Object Detection in Online Proctoring Through Two Camera Using Faster-RCNN

I Wayan Suardinata[1], Vivien Arief Wardhany [2]
[1] Digital Business Study Program,
Department of Informatics Engineering, State Polytechnic of Banyuwangi, Indonesia
[2] Computer Engineering Study Program,
Department of Informatics Engineering, State Polytechnic of Banyuwangi, Indonesia

[1]wayan.suardinata@poliwangi.ac.id, [2]vivienwardhany@poliwangi.ac.id

Abstract—The COVID-19 pandemic has prompted changes in teaching methods from offline to online, including the implementation of exams. But many reports say that the potential for online exam cheating is very high which can compromise the credibility of the exam, the quality of the examinees and the testing institution itself. The online exam monitoring system using one camera makes it difficult for officers to make decisions because of the lack of evidence and supporting data. In this study, we propose a monitoring approach using two cameras, namely a camera on a laptop to get a front view of the participant and a cellphone camera to get a side view of the examinee but because of the complexity of the problem, at this stage we only focus on the side camera. Implementation begins with the collection of video recording data, custom data sets for training and pretrained datasets from the zoo model. Training is carried out using a custom dataset to detect objects that are not recognized by the pretrained dataset. The evaluation of the training results using the COCO evaluator showed the average of the bbox-AP is 59,169. The fraud detection process is carried out using 6 exam videos with a total of 192,929 frames, producing two outputs, namely object detection videos and csv files. The csv file contains the frame number, time, object detected in each frame. The next process is to analyze the csv file and mark frames that have the potential to be fraudulent. The evaluation results show an accuracy of 0.884615385 and a recall of 0.821428571.

Keywords— ***computer vision, machine learning, object detection, pose detection, proctoring***

I. INTRODUCTION

The COVID-19 pandemic has prompted changes in teaching methods from offline to online, including the administration of exams. Online exams are carried out to avoid the spread of dangerous diseases, limited time and space and efficiency. In conditions where there is no risk of disease transmission, online exams remain an alternative if the time and place for the exam are not sufficient. In addition, several important exams that are conducted offline such as competency certification, employee recruitment and new student admissions, and others, still require that the exam be conducted under camera surveillance which is monitored by the center to avoid potential fraud committed by participants and the local organizing committee. Previous studies [1] have shown that online exams are prone to cheating. Based on a survey by King and Case [2], about 74% of students in 2013 reported that it was easy to cheat online exams and about 29% of students indicated cheating. This fraudulent behavior can undermine the credibility of online exams and the quality of their results.

Recently [3,15], a social media campaigner published an easy way to cheat online exam supervisor software, a method that will work if there is no second camera in the room. The examinee places his cell phone on the laptop screen and connects his laptop to the television. The second person (his friend) uses the television to view the exam questions, look for the answers and send messages containing the answers to the examinees. This fraud is possible due to the limited range of the camera on the laptop so that there are many blind spots such as below, behind, and some on the right and left side of the laptop webcam camera. In general, fraud techniques can be classified into two categories, namely internal and external fraud. Internal fraud is fraud that is carried out on a computer such as looking for answers on the internet, opening files on a computer, doing remote devices and others. While external cheating is carried out by bringing tools and materials that are not permitted to the examination table such as cell phones, watches, additional notes, and others. Given the many parts that must be checked and the limited resources we have, we focus on external fraud, especially the detection of illegal objects in online exams such as cellphones, watches, and others.

External fraud detection generally uses several sensors on the laptop, namely sound sensors, and cameras. The sound sensor (microphone) is used to detect cheating if someone helps through the voice while the camera is used to detect objects that are not allowed to be carried or used during the exam. Features that can be detected through the camera may include detection of head pose, eye movement or mouth movement. Through these features, warnings of potential fraud can be raised, such as if participants turn their heads other than the front. However, this technique cannot be used to determine whether fraud has occurred or not, so it requires additional evidence and verification. Supervisors will find it difficult to make decisions in these situations for fear of being unfair. So

that surveillance with one camera has difficulty in gathering evidence to make decisions for supervisors.

In this study, we propose a monitoring approach using two cameras, namely a camera on a laptop to get a front view of the participant and a cellphone camera to get a side view of the examinee to assist the supervisor in making decisions by providing additional data, evidence and verification. Given the many features that must be detected as well as the limited time and resources we have, this research is focused on detecting objects that are not allowed in the exam.

