Online-Journals.org

iJOE International Journal of Online and Biomedical Engineering

iJOE | eISSN: 2626-8493 | Vol. 19 No. 10 (2023) | OPEN ACCESS

https://doi.org/10.3991/ijoe.v19i10.39583

PAPER

Abnormal Behavior Detection in Online Exams Using Deep Learning and Data Augmentation Techniques

Muhanad Abdul Elah Alkhalisy[1](✉), Saad Hameed Abid[2]

[1]Informatics Institute for Postgraduate Studies, Iraqi Commission for Computer and Informatics, Baghdad, Iraq

[2]Computer Engineering Department, Al-Mansur University, Baghdad, Iraq

phd202020557@iips.icci.edu.iq

ABSTRACT

Massive open online courses (MOOCs) and other forms of distance learning have gained popularity in recent years. The success of remote online exam proctoring determines the integrity of the exam. Deep-learning-powered proctoring services have also grown in popularity. A large number of samples are needed for deep-learning training. The network's generalization ability is poor due to insufficient training data or an uneven lack of variation. This study illustrates how to analyze students' anomalous behavior by utilizing a YOLOv5 deep model trained using newly produced dataset. To overcome insufficient training data for deep-learning-related issues, this paper proposes a data-augmentation method based on semantic segmentation. The MobileNetV3 model was used to get an image semantic segmentation mask, which was used to get a binary mask, which in turn was used to replace the image background by using conditional subtraction with randomly selected background images. Finally, randomly pixel-based color augmentation was added to the resulting image. The behavioral detection model used in this study achieved 0.98 mean average precision (mAP) on the produced dataset, showing acceptable detection precision. The experimental findings indicate that the suggested augmentation method improves behavioral detection precision by more than 0.3%.

KEYWORDS

online exam, semantic segmentation, data augmentation, behavior analysis, anomaly detection

1 INTRODUCTION

Online exams and tests are becoming more and more common for course instructors to evaluate students' knowledge due to the rapid development of online learning over the past ten years [1]. Due to the COVID-19 lockdown in 2019, this trend was further significantly accelerated during natural disasters, war, and pandemics [2]. Most schools and universities have embraced online teaching and exams [3], [4]. Massive open online courses (MOOCs) are becoming increasingly popular [5]. Before students can receive a final course certificate, MOOCs like Coursera and EdX frequently require them to pass online exams [6], [7]. The subject of course evaluation has received the

Muhanad Alkhalisy, A.E., Abid, S.H. (2023). Abnormal Behavior Detection in Online Exams Using Deep Learning and Data Augmentation Techniques. *International Journal of Online and Biomedical Engineering (iJOE)*, 19(10), pp. 33–48. https://doi.org/10.3991/ijoe.v19i10.39583

Article submitted 2023-03-14. Resubmitted 2023-05-12. Final acceptance 2023-05-12. Final version published as submitted by the authors.

most attention in online education studies [8], [9]. The lack of direct student-teacher interaction is problematic for online course evaluation [10] Online exams require proctoring, like those taken in lecture halls and colleges [11]. Online tests like MOOCs and recruitment exams may require proctoring services powered by artificial intelligence [12]. On online exams, cheating is much simpler to commit [12].

Consequently, it is essential to use an AI-based system to monitor each student [13]. Online tests' integrity must identify unusual behavior to prevent cheating [3]. The development of deep learning has aided computer vision, and its methods can be applied to carry out routine computer-vision tasks [14]. Object-identification algorithms have succeeded in various fields, including spotting unusual exam behavior [15]. The exam could be stopped, or a report could be submitted for staff review as a result of systematic anti-cheating measures. Human proctors may use monitoring software to keep an eye on the students. A human proctor is contacted when cheating is detected, and the student's highly questionable actions are recorded [16].

The lack of sufficient training data or an unbalanced class distribution in the datasets is the most frequently cited issue in machine learning [17]. Data augmentation is one method of addressing this issue [17]. Recently, a relatively new data augmentation technique known as mixing and deleting various image regions and others have grown in popularity [18].

Finding a solution in abnormal behavioral detection research is extremely difficult due to the various complex cases encountered. Our methodology's primary objective is to automatically classify anomalous captured webcam frames obtained from online examination room using a YOLOv5 deep learning model. Deep learning consumes many computing resources and needs many training samples. The most common machine-learning problem is insufficient training data or an unequal lack of variety within the datasets. In this paper, we concentrated on the issue of insufficient training data. The primary contributions we made to this study are listed below:

- First, we created a real video dataset with cheating and non-cheating videos. The videos are of various lengths and show various forms of cheating.
- Second, we created the ground-truth dataset by manually labeling each selected object in a video frame with a specific behavioral attribute, such as a student using a phone, glancing to one side, moving their hands, or moving their eyes.
- Third, we proposed a data-augmentation approach based on deep-learning techniques, image segmentation, image manipulation, and colour pixel-based data augmentation. This method was added as a new offline augmentation technique during the data pre-processing.

The structure of the article is as follows. An overview of a few pertinent related works is provided in Section 2. The workflow and general system overview are described in Section 3. A scientific-technical description of the core modules and suggested methods can be found in Section 4. Results from system experiments are presented in Section 5. Section 6 concludes and offers suggestions for future work.

