





Exam Cheating Detection Based on Action Recognition Using Vision Transformer

Thuong-Cang Phan¹✉, Anh-Cang Phan², and Ho-Dat Tran²

¹ Can Tho University, Can Tho, Vietnam
ptcang@cit.ctu.edu.vn

² Vinh Long University Of Technology Education, Vinh Long, Vietnam
cangpa@vlute.edu.vn, datth@vlute.edu.vn

Abstract. Cheating is the use of prohibited actions to illegally gain in the process of taking tests and exams. These actions cause negative consequences, making learners less capable of learning and creating, leading to unqualified individuals in the social workforce. In this study, we propose an action recognition method based on LSTM, VGG-19, Faster R-CNN, Mask R-CNN, and Vision Transformer architectures to classify fraudulent actions such as copying or pulling up to copy other students' works, communicating, or transferring test material. Experimental results show that Vision Transformer identifies fraudulent actions with an accuracy of above 98%. This contributes to supporting teachers in managing candidates in exams, creating fairness and transparency in education.

Keywords: Vision Transformer · Cheating in exams · Action recognition

1 Introduction

Training quality is always a top concern in every country in the world. In order to assess learners' ability, tests and exams are usually conducted in the middle and at the end of every semester. Learners are required to rely on the knowledge they have learned to answer the requirements posed in the test without any help from anyone else or materials that are not allowed to be brought into the exam room. Exam fraud is the act of bringing unauthorized documents into the exam room, communicating with other candidates, copying other candidates' work, etc. [1, 2]. In Vietnam, Circular No. 15/2020/TT-BGDDT¹ issued on May 26, 2020, defines exam cheating as an act contrary to the regulations of the examination board such as copying and bring unauthorized materials into the exam room. Cheating

¹ <https://thuvienphapluat.vn/van-ban/EN/Giao-duc/Circular-15-2020-TT-BGDDT-promulgation-of-Regulation-on-high-school-graduation-exam/445907/tieng-anh.aspxaccessedon22June2023>.

in exams causes bad effects on students, making them in a state of dependence and a lack of willingness to strive for good academic results.

The application of advanced technology to detect cheating in exams has been included in research. Rehab and Ali [14] proposed to identify cheating behaviors in online exams by three different classification algorithms: Support Vector Machine (SVM), Random Forest (RF) and K-nearest neighbor (KNN). Students' behaviors were classified as cheating or not with an accuracy of up to 87%. Yulita et al. [20] detected cheating in online exams using deep learning techniques with the MobileNetV2 architecture. Student's activities in web camera in an online exam were monitored and cheating actions were detected with a F1-score of 84.52%. Tiong and Lee [17] proposed a fraud detection model in online exams using deep learning techniques. Students' behaviors were monitored to detect and prevent learners' cheating. The system achieved an accuracy of 68% for deep neural networks (DNNs), 92% for Long Short Term Memory (LSTM), 95% for DenseLSTM, and 86% for recurrent neural networks (RNNs). Waleed Alsabhan [3] employed SVM, LSTM, and RNN classifiers to identify whether a student is cheating or not in higher education. The system achieved an accuracy of 90%. Dilini et al. [4] explored the use of eye tracking to detect cheating in exams. This study analyzed eye movement patterns and staring behaviors to identify suspicious activities, such as viewing unauthorized material. This approach achieved a positive results with an accuracy of 92.04%. Li Zhizhuang et al. [12] proposed a method to detect cheating in multiple-choice tests based on LSTM and RAE algorithm (combining of linear regression and expectation-maximization algorithm). This was to determine each student's mastery of knowledge points based on the exam problem solving. The system achieved an average accuracy of 81%. Ozdamli et al. [13] used computer vision algorithms and deep learning algorithms to detect emotions and feelings of cheating students in distance learning. The system achieved an accuracy of 87.5% in real-time student tracking head, face, and expressions of fear emotion during exams. Kamalov et al. [11] proposed an approach to detect potential cheating cases in final exams using machine learning techniques. This model applied recurrent neural networks along with anomaly detection algorithms and achieved an average true positive rate of 95%. Hussein et al. [10] proposed a method to automatically detect cheating by classifying video sequences. This helped to detect cheating behavior of students in paper-based exams. The authors achieved an average accuracy of 91%.