A. Online Proctoring

Online study and exams are now gaining popularity in our world of Education. In online exams, supervision can be categorized into three types: online human proctoring, semi-automated proctoring, and full automated proctoring. Manual supervision means that there will be a remote officer who will supervise the examinee during the entire exam. This method is the method commonly used today. For example, the 2021 State Polytechnic of Banyuwangi contract employee acceptance test. Exam participants carry out the exam from home by accessing the exam application using their respective laptops. Then participants were asked to run the online meeting application using ZOOM to get videos of participants working on the exam in real time. The committee provides one officer to supervise two examinees through the zoom application. Supervisors watch videos from start to finish and report potential cheating if they see suspicious behavior. This method of course will require a lot of supervisors and of course will cost a lot if there are many participants.

To eliminate the use of human labor, some fully automated proctoring was proposed [4,5,10,13,14]. This method often uses machine learning techniques to identify fraudulent behavior. Currently, several online surveillance platforms, such as ProctorU (https://www.proctoru.com) and Proctorio (https://www.proctorio.com), use automated surveillance using machine learning. However, all automated monitoring approaches suffer from the same problems in using machine learning in education. The problem is the "black box" nature of machine learning algorithms and unreliable decision making due to biased training datasets [6]. Considering this, it is nearly impossible for us to only use automated techniques to ascertain whether a student is cheating or not.

To overcome the problem of fully automated proctoring, semi-automatic supervision is proposed that involves humans in final decision making [7, 8, 9]. One of the previous related studies is Massive Open Online Proctor proposed by Li et al. [8]. Specifically, they first use machine learning to detect fraudulent behavior and then the detection results are verified by teachers or supervisors. But this method does not provide supervisors with a convenient way to explore and analyze the student's cheating behavior.

B. Object Detection

A computer vision technology called object detection helps locate and identify things in an image or video. To be more precise, object detection creates bounding boxes around the items it has found, allowing us to determine their location inside (or how they move across) a scene. Before we continue, it's crucial to make the distinctions between object detection and picture recognition clear as they are sometimes misconstrued. An image is given a label through image recognition. The word "dog" is used to describe a picture of a dog. The word "dog" is still used to describe a picture of two canines. On the other hand, object detection surrounds each dog with a box that is labeled "dog". The model forecasts the location of each object and the appropriate label. To accomplish object detection, a number of models have been created, including :

- R-CNN, Mask R-CNN, Faster R-CNN. The R-CNN family of object detection models includes several well-known models. These architectures, sometimes known as region convolutional neural networks, are built on the region proposal structure mentioned above. They have improved in accuracy and computational efficiency over time. The most recent version, Mask R-CNN, was created by Facebook researchers and serves as a suitable foundation for server-side object identification models. number of popular object detection models belong to the R-CNN family. Short for region convolutional neural network, these architectures are based on the region proposal structure discussed above. Over the years, they've become both more accurate and more computationally efficient. Mask R-CNN is the latest iteration, developed by researchers at Facebook, and it makes a good starting point for server-side object detection models.
- MobileNet + SSD, YOLO, SqueezeDet. The single shot detector family includes a variety of other models. These versions' encoders and the particular way in which the predefined anchors are configured differ the most from one another. SqueezeDet uses the SqueezeNet encoder, the YOLO model has its own convolutional architecture, and the MobileNet + SSD models use a MobileNet-based encoder. Models intended for mobile or embedded devices are excellent candidates for SSDs.
- CenterNet. Recently, scientists have created object detection models that completely do away with the requirement for region recommendations. When estimating the X, Y coordinates of an object's center and its extension (height and width), CenterNet interprets objects as single points. In comparison to SSD or R-CNN methods, this method has demonstrated to be both more effective and accurate.

C. Human pose detection

Human Pose Estimation[16] identifies and classifies the poses of human body parts and joints in images or videos. In order to represent and infer human body positions in 2D and 3D space, a model-based technique is typically used. It basically involves describing the joints of the human body, such as the wrist, shoulder, knees, eyes, ears, ankles, and arms, which are crucial in pictures and movies that can depict a person's position. The posture estimator model then outputs the coordinates of these identified body parts and joints as well as a confidence score demonstrating the accuracy of the estimations after receiving an image or video as input.

For many years, the main topic of debate for numerous classical object detection applications has been the detection of persons. Using stance detection and pose tracking, computers can now read human body language thanks to recent advancements in machine learning algorithms. These detections' accuracy and hardware needs have now improved to the point where they are economically practical. Additionally, the coronavirus pandemic, where high-performing real-time pose detection and tracking will deliver some of the most significant developments in computer vision, has a profound impact on the technology's progress. By integrating human position estimate and distance projection algorithms, it can be used, for example, for social distance. In a crowded setting, it helps people keep their physical distance from one another.