2 RELATED WORK

As more and more institutions move to the digital world, exams must be fair, and students must demonstrate that they understand the material; this is made possible by impartial ongoing evaluations.

Saba developed a system that automatically detects exam activity, tracks students' body movements, and applies deep learning to categorize their actions into

six groups. The behaviors include acting normally, gazing in all directions, making gestures, and looking to the left or right. Four different augmentations methods are used, mirroring, Gaussian noise, salt-and-pepper noise, and color shifting [19].

Ramzan focuses on spotting and identifying unusual behaviors in academic settings such as exam rooms, which may aid invigilators in keeping an eye on the students and discouraging cheating or unethical tactics. They created a dataset for out-of-the-ordinary behaviors during the examination and suggested a deep-learning model to detect them. This work's data-augmentation process included rotation range, width shift, flipping the data horizontally or vertically, and rescaling [20].

A method for identifying classroom behavior that uses an enhanced object-detection model is proposed by Tang. The feature pyramid structure FPN and PAN is merged with a weighted bidirectional feature pyramid network, Bi FPN, in the neck network of the original YOLOv5 model. Then, they are analyzed utilizing feature fusion of different object sizes to mine the "fine-grained" characteristics of varied behaviors. Data enhancement was done on several images of the student while standing. The cropped image was given a horizontal flip, brightness enhancement, and Gaussian noise as data augmentation [21].

Kadyrov proposed a system for automatically detecting test-taker reading. They acquire a dataset of brief video clips that mimic an online examination setting. They use various video augmentation techniques, such as cropping and boosting brightness, to expand the training dataset. For training, two distinct deep-learning methodologies are used [22].

Genemo recognized suspicious student behavior during the exam by surveilling the exam rooms. L4-BranchedActionNet is the suggested name for a 63-layer-deep CNN model. The center of the proposed CNN structure is the modification of the VGG-16 with four additional branches. The developed framework is initially trained on the CUI-EXAM dataset using the SoftMax function, resulting in a pre-trained framework. Following feature extraction, the dataset for identifying suspicious activity is sent to this pre-trained algorithm. The obtained deep features are then subjected to feature-subset optimization. By flipping the images, image augmentation is used to expand the dataset [23].

However, there has not been research on the use of data-augmentation-based deep-learning image-semantic segmentation combined with image manipulations and conventional color-based augmentations, because of an insufficient amount and variety of training data. A newly created dataset with ground-truth data was used in our research to train a model for analyzing and spotting unusual student behavior, which will help ensure fair exam administration.

3 TOOLS AND METHODS

In this section, the models used in this work, such as the YOLOv5 detection model and other model used for an image-semantic segmentation process will be described in more detail.

3.1 YOLOv5 network architecture

The R-CNN and YOLO algorithm series are the foundation of most existing high-performance object-detection frameworks [24]. While R-CNN-based object-detection frameworks have shown impressive accuracy in various applications, their detection durations are sometimes too long for several uses. These frameworks are not always capable of real-time object detection. A YOLO method series addresses the time

crunch by recasting the picture-identification issue as a regression problem amenable to a simple cascade algorithm [25]. The YOLO model was introduced by Joseph et al. [26], and it is a one-step detection approach. Its cutting-edge, real-time, object-detection model garnered much interest in the academic world thanks to its groundbreaking results and state-of-the-art capabilities. Accurate predictions of the probability and coordinates of the bounding boxes enhance detection performance for multi-label targets.

YOLOv5 is the most recent and state-of-the-art iteration of the You Look Only One (YOLO) algorithm family. With its high performance and quick detection speed, the YOLOv5 model is up to the challenge of real-time applications. It can quickly perform all the necessary stages for object detection with only one neural network. The YOLOv5 may be broken down into three primary components: the backbone, the neck, and the head [27]. While the backbone extracts feature data from incoming photos, the neck builds feature maps at three scales. In order to find objects, the prediction head fuses extracted features to obtain richer target characteristics [27]. The non-maximum-suppression strategy is used in this model to eliminate from the target the prediction frames with high overlap and poor scores [26]. The prominent architecture of YOLOv5 is illustrated in Figure 1.

The model features a YOLO-detecting head, PANet neck, and SPP backbone, which YOLOv5 adjusts, The final detection data is obtained by using a non-maximum suppression (NMS) technique [28] and setting appropriate thresholds to eliminate any unnecessary data from the array. The conversion of the input image to bounding boxes (BBoxes) is known as the inference process. In contrast to the previous edition of the YOLO algorithm, bounding-box regression details in the YOLOv5 process can be explained by using Eq. (1).

$$\begin{aligned} g_x &= 2\sigma(s_x) - 0.5 + r_x \\ g_y &= 2\sigma(s_y) - 0.5 + r_y \\ g_h &= p_h(2\sigma(s_h))^2 \\ g_w &= p_w(2\sigma(s_w))^2 \end{aligned} \tag{1}$$

The formula above sets the feature map's upper left corner coordinate value to (0, 0). The predicted center point's unadjusted coordinates are (r_x) and (r_y). The information of the adjusted prediction box is represented by (g_x, g_y, g_w) and (g_h). (p_w) and (p_h) are previous anchor information; (s_x) and (s_y) refer to the prediction box offset obtained from the model [26].

