ensures that the two bounding boxes are equal.

Combination of the three techniques

The following image shows how the three techniques are applied to produce the final detection results.



First, the image is divided into grid cells. Each grid cell forecasts B bounding boxes and provides their confidence scores. The cells predict the class probabilities to establish the class of each object.

For example, we can notice at least three classes of objects: a car, a dog, and a bicycle. All the predictions are made simultaneously using a single convolutional neural network.

Intersection over union ensures that the predicted bounding boxes are equal to the real boxes of the objects. This phenomenon eliminates unnecessary bounding boxes that do not meet the characteristics of the objects

(like height and width). The final detection will consist of unique bounding boxes that fit the objects perfectly.

For example, the car is surrounded by the pink bounding box while the bicycle is surrounded by the yellow bounding box. The dog has been highlighted using the blue bounding box.

VI. RESULTS AND DISCUSSION

A. Building Trust through a Comprehensive Remote Proctoring System

Through the three rounds of design and revisions, we found the test-takers' cheating behaviors were reduced gradually. From the three rounds of interviews, we also found that the students indicated it was getting increasingly difficult to cheat in the tests. In the third interview, the participants admitted that they "couldn't cheat under this online proctoring environment on their own." The reliability of online proctoring also gained more and more acceptance among the participants. This is in line with González-González et al.'s 2020 research. Trust is the most important factor affecting schools, institutions, and test-takers when using online proctoring. It is of key value to ensure and maintain academic honesty and integrity in an online learning environment [22]. A well-designed remote proctoring system can effectively reduce cheating behaviors and maintain test equality [23], [24].

Previous research synthesized eight commonly-used online proctoring systems [13], [25], and identified various features that contributed to the improvement of the reliability of online proctoring, including live human proctors, students' interaction with proctors, monitoring students' screens, recording, automated proctoring, web camera, audio recording, etc.

In addition to those features, this study puts forward the following functions that may promote the reliability of remote proctoring: the necessity for the computer's front camera to capture the test-takers' face and hands, the need to place a second camera diagonally behind students (as illustrated in Fig. 2), the need to make explicit requirements regarding students' network connection and the resolution of their camera, the need to capture the testing environment via camera before taking the test, the requirement that students have their ears uncovered, the use of automated proctoring to assist human proctoring, and the preparation of backup tests in case of network interruptions.

B. Taking into Account Test-Takers' Mentality

Some research has pointed out the need to take into consideration the effect that new remote proctoring environments have on students: Some have pointed out that an unfamiliar environment may make students feel stressed and anxious [26], [27]. Hussein, Yusuf, Deb, Fong, and Naidu (2020) noted a majority of the students are accepting of the test proctoring method [13]; while they tend to get nervous after the test starts, the stress will soon diminish with the progression of the test.

This study also found most test-takers were stressed and anxious before and during the initial phase of the test, due to the fact that they were not familiar with online test proctoring. As a result, in order to reduce students' stress, this study explained the proctoring requirements and procedures to the students via videos and pictures, and recommended test-takers contact the proctors 20 minutes before the test began.

C. Recommendations Regarding Remote Test Proctoring

After three rounds of design, revision, testing, and soliciting users' feedback, this study produced a design that was much improved compared to the first round. According to the interviews, the third design can provide remote test proctoring more efficiently. Based on these findings, we suggest that proctors will benefit from doing the following:

Before testing: providing clear guidance regarding the testing procedures, prescribing rules concerning students' network and camera, recommending students access the testing system in advance, preparing backup tests, preparing Wi-Fi and mobile network.

Authentication before testing: authenticating students' identity via automated authentication systems, human proctors checking test-takers' ID, asking students to take videos of the testing environment, and having their ears not covered by hair. Live remote proctoring: using two cameras, one being the computer's front webcam, the other being put behind the students diagonally, asking students to turn on their

microphone to check if it works, asking students to share their screen, proctors videotaping the testing process, and automated proctoring system assisting the human proctors.

After testing: saving the recordings and data.

VII. CONCLUSION

Face recognition and object identification techniques are utilized in this study to give comprehensive knowledge for online tests. Our proposed method will aid in reducing inequity during the online exam. Human-induced detection is very important when conducting an online proctoring system, as it will aid in detecting students' suspicious behaviour throughout the test. We do not incorporate human activity detection in our suggested model, instead of relying on a single biometric solution and object recognition approaches for the online proctoring system. In the future, we hope to apply and investigate various human behaviours such as gazing out the window, conversing with people, focusing on other directions, moving about, and so on. We only utilize the YOLOv3 model because of its quicker object detection algorithms, although there are several other object detections approaches available. In the next study, we will focus on such approaches and compare them to our current suggested system. We have evaluated our proposed system using two datasets. However, the system has not been tested in a real life deployment with a large number of users. Future work will look into further testing and development of the system in real-life environments. The current proctoring systems are commercial and their designs and sources are not available openly. This work is an effort to develop open systems so the community can learn from each other leading to faster innovations in the field under open-source developments.