D. Faster RCNN

The RPN serves as a region proposal method, while the Fast R-CNN serves as a detector network in the Faster R-CNN design. The Fast R-CNN detector [11] also consists of a CNN backbone, an ROI pooling layer and fully connected layers followed by two sibling branches for classification and bounding box regression as shown in Fig. 1.

To obtain the feature map (Feature size: 60, 40, 512), the input image is first run through the backbone CNN. The benefits of weight sharing between the RPN backbone and the Fast R-CNN detector backbone are another important factor that justifies utilizing an RPN as a proposal generator in addition to test time effectiveness. Following that, features from the backbone feature map are gathered using the bounding box proposals from the RPN. The ROI pooling layer carries out this task. The backbone feature map region corresponding to a proposal is taken from the ROI pooling layer, which then divides this region into a given number of sub-windows and performs max-pooling over these sub-windows to produce an output of a fixed size. To understand the details of the ROI pooling layer and it's advantages, read Fast R-CNN. Reading Fast R-CNN can help us comprehend the ROI pooling layer's specifics and benefits.

The size of the ROI pooling layer's output is (N, 7, 7, 512), where N is the quantity of region proposal algorithm proposals. The features are fed into the sibling classification and regression branches after being passed through two fully linked layers. Notably, these classification and detection branches diverge from RPN counterparts. Here, the classification layer has C units for each of the detection task's classes (along with a general-purpose background class). The classification scores—the likelihood that a proposal belongs to each class—are obtained by passing the characteristics through a softmax layer. The projected bounding boxes are enhanced using the regression layer coefficients. In this case, the regressor is class-specific rather than size-specific (unlike the RPN). In other words, the regression layer has separate regressors for each class, each with 4 parameters and C*4 output units.

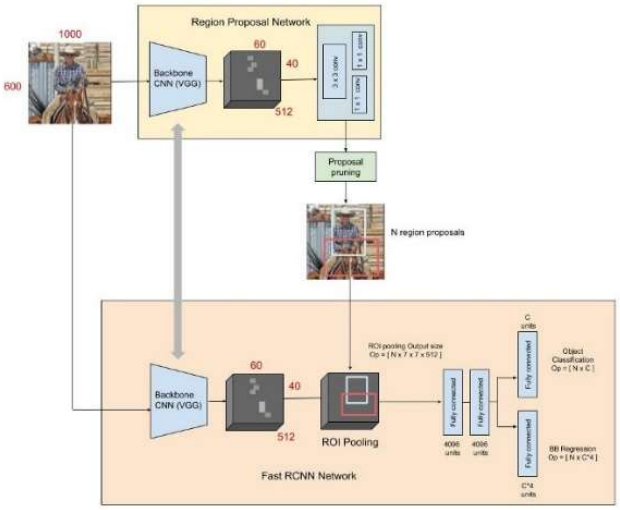

Figure 1. The RPN for region proposals and Fast R-CNN as a detector in the Faster R-CNN detection pipeline

E. Detectron

A brand-new object detection algorithm from Facebook AI Research (FAIR) dubbed Detectron was released in 2018. It was a fantastic library that used cutting-edge object identification techniques, like Mask R-CNN. It was created using the Caffe2 deep learning framework and Python. Many research initiatives, including Feature pyramid network (FPN), Data Distillation, Omni-Supervised Learning, and Mask R-CNN, were eventually published as a result of Detectron. ResNet (50, 101, 152), ResNeXt (50, 101, 152), and FPN (Feature Pyramid Networks) with Resnet/ResNeXt and VGG16 served as the foundation for the Detectron backbone network framework.

The goal of detectron was pretty simple to provide a high-performance codebase for object detection, but there were many difficulties like it was very hard to use since it's using caffe2 & Pytorch combined, and it was becoming difficult to install. The objective of detectron was very straightforward: to offer a high-performance codebase for object detection. However, there were a number of challenges, including the fact that it was highly challenging to use because it integrated Pytorch and Caffe2 and was becoming challenging to install.

F. Object Detection Evaluation metrics

Average Precision (AP) and mean Average Precision (MAP), two metrics, are used to assess how well the object detection and localization method performs. Average Precision (AP) (and mean average precision) is a metric used to measure how well the object detection and localization algorithm performs. Before we get into the detail of what AP is, let's make one thing clear about what it is NOT. AP is calculated with the help of several other metrics such as IoU, confusion matrix (TP, FP, FN), precision and recall, etc. as shown in the Fig. 2.