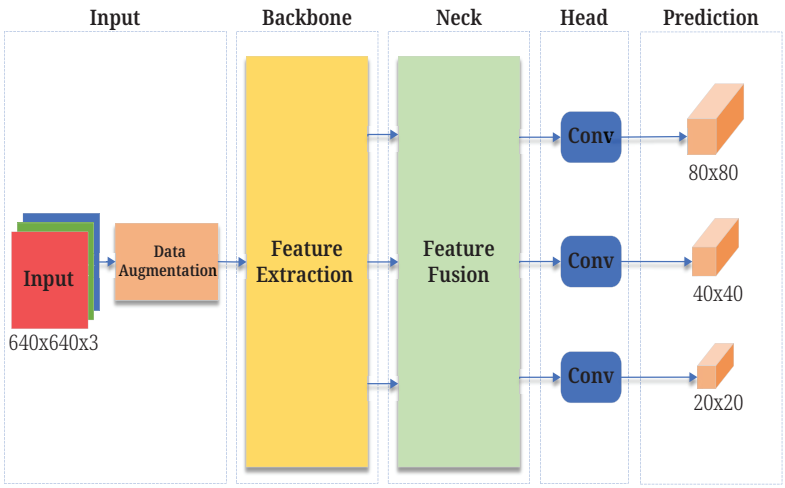

Fig. 1. YOlOv5's prominent architecture

3.2 Semantic-segmentation model

Semantic segmentation is a kind of deep-learning method that assigns a label or category to every one of an image's pixels. It can identify sets of pixels that can be separated into meaningful groups. Unlike object detection, which focuses on identifying the location of objects in an image, semantic segmentation produces a dense pixel-wise output, which means that every pixel in an image is classified as belonging to a particular semantic class. One of the popular semantic segmentation is MediaPipe's selfie, which it based on the MobileNetV3 model [29].

MediaPipe's selfie segmentation is a feature of that open-source framework's ability to facilitate the creation of portable pipelines for processing multimedia content, regardless of the platform used. This method uses machine learning to separate the subject from a selfie backdrop. The outcome is a binary mask distinguishing between the picture's backdrop and the foreground (person, face, body) [30].

The selfie segmentation method generates highly accurate and reliable results using deep neural networks trained on massive datasets. It is adaptable to different lighting scenarios, backdrop changes, and occlusions, making it useful in many practical settings [31].

The MediaPipe selfie segmentation system includes both a general and a landscape model. Both models are built on top of MobileNetV3, a deep model. However, they have been tweaked to improve their performance in semantic segmentation by introducing a low-latency segmentation decoder known as lite-reduced atrous spatial pyramid pooling (LR-SPP) [32]. There are three distinct pathways in this novel decoder: one for low-resolution semantic characteristics, one for high-resolution details, and one for lightweight attention. They have over 35% latency reduction on high-resolution cityscapes.

A dataset is created when LR-SPP is used with MobileNetV3. The standard model takes an input $256 \times 256 \times 3$ (HWC) tensor and produces an output $256 \times 256 \times 1$ (segmentation mask) tensor [30]. Like the general model, the landscape model performs on a $144 \times 256 \times 3$ (HWC) tensor. It uses fewer FLOPs than the average model, allowing quicker processing times. The generic model takes advantage of ML Kit's selfie segmentation API, which accepts an input picture and returns a mask [33].

4 PROPOSED METHODOLOGY

This study uses deep-learning models for real-time cheating detection using recorded video frames. This study aims to build an effective and efficient solution for online test systems based on the YOLOv5 model, using new data augmentation to handle data-imbalance problems. The critical components of the suggested method are depicted in Figure 2.

The following subjects are covered in more detail in the following paragraphs: dataset development, data labeling, dataset augmentation, training of behavioral detection models, performance evaluation, results, and comparative analysis.

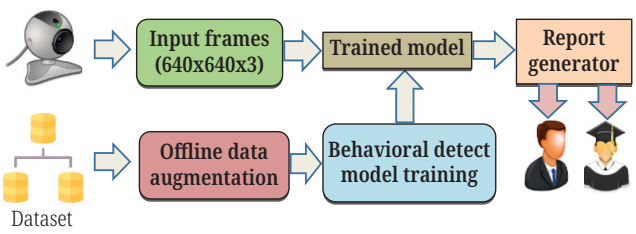

Fig. 2. The components of the suggested method

4.1 Dataset development

Due to the lack of a publicly accessible dataset of videos taken during actual online tests [21], an online exam student-behavior dataset was manually created for this paper. Twenty-four videos with a resolution of 1280 × 720 and a frame rate of 15 frames per second were acquired using a webcam as part of the data collection. In order to collect our data, we used a web application we had developed that served as an online exam simulator and had video-capture capabilities [34]. With hundreds of multiple-choice questions chosen from a question pool, the application evaluated participants' knowledge of students.

The participants were required to engage in some deliberate fraud scenarios. The use of a mobile phone and the movement of the hands, eyes, or head to the left or right are examples of these actions. All participants joined in the experiment from various locations, with various cameras and lighting setups. These variations made it challenging to catch cheating incidents. The videos mimicked a student taking an online test in front of a webcam. The total running time of all videos was approximately 6 hours, with each video lasting roughly 15 minutes per scenario. The six participant students were seated in front of computers.

Images for the dataset were extracted from recorded videos at predetermined frame intervals. After filtering, there were 8,520 images total in the dataset.