REFERENCES

[1] ASRA SARWATH, Dr. RAAFIYA GULMEHER. A Survey of Deep Learning Algorithms for Cyber Security Applications. March 2023.
[2] Tayeb Alipourfard , Hossein Arefi and Somayeh Mahmoudi.. A Novel Deep Learning Framework by Combination of Subspace-Based Feature Extraction and Convolutional Neural Networks for Hyperspectral Images Classification 2018
[3] D. L. Baggio. Enhanced human computer interface through webcam image processing library. Natural User Interface Group Summer of Code Application, pages 1–10, 2008.
[4] S. V. Bailey and S. V. Rice. A web search engine for sound effects. In Audio Eng. Society Convention 119. Audio Eng. Society, 2005.
[5] Asep Hadian S. G., Yoanes Bandung. A Design of Continuous User Verification for Online Exam Proctoring on M-Learning. 2019 International Conference on Electrical Engineering and Informatics (ICEEI) July 2019, 9 - 10, Bandung, Indonesia
[6] J. Chen and X. Liu. Transfer learning with one-class data. Pattern Recognition Letters, 37:32–40, 2014.
[7] G. Cluskey Jr, C. R. Ehlen, and M. H. Raiborn. Thwarting online exam cheating without proctor supervision. Journal of Academic and Business Ethics, 4:1–7, 2011.
[8] P. Guo, H. feng yu, and Q. Yao. The research and application of online examination and monitoring system. In IT in Medicine and Education,2008. IEEE Int. Sym. on, n, pages 497–502, 2008.
[9] Istiak Ahmad1, Fahad AlQurashi2, Ehab Abozinadah3, 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
[10] H. Hung and D. Gatica-Perez. Estimating cohesion in small groups using audio-visual nonverbal behavior. IEEE Trans. Multimedia, 12(6):563–575, 2010.
[11] Y.-G. Jiang, Q. Dai, T. Mei, Y. Rui, and S.-F. Chang. Super fast event recognition in internet videos. IEEE Trans. Multimedia, 17(8):1174–1186, 2015.
[12] A. Jourabloo and X. Liu. Pose-invariant 3d face alignment. In Proc. Int. Conf. Computer Vision (ICCV), pages 3694–3702, 2015.
[13] Yousef Atoum, Liping Chen, Alex X. Liu, Stephen D. H. Hsu, and Xiaoming Liu . Automated Online Exam Proctoring DOI 10.1109/TMM.2017.2656064, IEEE.
[14] MIKEL LABAYEN, RICARDO VEA , JULIÁN FLÓREZ, (Member, IEEE), NAIARA AGINAKO AND BASILIO SIERRA . Online Student Authentication and Proctoring System Based on Multimodal Biometrics Technology . May 21, 2021.