2 Background

2.1 Exam Fraud

Cheating in exams is an action contrary to the regulations of the examination board. Several types of cheating in exams are copying/pulling up to copy other students' works, viewing cheat sheets, and communicating/exchanging test materials with others. These actions negatively affect the fairness of the assessment process and lead to dishonest test results.

2.2 Network Models

2.2.1 LSTM Long Short Term Memory (LSTM) [7] is a type of recurrent neural network (RNN) architecture. LSTM was introduced by Hochreiter and Schmidhuber in 1997. It has been successfully applied to various sequential data tasks, such as natural language processing, speech recognition, machine translation, and video analysis. It is outstanding at capturing long-term dependencies in sequence, which is important for tasks involving context or temporal relationships. LSTM has played an important role in advancing the field of deep learning for sequential data analysis. The ability to model and capture complex temporal dependencies has resulted in improved performance in various domains, making them an influential and widely adopted architecture in the field of study.

2.2.2 VGG-19 VGG-19 [16] is a convolutional neural network (CNN) architecture introduced by K. Simonyan and A. Zisserman. The VGG-19 model is a variant of the VGG network, which has 19 layers including 16 convolutional layers and 3 fully connected layers. It gained popularity for its simplicity and high performance in large-scale image recognition tasks. It was designed with the goal of exploring the effect of network depth on performance in image classification tasks. VGG family, including VGG19, has achieved excellent performance on a variety of benchmark datasets. Researchers often use VGG19 pre-trained weights, which are trained on large-scale image datasets, as a starting point for computer vision tasks.

2.2.3 Faster R-CNN Faster R-CNN [15] is an efficient and widely applied object detection model in computer vision. It was introduced by Shaoqing Ren et al. in 2017. Faster R-CNN is flexible and can be easily adapted to different object detection tasks. By modifying the architecture and training data, it can be used for different types of objects and scales, making it suitable for a wide range of applications, which in turn can handle detection tasks in real time or near real time. This helps to make it suitable for applications that require fast and efficient processing. It can process images in a batch-wise manner, allowing for scalability and efficient inference on large datasets.

2.2.4 Mask R-CNN Mask R-CNN [9] was introduced by Kaiming He et al. in 2017. Mask R-CNN achieves good performance on several benchmark datasets, consistently outperforming previous methods in terms of accuracy and durability. It can be applied to many computer vision tasks beyond segmentation, including object detection, object tracking, and semantic segmentation. Its flexibility and precision make it a suitable choice for various applications such as autonomous driving, robotics, and medical imaging.

2.2.5 Vision Transformer Vision Transformer (ViT) [16] is a neural network architecture introduced by A. Dosovitskiy et al. (2020). It is a successful

extension of Transformers, originally designed for natural language processing, into the field of computer vision. Vision Transformer splits the input image into smaller patches and processes them as a token string. It applies the standard Transformer architecture, which includes self-attention mechanisms and feed-forward neural networks, to capture the local and global relationships between these image arrays.

ViT has demonstrated outstanding performance on various computer vision standards, competing with or even surpassing convolutional neural networks in certain settings [8]. Unlike traditional convolutional neural networks, ViT can process images of arbitrary size. This scalability makes it suitable for tasks that require different input resolutions without modifying the training architecture [5]. ViT’s architecture allows it to be applied to various computer vision tasks without significant change. By fine-tuning the pre-trained ViT on tasks, it can adapt to image classification, object detection, and other related tasks that other neural networks force the user to reprocess. The accuracy of the Vision Transformer model compared to other models may vary depending on specific datasets, tasks, and test setup. According to Li Yuan et al. [19], ViT demonstrated competitive accuracy on the ImageNet dataset compared to traditional CNN models, demonstrating ViT’s potential to achieve high accuracy without relying on transfer learning. Hugo Touvron et al. [18] indicate that the ViT model can achieve the same accuracy as a CNN while requiring fewer labeled training samples. This highlights ViT’s potential for effective learning and generalization.