Our dataset can be downloaded from: https://doi.org/10.7910/DVN/WUWRAB.

4.2 Data labeling

Another difficult task was labeling each frame with a class subject. We created the ground truth dataset by manually annotating each frame with labelImg and makesense labeling software to assign each activity to a particular behavior.

The dataset contains four classes: mobile using, hand moving, eye moving, and looking aside. Each image has a corresponding text file containing annotation information. Each text file contains the label category (class ID) and coordinates information for the object localization in the frame. Details about the created dataset are provided in Table 1.

Table 1. Developed dataset details

Index	Class	Number of Images	Description
0	Mobile_Using	1720	Students using mobile phone
1	Hand_Move	1700	Student moving his hand
2	Eye_Move	1700	Student moving his eye left or right
3	Looking Side	1400	Student looking left or right
Total		**8520**	

4.3 Data augmentation

Deep learning uses many samples for training and requires a lot of computing power. Diversity in dataset samples enables the model to train better and achieve more

precise detection. Insufficient training data, the most commonly mentioned issue in machine learning, is the main focus of this paper. To solve this problem, we suggested a brand-new image-processing and deep-learning-based data-augmentation technique; this method was added as a new offline augmentation method in the data-preprocessing stage to expand our developed dataset size and variation.

We utilized three techniques: image-segmentation-based deep networks for image background removal, image masking to add new image background to the segmented foreground image, and color augmentation applied to the resulting image. Figure 3 illustrates the essential parts of the newly proposed augmentation methods.

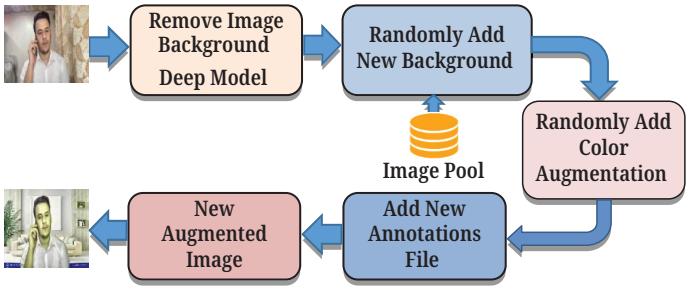

Fig. 3. The essential parts of the new proposed augmentation methods

As illustrated in Figure 5, the first stage in our proposal augmentation methods was to remove the image background from the selected image using the MediaPipe selfie image semantic segmentation, which it based on the MobileNetV3 model. We used the ML Kit's selfie segmentation API, which takes an input image and produces an output mask. The mask defaults to the same size as the input image. A float number between 0.0 and 1.0 is assigned to each mask pixel. The probability that a pixel represents a person increases as the number gets closer to 1.0, and vice versa. The resulting output is a probability-map image (probability mask). By applying a global thresholding technique to the probability mask, we got a binary image map (binary mask). The binary map was calculated by using Eq. (2):

$$bi(x, y) = \begin{cases} 1, \text{ if } pm(x, y) > T \\ 0, \text{ if } pm(x, y) \leq T \end{cases} \quad (2)$$

where $bi(x, y)$ is the binary map pixel value, $pm(x, y)$ is the probability map pixel value, and (T) represents the threshold value of 0.1 in our work. After that, we used the result in the binary mask to get the segmented image by utilizing conditional mapping between the original image and the binary mask using Eq. (3):

$$g(x, y) = \begin{cases} g(x, y), \text{ if } bi(x, y) = 1 \\ 255, \text{ if } bi(x, y) = 0 \end{cases} \quad (3)$$

where $g(x, y)$ is the original image pixel value and $bi(x, y)$ is the binary map pixel value.

Based on the above rule, if a given value in the mask array was equal to 1, then the original image array's corresponding value was left unchanged; otherwise, we changed the value to 255. Figure 4 shows the output of the segmentation process.

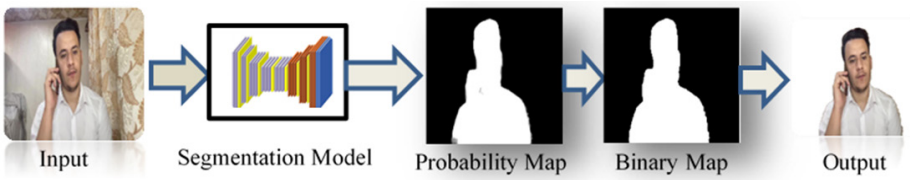

Fig. 4. The output of the segmentation process

The second stage was to randomly select an image from a pool of images and add it as background to the resulting segmented image; this is done by utilizing conditional mapping between the binary mask image, segmented image, and the background image by using Eq. (4):

$$g(x,y)=\begin{cases} g(x,y), \text{ if } bi(x,y)=1 \\ bg(x,y), \text{ if } bi(x,y)=0 \end{cases} \tag{4}$$

where $g(x, y)$ is the original image pixel value, $bi(x, y)$ is the binary map pixel value, and $bg(x, y)$ is the background image pixel value. Based on the above rule, if a given value in the mask array equals 1, the original image's corresponding value is left unchanged; otherwise, we replace the value in the original images with the corresponding value in the background images. Figure 5 shows the adding-background process.