[15] D. L. King and C. J. Case. E-cheating: Incidence and trends among college students. Issues in Information Systems, 15(1), 2014.
[16] M. A. Sarrayrih and M. Ilyas, "Challenges of online exam, performances and problems for online university exam," International Journal of Computer Science Issues (IJCSI), vol. 10, no. 1, p. 439, 2013.
[17] A. Ullah, H. Xiao, M. Lilley, and T. Barker, "Privacy and usability of image and text based challenge questions authentication in online examination," in 2014 International Conference on Education Technologies
and Computers (ICETC). IEEE, 2014, pp. 24–29.
[18] G. Fenu, M. Marras, and L. Boratto, "A multi-biometric system for continuous student authentication in e-learning platforms," Pattern Recognition Letters, vol. 113, pp. 83–92, 2018.
[19] V. Selvi, R. Sankar, and R. Umarani, "The design and implementation of on-line examination using firewall security," IOSR Journal of Computer Engineering, vol. 16, no. 6, pp. 20–24, 2014.
[20] L. Wei, Z. Cong, and Y. Zhiwei, "Fingerprint based identity authentication for online examination system," in 2010 Second International Workshop on Education Technology and Computer Science, vol. 3. IEEE, 2010, pp. 307–310.
[21] K. Garg, K. Verma, K. Patidar, and N. Tejra, "Convolutional neural network based virtual exam controller," in 2020 4th International Conference on Intelligent Computing and Control Systems (ICICCS). IEEE, 2020, pp. 895–899.
[22] Y. Atoum, L. Chen, A. X. Liu, S. D. Hsu, and X. Liu, "Automated online exam proctoring," IEEE Transactions on Multimedia, vol. 19, no. 7, pp. 1609–1624, 2017.
[23] R. Bawarith, D. Abdullah, D. Anas, and P. Dr., "E-exam Cheating Detection System," International Journal of Advanced Computer Science and Applications, vol. 8, no. 4, pp. 176–181, 2017. [24] W. Rosen and M. Carr. An autonomous articulating desktop robot for proctoring remote online examinations. In Frontiers in Education Conf.,2013 IEEE, pages 1935–1939, 2013.
[25] Prathmesh Mohite, Rupam Patil, Vinaya Borhude, Aditya Pawar. Proctored Online Examination System Using Deep Learning and Computer Vision . https://doi.org/10.32628/IJSRST218282 . 23 April 2021
[26] Andrade Meira, Carneiro Praca, Alonso-Betanzos Bolón-Canedo, Marreiros, Performance evaluation of unsupervised techniques in cyber-attack anomaly detection, J Amb. Intell. Huma. Comput. (2019) 1–13.
[27] Thing, Network anomaly detection and attack classification: A deep learning approach, in: IEEE Wireless Communications and Networking Conference, 2017, pp. 1–6.
[28] Carro Lopez-Martin, Sanchez-Esguevillas, Lloret, Sensors 17 (9) (2017) 1967.
[29] Diro, Chilamkurti, Distributed attack detection scheme using deep learning approach for Internet of Things, Future Gener. Comput. Syst. 82 (2018) 761–768.
[30] Alice Macharia Njuguna . User Experience of Online Examinations and Proctoring: A Case Based Study.
July 2022 DOI: 10.47191/ijcsrr/V5-i7-12
[31] Bakhitzhan Kadyrov, Shirali Kadyrov(□), Alfira Makhmutova . Automated Reading Detection in an Online Exam . https://doi.org/10.3991/ijet.v17i22.33277 . 22 2022
[32] Florian Skopik, Giuseppe Settanni, Roman Fiedler, A problem shared is a problem halved: A survey on the dimensions of collective cyber defence through security information sharing, Comput. Secur. 60 (2016) 154–176
[33] H. Zhang, Y. Li, Z. Lv, A.K. Sangaiah, T. Huang, A real-time and ubiquitous network attack detection based on deep belief network and support vector machine, IEEE/CAA J. Autom. Sin. 7 (3) (2020b) 790–799.
[34] Rajendran Balakrishnan, Pelusi, Ponnusamy, Deep belief network enhanced intrusion detection system to prevent security breach in the internet of things, Internet Things (2019) 100112
[35] G. Thamilarasu, S. Chawla, Towards deep-learning-driven intrusion detection for the IoT, Sensors 19 (1977) (2019).
[36] M. Nabil, M. Ismail, M. Mahmoud, MostafaShahin, K. Qaraqe, E. Serpedin, Deep learning-based detection of electricity theft cyber-attacks in smart grid AMI networks, in: Deep Learning Applications for Cyber Security, 2019, pp. 73–102.
[37] Tanzila Saba , Amjad Rehman , Nor Shahida Jamai, SouadLarabi Marie-Sainte , Mudassar Raza , and Muhammad Sharif . Categorizing the Students' Activities for Automated Exam Proctoring using Proposed Deep L2-GraftNet CNN Network and ASO Based Feature Selection Approach . Digital Object Identifier 10.1109/ACCESS.2017.Doi Number 2017