2.3 Evaluation Metrics

The simplest and most commonly used metric in evaluating network models is accuracy [6]. This evaluation simply calculates the ratio between the number of correctly predicted samples and the total number of samples in the dataset.

$$accuracy = \frac{TP + TN}{TP + TN + FP + FN} \quad (1)$$

where:

- TP: True positive
- TN: True negative
- FP: False positive
- FN: False negative

Mean average precision (mAP) is a commonly used method in multi-class classification problems. The mAP measure is calculated using Eq. 2 after obtaining the AP (average precision) where N is the number of classes. The AP measure is calculated by Eq. 3, with $\rho_{interp}(r)$ performing 11-point interpolation to summarize the shape of the Precision x Recall curve by averaging the accuracy at a set of 11 equally spaced points $[0, 0.1, 0.2, \dots, 1]$ and $\rho(\tilde{r})$ is the precision measured on \tilde{r} .

$$mAP = \frac{1}{N} \sum_{i=1}^N AP_i \quad (2)$$

$$AP = \frac{1}{11} \sum_{r \in \{0, 0.1, \dots, 1\}} \rho_{interp}(r) \quad \text{with} \quad \rho_{interp}(r) = \max_{\tilde{r}: \tilde{r} \geq r} \rho(\tilde{r}) \quad (3)$$

3 Proposed Method

In this study, we use deep learning techniques and Vision Transformer to detect sequences of cheating actions in exams. The proposed approach consists of two phases: training and testing. The details of the phases are shown in Fig. 1.

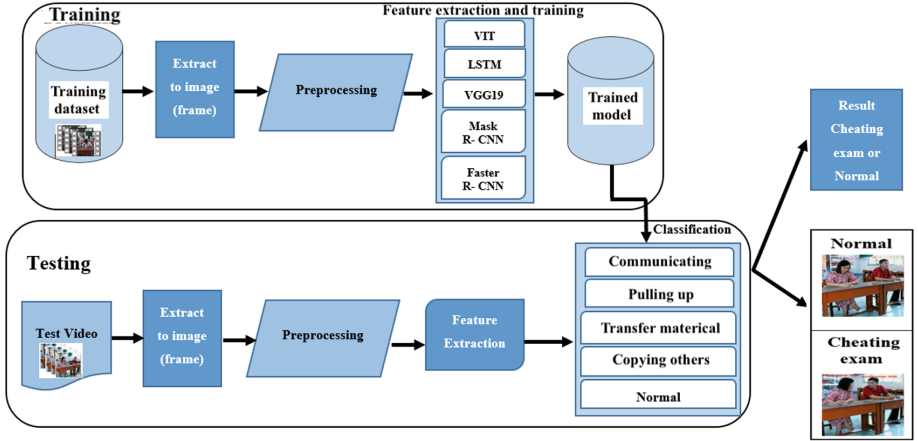


Fig. 1. The proposed approach in action identification and classification.

3.1 Training Phase

3.1.1 Pre-processing The dataset consists of videos recorded from the students' exams or tests. Each video is 5 to 10 s long at 30 frame per second (FPS). We normalize the input images to 224×224 pixels to maintain the ability compatible and consistent across different models. To enhance the dataset, image flipping and rotating were used to obtain more training observations. Deep learning models require large amounts of training data to perform well. When there is a lack of data, models can suffer from overfitting, where they perform exceptionally well on the training data but fail to generalize to new, unseen data. Image rotating and flipping are basic data augmentation techniques that helps mitigate this issue by generating new, diverse samples, effectively simulating a larger dataset.

3.1.2 Feature Extraction and Training With the advantages of deep learning networks presented earlier, we conduct feature extraction of action types based on extracted images with five models including LSTM, VGG-19, Faster R-CNN, Mask R-CNN, and Vision Transformer. These network models have a classification layer that helps in classifying fraudulent actions. Especially with the Vision Transformer model, we cut the input images into patches and passed them through the layers to conduct encryption, feature extraction, and classification (Fig. 2). The result of this process will be the classified and partitioned image.