Fig. 5. Adding background process

The last step is to add one randomly selected color augmentation to the resulting image. Contrast-limited adaptive histogram equalization (CLAHE), Gaussian noise, random gamma, optical distortion, hue saturation, and random brightness contrast are among the applied color augmentation methods. Figure 6 shows the adding-color augmentation process.

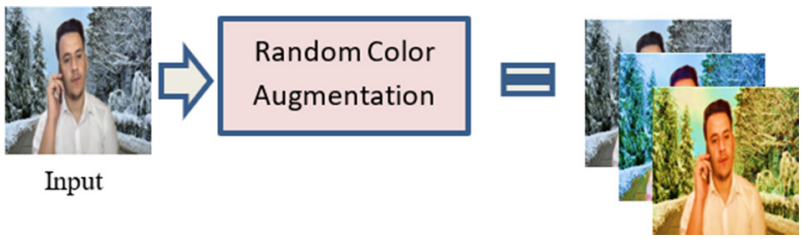

Fig. 6. Adding-color augmentation process

Finally, we copy the bounding-box coordinate information from the source-image annotation file and create another file for the newly resulting augmented image with the same annotation information. Algorithm 1 summarizes this phase.

Algorithm (1) Data Augmentation

```
Input: Image: Img
Probability mask: Pm
Binary mask: Bm
Threshold: T
Background image: bgImg
Output: Augmented Image: AugImg
Begin
Step1: Load image ← Img
Step2: Get image probability mask by MobileNetV3 ← Pm
Step3: Set T = 0.1 // global threshold value
Step4: Get binary mask from probability mask by using the
       Eq. (1) ← Bm
Step5: Get the segmented image by using Eq. (2)
Step6: Randomly select the background image ← bgImg
Step7: Add the background image to the segmented foreground
       image by using Eq. (3)
Step8: Add random color augmentation to resulting image
Step9: Create a new annotation file for the augmented image
Step10: Return → AugImg
End.
```

By fusing the look of other pictures with the content of a base image, the suggested approach enables the creation of new images with high perceptual quality. Figure 7 shows some results of the proposed augmentation methods.

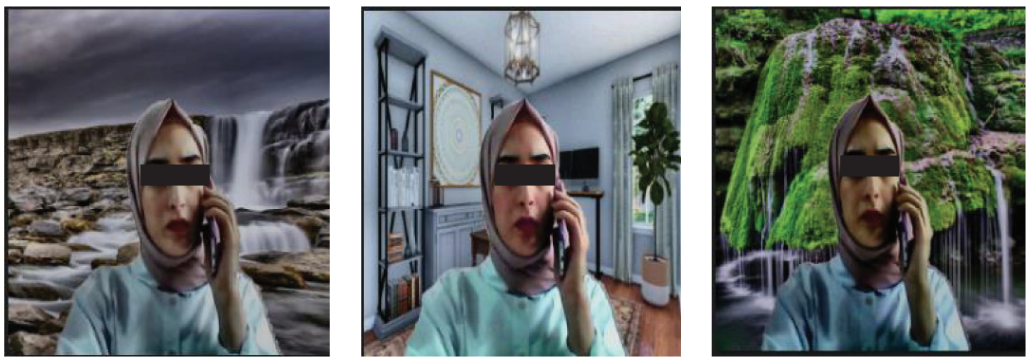

Fig. 7. Some results of the proposed augmentation methods.

The specified neural network may be trained using freshly produced pictures to increase the effectiveness of the training process. Table 2 provides details regarding the augmented dataset.

Table 2. Augmented dataset details

Class	Original	Augmented
Mobile_Using	1720	3300
Hand_Move	1700	3300
Eye_Move	1700	3300
Looking_Side	1400	3000
Total	**8520**	**15600**

5 EXPERIMENTAL SETTINGS AND MODEL TRAINING

Our methodology was put into practice using image segmentation based on deep-learning models, image manipulation, color argumentation, and YOLO models.

5.1 Training of behavioral-detection model

Before beginning the training process, the dataset had to be rigorously categorized and divided into training and validation components. We divided the dataset as Test 10%, Valid 20%, and Train 70%. We utilized our proposed data augmentation method in our private dataset before the training process began to prevent insufficient training data or uneven variation within the data. The results of the behavioral-detection-model training are shown in Figure 8.