[38] Z. Lipton, J. Berkowitz, C. Elkan, A critical review of recurrent neural networks for sequence learning, 2015, ArXiv preprint arXiv:1506.00019.
[39] H.Young, Poria, Cambria, Recent trends in deep learning based natural language processing, IEEE Comput. Intell. Mag. 13 (3) (2018) 55–75
[40] V. Mnih, AdriaPuigdomenechBadia, M. Mirza, A. Graves, T. Lillicrap, T.Harley, D. Silver, K. Kavukcuoglu, Asynchronous methods for deep reinforcement learning, in: International Conference on Machine Learning, 2016, pp. 1928–1937.
[41] Muhanad Abdul Elah Abbas , Saad Hameed . A Systematic Review of Deep Learning Based Online Exam Proctoring Systems for Abnormal Student Behaviour Detection. doi : https://doi.org/10.32628/IJSRSET229428 22 July 2022.
[42] T. Nguyen, Vijay JanapaReddi, Deep reinforcement learning for cyber security, 2019, ArXiv preprint ArXiv:1906.05799.
[43] A. Ferdowsi, U. Challita, WalidSaad, Narayan B. Mandalay, Robust deep reinforcement learning for security and safety in autonomous vehicle systems, in: IEEE International Conference on Intelligent Transportation Systems, ITSC, 2018, pp.307–312.
[44] Lantao Yu, Yi Wu, Rohit Singh, Lucas Joppa, Fei Fang, Deep reinforcement learning for green security game with online information, in: Workshops at the Thirty-Second AAAI Conference on Artificial Intelligence, 2018.
[45] Alec Radford, Luke Metz, SoumithChintala, Unsupervised representation learning with deep convolutional generative adversarial network, 2015, ArXiv preprint arXiv:1511.06434
[46] Z. Katzir, Y. Elovici, Quantifying the resilience of machine learning classifiers used for cyber security, Expert Syst. Appl. 92 (2018) 419–429.
[47] Zilong Lin, Yong Shi, ZhiXue, Idsgan: Generative adversarial networks for attack generation against intrusion detection, 2018, ArXiv preprint arXiv:1809.02077.
[48] S. Chhetri, A. B Lopez, J. Wan, Mohammad A. Al Faruque, GAN-Sec: Generative adversarial network modeling for the security analysis of cyber-physical production systems. IEEE: Automation and Test in Europe Conference and Exhibition, DATE, 2019, pp. 770–775.
[49] Chuanlong Yin, Yuefei Zhu, Shengli Liu, JinlongFei, He tong Zhang, Anenhancing framework for bonnet detection using generative adversarial networks, in: 2018 International Conference on Artificial Intelligence and Big Data, 2018, pp 228–234.
[50] Yavanoglu, O., Aydos, M.: A review on cyber security datasets for machine learning algorithms. In: 2017 IEEE International Conference on Big Data (Big Data), pp. 2186–2193 (2017)
[51] Fraley, J.B., Cannady, J.: The promise of machine learning in cybersecurity. SoutheastCon 2017, 1–6 (2017)
[52] Xie, M., Hu, J., Slay, J.: Evaluating host-based anomaly detection systems: application of the one-class SVM algorithm to ADFA-LD. In: 2014 11th International Conference on Fuzzy Systems and Knowledge Discovery (FSKD), pp. 978–982 (2014)
[53] Chowdhury, S., et al.: Botnet detection using graph-based feature clustering. J. Big Data 4 (1), 14 (2017)
[54] Neethu, B.: Adaptive intrusion detection using machine learning. Int. J. Comput. Sci. Netw. Secur. 13(3), 118 (2013)
[55] Kozik, R., Choraś, M., Renk, R., Hołubowicz, W.: A proposal of algorithm for web applications cyber attack detection. In: IFIP International Conference on Computer Information Systems and Industrial Management, pp. 680–687 (2015)
[56] M. Tavallaee, E. Bagheri,W. Lu, and A. A. Ghorbani, ``A detailed analysis of the KDD CUP 99 data set," in Proc. IEEE Int. Conf. Comput. Intell. Secur. Defense Appl., Jul. 2009, pp. 1_6.
[57] G. Meena and R. R. Choudhary, ``A review paper on IDS classification using KDD 99 and NSL KDD dataset in WEKA," in Proc. Int. Conf. Comput., Commun. Electron., 2017, pp. 553_558.
[58] Javaid, A., Niyaz, Q., Sun, W., Alam, M.: A deep learning approach for network intrusion detection system. In: Proceedings of the 9th EAI International Conference on Bio-inspired Information and Communications Technologies, pp. 21–26 (2016)
[59] Bhamare, D., Salman, T., Samaka, M., Erbad, A., Jain, R.: Feasibility of supervised machine learning for cloud security. In: 2016 International Conference on Information Science and Security (ICISS), pp. 1–5 (2016)
[60] Lippmann, R., Haines, J.W., Fried, D.J., Korba, J., Das, K.: The 1999 DARPA off-line intrusion detection evaluation. Comput. Netw. 34(4), 579–595 (2000)

[61] Saad, S., et al.: Detecting P2P botnets through network behavior analysis and machine learning. In: 2011 Ninth Annual International Conference on Privacy, Security and Trust, pp. 174–180 (2011)
[62] B. Alom, Taha, Intrusion detection using deep belief networks, , NAECON, 2015, pp. 339-344.
[63] Ugo Fiore, Francesco Palmieri, Network anomaly detection with the restricted Boltzmann machine, Neurocomputing 122 (3) (2014) 13–23.