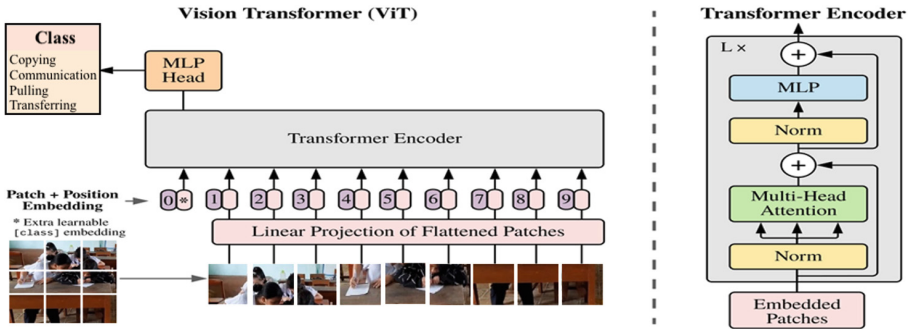


Fig. 2. Proposed Vision transformer model.

3.2 Testing Phase

To evaluate the performance of network models, we test the trained models using the testing dataset. This includes videos that we have collected combined with the segmentation of people on the YOLO algorithm. The test results will identify the actions in the video as normal actions or fraudulent actions.

4 Experiments

4.1 Installation Environment and Dataset Description

The system is installed in Python language and runs on the Colab environment with Windows 10 and the configuration of 12 GB RAM and Nvidia Geforce GPU. The libraries to support training network models are Tensorflow and Keras.

The dataset used in this study is video clips taken from students' exams or tests. These videos are divided into fraudulent actions such as copying other student works (1,480 images), exchanging test materials (1,850 images), pulling up to copy other students (1,560 photos), communicating with others (2,078 images), and acting normally (1,680 images). This dataset is divided into a training dataset and a validation dataset with a ratio of 80% and 20%, respectively.

For testing, we build a hypothetical scenario where the students were doing the exam, then recorded a video, and tested it with a number of 15 videos on a cheating action to check the accuracy of the network models.

4.2 Scenarios

To conduct the experiment, we perform five scenarios with training parameters as shown in Table 1.

Table 1. Proposed scenarios and training parameters

Scenarios	Models	Epochs	Learning_rate	Batch_size	Image_size
1	Vision transformer	200	0.001	9	224×224
2	LSTM	400	0.001	4	64×64
3	VGG-19	400	0.001	4	224×224
4	Mask R-CNN	400	0.001	4	224×224
5	Faster R-CNN	400	0.001	4	224×224

4.3 Training Results

Figure 3 shows the loss values of five scenarios during the training phase. The validation loss values of scenarios 1 to 5 are 0.0048, 0.1872, 0.1426, 0.1989, and 0.1643, respectively. In scenarios 2 and 4, the validation loss is high and higher than the training loss, which shows that the models are not optimal and can lead to errors in the prediction process. Scenarios 3 and 5 show the loss values better than scenarios 2 and 4, but the validation loss values are still higher than the training loss. In scenario 1, the training loss and validation loss values are more stable and approaching 0 with 200 epochs, which is less by half than other models. With this result, scenario 1 has a very low loss value, less than 5% compared to other models after going through 200 training steps. This means that the error rate when predicting scenario 1 is the lowest compared to the remaining models. In the remaining scenarios, the validation loss value is still high, thus the models will have a higher error rate when making predictions.

Figure 4 shows the accuracy of five scenarios during the training phase. Scenarios 1 to 5 achieved an accuracy of 98.2%, 89.6%, 93.8%, 91.3%, and 92.7%, respectively. In scenarios 2 and 4, the validation accuracy is low and lower than the training accuracy, which can lead to incorrect identification of actions. Scenarios 1, 3, and 5 give more optimal results with train accuracy relatively close to validation accuracy. However, in scenarios 3 and 5, the validation accuracy value is still low. In scenario 1, the validation accuracy and train accuracy values gradually move towards 1. Thus, this model is better at prediction than the other models.