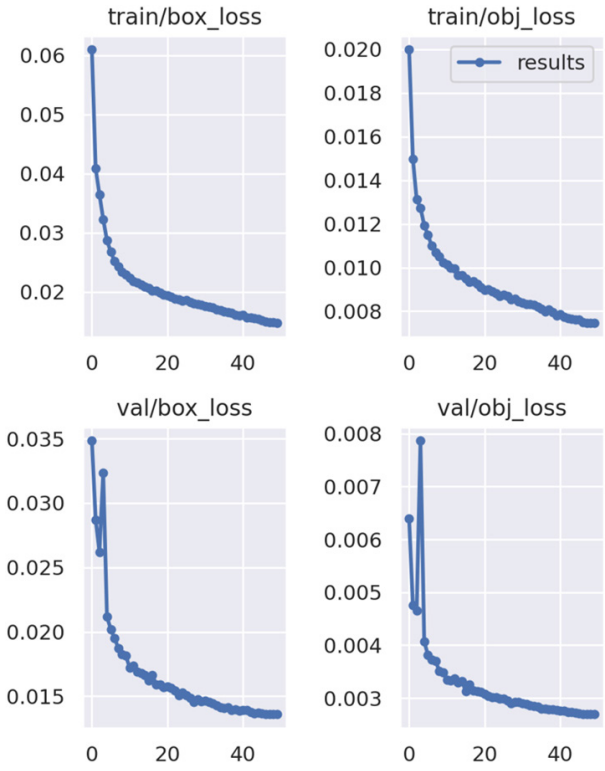

Fig. 8. Behavioral-detection-model training results

Careful hyperparameter tuning was required to produce a successful deep-learning model. The initial learning rate, cls anchor-multiple thresholds, SGD momentum/Adam, batch size, and epochs are examples of common hyperparameters in the YOLOv5 model. We began training with 2750 picture patches of entities from four different classes, starting with a learning rate of 0.01, anchor-multiple thresholds of 5.0, SGD momentum of 0.936, batch size of 16, and epochs of 50. The weights of the model were enhanced using the Adam optimizer. As a hardware

accelerator, we trained our model in Google Colab using the GPU. With good results for accuracy = 0.96, mAp@5 = 0.995, and mAp@5:095 = 0.838, the model converged during training.

5.2 Performance evaluation

Experiments were carried out to investigate models and system effectiveness. We used confusion matrices to evaluate the performance of a classification algorithm, with the terms used in the analysis defined as follows:

True Positive (TP): Positive samples with precise labels.

True Negative (TN): Number of samples with proper negative labels.

False Positive (FP): Incorrectly classifying negative samples as positive.

False Negative (FN): Incorrectly labeled positive samples.

Accuracy: The proportion of classes that were correctly predicted [as represented by Eq. (5)]:

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \times 100 \tag{5}$$

Precision: The proportion of correctly anticipated and positive classifications [as represented by Eq. (6)]:

$$Precision = \frac{TP}{TP + FP} \times 100 \tag{6}$$

Recall: All positive courses accurately predicting the proportion of classes [as represented by Eq. (7)]:

$$Recall = \frac{TP}{TP + FN} \times 100 \tag{7}$$

Models for object detection, such as YOLO, were evaluated using the mean average precision (mAP). The mAP calculates a score by comparing the detected box to the baseline bounding box. The model's detection became more accurate as the score increased.

Average Precision (AP): When calculating the weighted average of the precisions at each threshold, the increase in recall from the prior threshold was taken into account [as represented by Eq. (8)]:

$$AP = \sum_{k=0}^{k=n-1} [\text{Recall}(k) - \text{Recall}(k+1)]^{*}\text{Precisions}(k) \tag{8}$$

Mean Average Precision (mAP): The mean average precision (MAP) measured the average AP for each class [as represented by Eq. (9)]:

$$\text{mAP} = 1/n \sum_{k=1}^{k=n} APk \tag{9}$$

5.3 Results and comparative analysis

The YOLOv5 object identification model, which was trained on a private dataset, served as the foundation for the behavioral detector in our suggested solution. YOLOv5 was discovered to have a high mean average accuracy (mAP) and a considerably quicker inference time throughout the assessment. Table 3 displays the outcomes of 250 photos tested on the trained models before applying the proposed argumentation to the built dataset.

Table 3. Model's outcomes before applying the argumentation method

Class	Precision (%)	Recall (%)	mAP (%)
Mobile_Use	0.64	0.67	0.67
Hand_Move	0.74	0.72	0.72
Eye_Move	0.71	0.72	0.73
Looking_Side	0.62	0.60	0.61

Table 4 displays the test result after applying the proposed argumentation method to the built dataset.

Table 4. Model's outcomes after applying the argumentation method

Class	Precision (%)	Recall (%)	mAP (%)
Mobile_Use	0.95	0.97	0.97
Hand_Move	0.98	0.97	0.97
Eye_Move	0.98	0.97	0.98
Looking_Side	0.97	0.97	0.97

Figure 9 depicts a few of our findings resulting from the behavior-detection model trained by the built dataset after applying the proposed augmentation method.

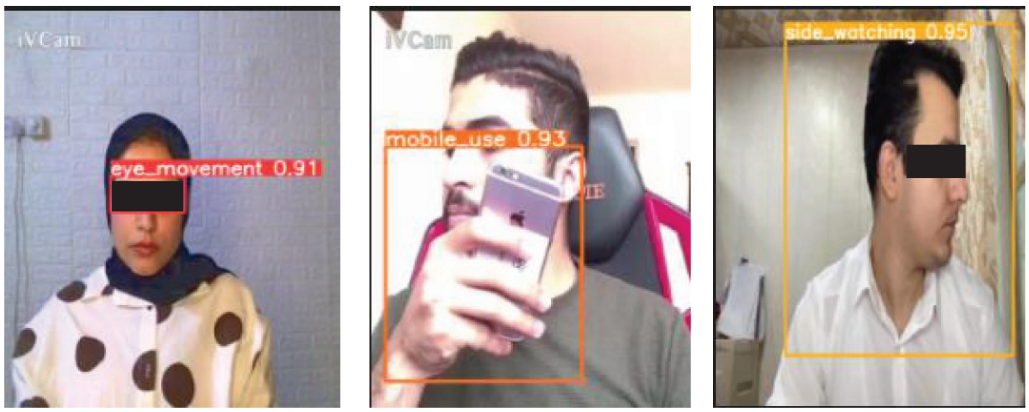

Fig. 9. Sample model-detection results

According to the experimental findings, the suggested augmentation method improved behavioral classification accuracy by more than 0.3%. The deep-learning-based behavioral analysis of students was given new technical support by the

proposed argumentation method. We combined deep-learning strategies with conventional data-augmentation techniques to create a new data augmentation. Table 5 compares the work based on data-augmented methods used.