[64] J. Yang, J. Deng, S. Li, Hao, Improved traffic detection with support vector machine based on restricted Boltzmann machine, Soft Comput. 21 (11) (2017a) 3101–3112.
[65] Kato, K., Klyuev, V.: An intelligent DDoS attack detection system using packet analysis and support vector machine. In: IJICR, pp. 478–485 (2014)
[66] Yusof, A.R., Udzir, N.I., Selamat, A.: An evaluation on KNN-SVM algorithm for detection and prediction of DDoS attack. In: International Conference on Industrial, Engineering and Other Applications of Applied Intelligent Systems, pp. 95–102 (2016)
[67] Saad, S., et al.: Detecting P2P botnets through network behavior analysis and machine learning. In: 2011 Ninth Annual International Conference on Privacy, Security and Trust, pp. 174–180 (2011)
[68] Hoque, N., Bhattacharyya, D.K., Kalita, J.K.: A novel measure for low-rate and high-rate DDoS attack detection using multivariate data analysis. In: 2016 8th International Conference on Communication Systems and Networks, pp. 1–2 (2016)
[69] T. Giménez, C., P. Villegas, A., A. Maranón, G.: An anomaly-based approach for intrusion detection in web traffic (2010)
[70] Lippmann, R., Haines, J.W., Fried, D.J., Korba, J., Das, K.: The 1999 DARPA off-line intrusion detection evaluation. Comput. Netw. 34(4), 579–595 (2000)
[71] T. Gimenez, C., P. Villegas, A., Alvarez, G.: A self-learning anomaly-based web application firewall. In: Computational Intelligence in Security for Information Systems, pp. 85–92. Springer (2009)
[72] M. Hatada, M. Akiyama, T. Matsuki, T. Kasama, Empowering anti-malware research in Japan by sharing the MWS datasets, J. Inf. Process. 23 (5) (2015) 579–588.
[73] J. Song, H. Takakura, Y. Okabe, M. Eto, D. Inoue, K. Nakao, Statistical analysis of honeypot data and building of Kyoto 2006+ dataset for NIDS evaluation, in: Proceedings of the First Workshop on Building Analysis Datasets and Gathering Experience Returns for Security, 2011, pp. 29–36.
[74] N. Moustafa, J. Slay, UNSW-NB15: a comprehensive data set for network intrusion detection systems (UNSW-NB15 network data set), in: 2015 Military Communications and Information Systems Conference, MilCIS, IEEE, 2015, pp. 1–6.
[75] R. Panigrahi, S. Borah, A detailed analysis of CICIDS2017 dataset for designing intrusion detection systems, Int. J. Eng. Technol. 7 (3) (2018) 479–482, 24.
[76] M. Xie, J. Hu, X. Yu, and E. Chang, ``Evaluating host-based anomaly detection systems: Application of the frequency-based algorithms to ADFA-LD," in Proc. Int. Conf. Netw. Syst. Secur., 2014, pp. 542_549
[77]. Brugger S, Chow J (2005) An assessment of the DARPA IDS evaluation dataset using snort. Tech. Rep. CSE-2007-1, Department of Computer Science, University of California, Davis (UCDAVIS)
[78] R. P. Lippmann et al., ``Evaluating intrusion detection systems: The 1998 DARPA off-line intrusion detection evaluation," in Proc. DARPA Inf. Survivability Conf. Expo. (DISCEX), vol. 2, 2000, pp. 12_26.
[79] Y. LeCun, Y. Bengio, and G. Hinton, ``Deep learning," Nature, vol. 521, pp. 436_444, May 2015.
[80] G. E. Hinton, ``Deep belief networks," Scholarpedia, vol. 4, no. 5, p. 5947, 2009.
[81] Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner, ``Gradient-based learning applied to document recognition," Proc. IEEE, vol. 86, no. 11, pp. 2278_2324, Nov. 1998.
[82] L. Deng and D. Yu, ``Deep learning: Methods and applications," Found. Trends Signal Process., vol. 7, nos. 3_4, pp. 197_387, Jun. 2014.
[83] Harshal A. Kute dan D. N. Rewadkar., "A Survey on Continuous User Identity Verification Using Biometrics Traits for Secure Internet Services", International Journal of Science and Research (IJSR), 2012.
[84] C. Shen, H. Zhang, Z. Yang and X. Guan, "Modeling Multimodal Biometric Modalities for Continuous User Authentication", IEEE International Conference on Systems, Man, and Cybernetics, 2016.