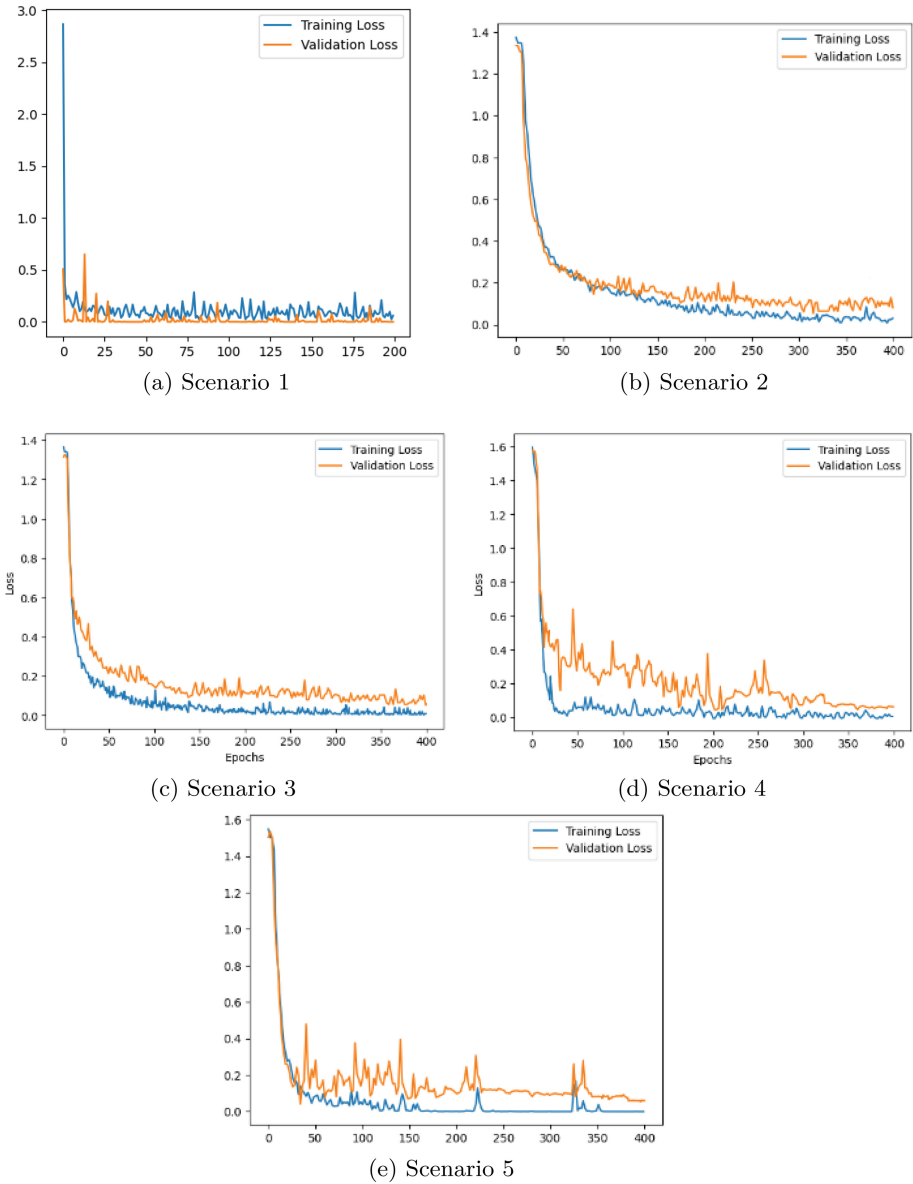
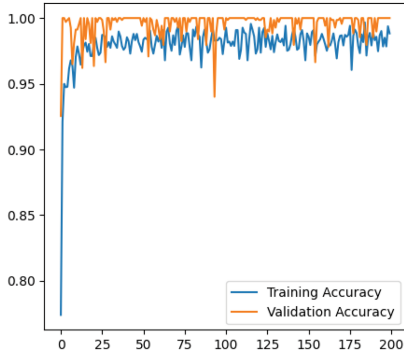
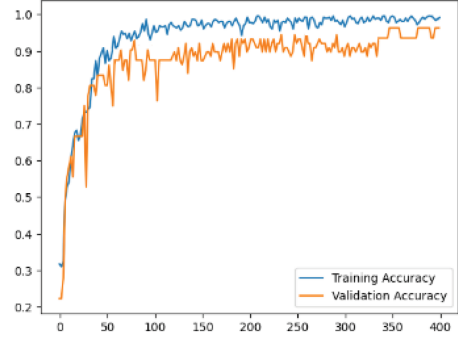


Fig. 3. Loss of five scenarios.