Table 5. Comparison of the proposed work based on augmented methods used

Exper. No.	Detector	Argumentation Methods Used	Precision	mAP
1	YOLOv5	No data augmentation used	0.61	0.65
2	YOLOv5	Horizontal or vertical flip, hsv-hue, rescale, and hsv-saturation, mosaic	0.75	0.77
3	YOLOv5	Our proposed data augmentation	0.97	0.98

Table 6 compares the present work with earlier studies regarding the detection model, augmentation method, and dataset used.

Table 6. Comparison of the proposed work with related works

Ref.	Detector	Argumentation Methods Used	Dataset
[19]	L2-GraftNet	Mirroring, adding Gaussian noise, adding salt and pepper noise, color shifting	CUI-EXAM
[20]	AUAR	Horizontal or vertical flip, width shift, rescale, and rotation range	EUA (Private)
[21]	Modified YOLOv5	Horizontal flip, brightness enhancement, and Gaussian noise	Private
[23]	L4-BranchedActionNet	Flipping the images	EUA (Private)
This work	YOLOv5	Proposed deep-based augmentation methods	Ours

6 CONCLUSION

This paper presented a strategy to avoid and analyze anomalous student behavior during online exams utilizing deep neural networks and also concentrates on insufficient training data for deep-learning-related issues. Our approach included detection such as eye movement, head movement, hand movement, and mobile-phone use; the detection is achieved by training a YOLOv5 pre-trained model on a produced private dataset. For training deep models, deep learning needs many samples. If there is insufficient training data, the neural network will be susceptible to overfitting problems. In order to address this issue, this paper proposed a data-augmentation technique based on deep learning and image processing. We utilized three techniques: semantic-segmentation-based deep networks, image masking, and pixels-based data augmentation. An image semantic segmentation mask was created using the MobileNetV3 model. This mask was then used to create a binary mask, which was then used to replace the picture background, using conditional subtraction and a set of randomly chosen background images. The final step was randomly applying pixel-based color augmentation to the picture. The three evaluation criteria were recall, precision, and mean average precision. As a result, the behavioral detection model used in this study achieved 0.98 mean average precision (mAP) on the produced dataset, showing acceptable detection accuracy. Also, the experimental findings indicated that the suggested augmentation

method improved behavioral detection accuracy by more than 0.3%. The proposed augmentation method could offer some technological support for established data-augmentation methods.

7 REFERENCES

[1] H. Li, M. X. Xu, Y. Wang, H. Wei, and H. Qu, "A visual analytics approach to facilitate the proctoring of online exams," *Conf. Hum. Factors Comput. Syst. – Proc.*, 2021, https://doi.org/10.1145/3411764.3445294

[2] C. Zhang, "Influences of problem-based online learning on the learning outcomes of learners," *Int. J. Emerg. Technol. Learn.*, vol. 18, no. 1, pp. 152–163, 2023, https://doi.org/10.3991/ijet.v18i01.36705

[3] 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, https://doi.org/10.1007/s10639-021-10458-7

[4] A. Fensie, "A conceptual model for meeting the needs of adult learners in distance education," *Lect. Notes Networks Syst.*, vol. 581 LNNS, no. 2, pp. 136–149, 2023, https://doi.org/10.1007/978-3-031-21569-8_13

[5] H. Alessio and K. Maurer, "The impact of video proctoring in online courses," *J. Excell. Coll. Teach.*, vol. 29, pp. 183–192, 2018.

[6] 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, https://doi.org/10.1109/ACCESS.2021.3079375

[7] S. Sakulwichitsintu, "Mobile technology – an innovative instructional design model in distance education," *Int. J. Interact. Mob. Technol.*, vol. 17, no. 7, pp. 4–31, 2023, https://doi.org/10.3991/ijim.v17i07.36457

[8] L. Shi and C. Fan, "A new learning resource recommendation method for improving the efficiency of students' online independent learning," *Int. J. Emerg. Technol. Learn.*, vol. 18, no. 05, pp. 128–143, 2023, https://doi.org/10.3991/ijet.v18i05.38503

[9] K. Lee and M. Fanguy, "Online exam proctoring technologies: Educational innovation or deterioration?," *Br. J. Educ. Technol.*, vol. 53, no. 3, pp. 475–490, 2022, https://doi.org/10.1111/bjet.13182

[10] 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, https://doi.org/10.4018/IJDET.20210401.oa3

[11] F. Sabrina, S. Azad, S. Sohail, and S. Thakur, "Ensuring academic integrity in online assessments: A literature review and recommendations," *Int. J. Inf. Educ. Technol.*, vol. 12, no. 1, pp. 60–70, 2022, https://doi.org/10.18178/ijiet.2022.12.1.1587

[12] M. Babitha *et al.*, "Trends of artificial intelligence for online exams in education," *Int. J. Early Child. Spec. Educ. (INT-JECSE)*, vol. 14, pp. 2457–2463, 2022,

[13] 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. 1–11, 2022, https://doi.org/10.1128/jmbe.00292-21

[14] F. Noorbehbahani, A. Mohammadi, and M. Aminazadeh, *A systematic review of research on cheating in online exams from 2010 to 2021*, no. 0123456789. Springer US, 2022, https://doi.org/10.1007/s10639-022-10927-7

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

[16] A. Tweissi, W. Al Etaiwi, and D. Al Eisawi, "The accuracy of AI-based automatic proctoring in online exams," *Electron. J. e-Learning*, vol. 20, no. 4, pp. 419–435, 2022, https://doi.org/10.34190/ejel.20.4.2600

[17] A. Mikołajczyk and M. Grochowski, "Data augmentation for improving deep learning in image classification problem," *2018 Int. Interdiscip. PhD Work. IIPhDW 2018*, pp. 117–122, 2018, https://doi.org/10.1109/IIPHDW.2018.8388338

[18] H. Naveed, "Survey: Image Mixing and Deleting for Data Augmentation," 2021, [Online]. Available: http://arxiv.org/abs/2106.07085

[19] 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, https://doi.org/10.1109/ACCESS.2021.3068223

[20] M. Ramzan, A. Abid, and S. M. Awan, "Automatic unusual activities recognition using deep learning in academia," *Comput. Mater. Contin.*, vol. 70, no. 1, pp. 1829–1844, 2021, https://doi.org/10.32604/cmc.2022.017522

[21] 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, https://doi.org/10.3390/app12136790

[22] B. Kadyrov, S. Kadyrov, and A. Makhmutova, "Automated reading detection in an online exam," *International Journal of Emerging Technologies in Learning (iJET)*, vol. 17, no. 22, pp. 4–19, 2022, https://doi.org/10.3991/ijet.v17i22.33277

[23] M. D. Genemo, "Suspicious activity recognition for monitoring cheating in exams," *Proc. Indian Natl. Sci. Acad.*, vol. 88, no. 1, 2022, https://doi.org/10.1007/s43538-022-00069-2

[24] A. H. S. Ganidisastra and Y. Bandung, "An incremental training on deep learning face recognition for M-learning online exam proctoring," *Proc. – 2021 IEEE Asia Pacific Conf. Wirel. Mobile, APWiMob 2021*, pp. 213–219, 2021, https://doi.org/10.1109/APWiMob51111.2021.9435232

[25] W. Wu *et al.*, "Application of local fully convolutional neural network combined with YOLO v5 algorithm in small target detection of remote sensing image," *PLoS One*, vol. 16, no. 10, pp. 1–15, 2021, https://doi.org/10.1371/journal.pone.0259283

[26] R. Li and Y. Wu, "Improved YOLO v5 wheat ear detection algorithm based on attention mechanism," *Electron.*, vol. 11, no. 11, 2022, https://doi.org/10.3390/electronics11111673

[27] 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, p. 110683, 2022, https://doi.org/10.1016/j.measurement.2021.110683

[28] J. Hosang and R. Benenson, "Learning non-maximum suppression," Proceeding, 2017. https://doi.org/10.1109/CVPR.2017.685

[29] C. Shorten and T. M. Khoshgoftaar, "A survey on image data augmentation for deep learning," *J. Big Data*, vol. 6, no. 1, 2019, https://doi.org/10.1186/s40537-019-0197-0

[30] B. Subramanian, B. Olimov, S. M. Naik, S. Kim, K. H. Park, and J. Kim, "An integrated mediapipe-optimized GRU model for Indian sign language recognition," *Sci. Rep.*, vol. 12, no. 1, pp. 1–17, 2022, https://doi.org/10.1038/s41598-022-15998-7

[31] V. B. Yesilkaynak, Y. H. Sahin, and G. Unal, "EfficientSeg: An Efficient Semantic Segmentation Network," 2020, [Online]. Available: http://arxiv.org/abs/2009.06469

[32] A. Howard *et al.*, "Searching for MobileNetV3," *Proc. IEEE Int. Conf. Comput. Vis.*, pp. 1314–1324, 2019, https://doi.org/10.1109/ICCV.2019.00140

[33] Y. Jiang and W. Tong, "Improved lightweight identification of agricultural diseases based on MobileNetV3," pp. 1–5, 2022, https://doi.org/10.48550/arXiv.2207.11238

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

8 AUTHORS

Muhanad Abdul Elah Alkhalisy received the BSc and MSc degrees in Computer Science from and works as a lecturer at the University of Information Technology and Communication (UOITC), Baghdad, Iraq, He often worked in *Informatics Institute for Postgraduate Studies*, Iraqi Commission for Computers and Informatics (ICCI) (https://scholar.google.com/citations?user=dAS7WYYAAAAJ&hl=en; https://orcid.org/0000-0003-2545-8950; email: muhanad_alkhalisy@uoitc.edu.iq).

Saad Hameed Abid is an Associate Professor and Department Head of computer Engineering at Al-Mansur University college, Baghdad, Iraq. He received a Ph.D. degree from Hunan University, Hunan, China. His main experience is in field of Human-Computer-Interaction (https://scholar.google.com/citations?user=1oALcmgAAAAJ&hl=en&oi=sra; https://orcid.org/0000-0002-1394-5500; email: saad.hameed@muc.edu.iq).