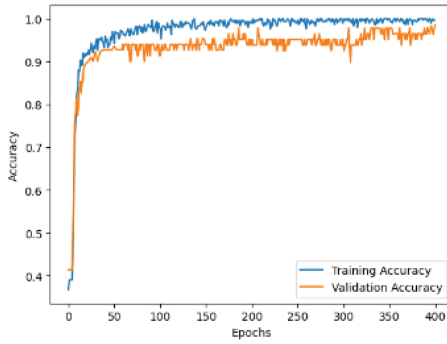
Figure 5 shows the training time of the proposed scenarios. The training time of scenario 1 is 117.6 min, scenario 2 is 93.6 min, scenario 3 is 125.6 min, scenario 4 is 114.3 min, and scenario 5 is 98.4 min. Through the above results, scenarios 2 and 5 give faster training time than the other scenarios. Scenario model 1 gives



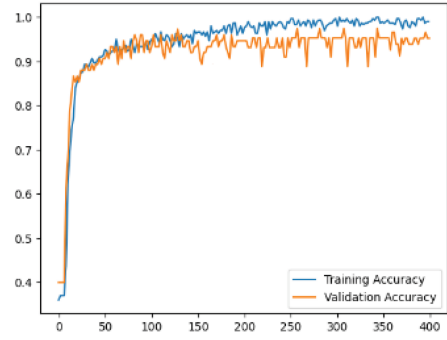
(a) Scenario 1



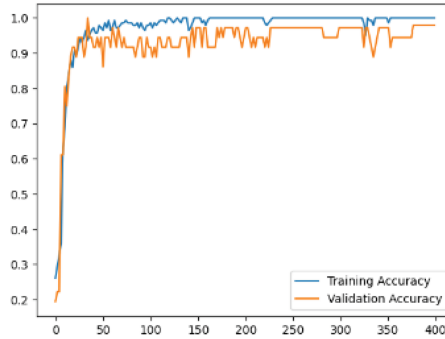
(b) Scenario 2



(c) Scenario 3



(d) Scenario 4



(e) Scenario 5

Fig. 4. Accuracy of five scenarios.

a close training time compared to other scenarios with less than half number of training steps.

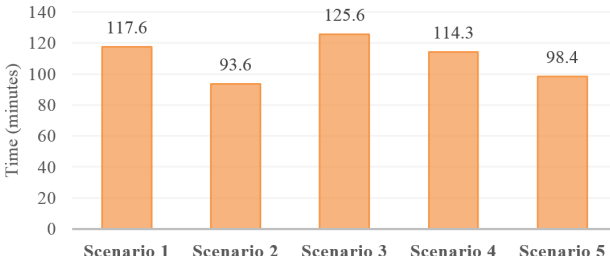


Fig. 5. Training time of five scenarios.

4.4 Testing Results

Table 3 shows some illustrations of the five scenarios. In the case of communication, all five scenarios can recognize the action, and scenario 1 has the highest accuracy. Scenarios 2, 3, and 4 can recognize the action of one student but the second student’s action is incorrectly identified as pulling up. In scenario 5, only one student is identified with cheating actions while the other student’s action cannot be recognized. In the case of transferring material, all scenarios can recognize the action and give relatively good results. However, compared to the scenarios, scenario 1 predicts the best outcome, nearly 99%. In the case of copying others, scenario 1 gives the correct result with an accuracy of 100%. Scenario 2 does not recognize fraudulent actions, possibly because the identification and classification activities give a low accuracy rate. The remaining scenarios can recognize and give relatively good results. In the case of pulling up and normal, scenario 2 cannot recognize the actions, possibly due to the actions being relatively unclear. Scenarios 1, 3, 4, and 5 all recognize the actions with relatively

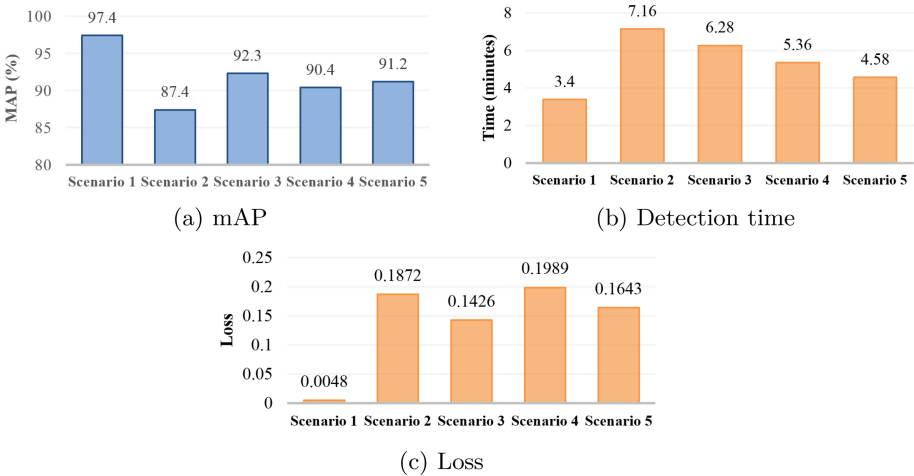



















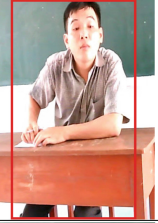







Fig. 6. mAP, Loss, and detection time on testing dataset.

Table 2. Average accuracy of the scenarios on fraudulent actions

Scenarios	Communicating	Pulling up	Copying others	Transfer material
1	98.2%	97.1%	96.5%	97.8%
2	87.1%	87.8%	86.2%	88.5%
3	92.7%	91.5%	90.8%	94.2%
4	90.5%	89.6%	91.8%	89.7%
5	91.6%	92.1%	90.8%	90.3%

Table 3. Illustration of classification results from the experimental dataset.

Case	Scenario 1	Scenario 2	Scenario 3	Scenario 4	Scenario 5
Communicating					
Transferring material					
Copying others					
Pulling up					
Normal					

high accuracy. Scenario 1 gives the highest accuracy among the scenarios. The above results show that scenario 1 gives better performance than the remaining scenarios in identifying and classifying fraudulent actions.

The comparison results are summarized in Fig. 6. The mAP of the five scenarios is shown in Fig. 6a. Scenario 1 shows the highest mAP of 97.4% compared to scenario 2 (87.4%), scenario 3 (92.3%), scenario 4 (90.4%), and scenario 5 (91.2%). On the testing dataset, the shortest detection time is of scenario 1 with 3.4 min (Fig. 6b), which is almost two times faster than scenarios 2, 3, and 4 and almost 1.5 times faster than scenario 5. The loss value of scenario 1 is 0.0048 (Fig. 6b), which is the lowest among the five scenarios. From the above results, scenario 1 using Vision Transformer can identify and classify fraudulent actions more effectively than the remaining models.

Table 2 shows the average accuracy of the fraudulent actions in five scenarios. Scenario 1 provides better accuracy in four classes compared to the other four scenarios. Scenario 2 has the lowest accuracy in the four fraudulent classes.

5 Conclusion

The consequences of cheating in exams for students are enormous. It creates bad habits and bad qualities for students, affecting the process of being human. In this work, we bring our contributions to building a training dataset of action recognition with the use of advanced network architectures such as LSTM, VGG-19, Mask R-cNN, Faster R-CNN, and Vision Transformer. Fraudulent behaviors considered in this study are copying others, pulling up, communicating, and exchanging test materials. The detection of fraudulent actions all gives extremely positive results with an accuracy of above 85%, in which Vision Transformer has the best accuracy of above 98%. This helps to detect fraudulent action sequences in a timely and effective manner. Although bringing high results in the identification process, this work only stops at identifying common fraudulent behaviors, we will continue to recognize various cheating actions and more sophisticated cheating techniques. Future development could focus on addressing these limitations by incorporating additional features, exploring new architectures, and combining data from multiple sources, such as audio or oral gestures to identify cheating.

References

1. <https://www.adelaide.edu.au/student/academic-skills/cheating-in-exams>. Accessed 22 June 2023
2. <https://www.niu.edu/academic-integrity/faculty/types/index.shtml>. Accessed 22 June 2023
3. Alsabhan, W.: Student cheating detection in higher education by implementing machine learning and LSTM techniques. *Sensors* **23**(8), 4149 (2023)
4. Dilini, N., Senaratne, A., Yasarithna, T., Warnajith, N., Seneviratne, L.: Cheating detection in browser-based online exams through eye gaze tracking. In: 2021 6th International Conference on Information Technology Research (ICITR), pp. 1–8. IEEE (2021)

5. Dosovitskiy, A., et al.: An image is worth 16x16 words: transformers for image recognition at scale. arXiv preprint: [arXiv:2010.11929](https://arxiv.org/abs/2010.11929) (2020)
6. Goutte, C., Gaussier, E.: A probabilistic interpretation of precision, recall and *F*-score, with implication for evaluation. In: Losada, D.E., Fernández-Luna, J.M. (eds.) ECIR 2005. LNCS, vol. 3408, pp. 345–359. Springer, Heidelberg (2005). https://doi.org/10.1007/978-3-540-31865-1_25
7. Graves, A.: Long short-term memory. In: Graves, A. (ed.) Supervised Sequence Labelling with Recurrent Neural Networks. Studies in Computational Intelligence, vol. 385, pp. 37–45. Springer, Berlin (2012). https://doi.org/10.1007/978-3-642-24797-2_4
8. Han, K., et al.: A survey on vision transformer. *IEEE Trans. Pattern Anal. Mach. Intell.* **45**(1), 87–110 (2022)
9. He, K., Gkioxari, G., Dollár, P., Girshick, R.: Mask R-CNN. In: Proceedings of the IEEE International Conference on Computer Vision, pp. 2961–2969 (2017)
10. Hussein, F., Al-Ahmad, A., El-Salhi, S., Alshdaifat, E., Al-Hami, M.: Advances in contextual action recognition: automatic cheating detection using machine learning techniques. *Data* **7**(9), 122 (2022)
11. Kamalov, F., Sulieman, H., Santandreu Calonge, D.: Machine learning based approach to exam cheating detection. *PLoS ONE* **16**(8), e0254340 (2021)
12. Li, Z., Zhu, Z., Yang, T.: A multi-index examination cheating detection method based on neural network. In: 2019 IEEE 31st International Conference on Tools with Artificial Intelligence (ICTAI), pp. 575–581. IEEE (2019)
13. Ozdamli, F., Aljarrah, A., Karagozlu, D., Ababneh, M.: Facial recognition system to detect student emotions and cheating in distance learning. *Sustainability* **14**(20), 13230 (2022)
14. Rehab, K.k., Ali, Z.H.: Cheating detection in online exams using machine learning. *J. AL-Turath Univ. Coll.* **2**(35) (2023)
15. Ren, S., He, K., Girshick, R., Sun, J.: Faster R-CNN: towards real-time object detection with region proposal networks. In: Advances in Neural Information Processing Systems, vol. 28 (2015)
16. Simonyan, K., Zisserman, A.: Very deep convolutional networks for large-scale image recognition. arXiv preprint: [arXiv:1409.1556](https://arxiv.org/abs/1409.1556) (2014)
17. Tiong, L.C.O., Lee, H.J.: E-cheating prevention measures: detection of cheating at online examinations using deep learning approach—a case study. arXiv preprint: [arXiv:2101.09841](https://arxiv.org/abs/2101.09841) (2021)
18. Touvron, H., Cord, M., Douze, M., Massa, F., Sablayrolles, A., Jégou, H.: Training data-efficient image transformers & distillation through attention. In: International Conference on Machine Learning, pp. 10347–10357. PMLR (2021)
19. Yuan, L., et al.: Tokens-to-token ViT: training vision transformers from scratch on ImageNet. In: Proceedings of the IEEE/CVF International Conference on Computer Vision, pp. 558–567 (2021)
20. Yulita, I.N., Hariz, F.A., Suryana, I., Prabuwo, A.S.: Educational innovation faced with COVID-19: deep learning for online exam cheating detection. *Educ. Sci.* **13**(2), 194 (2023)