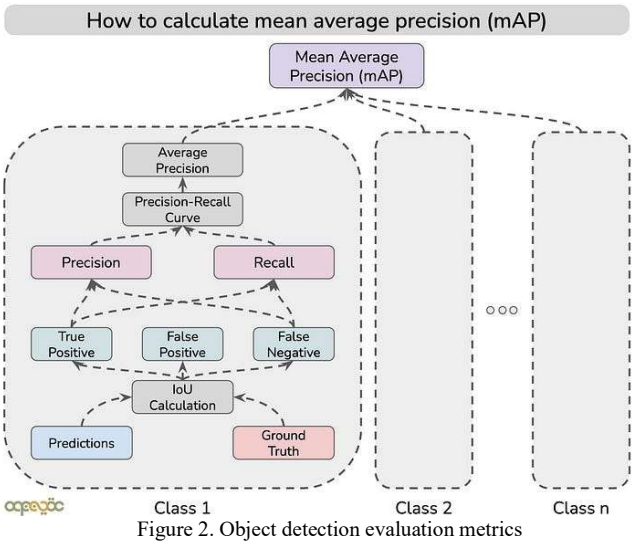

Figure 2. Object detection evaluation metrics

We must first comprehend these measurements in order to comprehend AP.

- Intersection over Union (IoU). The IoU measures how closely the two bounding boxes (ground truth and prediction) are spaced apart. It has a value ranging from 0 to 1. If both bounding boxes entirely overlap, the prediction is accurate, and the IoU is 1. On the other hand, the IoU is zero if the two bounding boxes do not overlap. The area of intersection and the area of the union of two bonding boxes are compared to determine the IoU.
- False Positive, False Negative, True Positive. If the class label of the predicted bounding box and the ground truth bounding box are the same and the IoU between them is larger than a threshold number, the prediction is considered to be true. We compute the following three metrics based on the IoU, threshold, and class labels of the ground truth and predicted bounding boxes.
  - True Positive: The model correctly (true) predicted the existence of a bounding box at a specific place.
  - False Positive: A bounding box was predicted by the model to exist at a specific location (positive), but it turned out to be untrue.
  - False Negative: A ground truth bounding box actually exists at that place, proving that the model's prediction of no bounding box at a certain position was false and negative.
  - True Negative: The model was accurate (true) and did not anticipate a bounding box (negative). This relates to the background, or the region devoid of bounding boxes, and is not taken into account when determining the final measurements.
- Precision, Recall. Based on the TP, FP, and FN, for each labeled class, we calculate two parameters: precision and recall.
  - Precision: demonstrates the accuracy of our model by indicating the proportion of real cats among all detected cats, for example. As a result, it is the ratio of true positives to all cat predictions (i.e., the model's total of true positives plus false positives).
  - Recall: Describes how well the model recalls classes from images, for example, how many cats the model was able to identify out of all the cats in the input image. As a result, it is the model's calculation of the ratio of real positives to all of the ground truth cats, which is comparable to adding true positives and false negatives.
- Average Precision. It might be challenging and subjective to choose a confidence value for your application. In order to break the dependence on choosing only one confidence threshold value, average precision, which is defined by The region beneath the PR curve represents average precision. The PR Curve is reduced by AP to a single scalar value. Across a range of confidence threshold values, the average precision is high when both precision and recall are high and low when either of them is low. The AP scale ranges from 0 to 1.
- Mean Average Precision. Each class's AP score can be determined. By averaging AP over all classes being taken into account, the mean average accuracy is determined.

II. METHOD

A. System Overview

As the flowchart in Fig. 3 shows, our approach begins with data collection consisting of:

- Pretrained model data from model zoo which is provided by Facebook [12]. We choose COCO-Detection/ faster_rcnn_R_101_FPN_3x because it has high accuracy box AP=42 with training speed = 0.286 second/iteration and referencing speed=0.051 second/image.
- Custom dataset for training objects which is not recognized using pretrained data model.
- Data from the test recordings from students in the form of front video recordings from laptop cameras, video recordings from the side via smartphones and ground truth data containing participant statements about the technique and timing of cheating carried out.

Furthermore, the recorded data and custom datasets are cleaned first and matched with ground truth before being used. The training process is carried out to detect objects that cannot be detected using a pretrained data model. Custom datasets are labeled and annotated and prepare the data needed for training. Custom datasets that are ready are then trained. The result is a new data model that will be used to detect objects on the test video footage along with the pretrained data model. The results of object detection are in the form of two, namely a new video containing object detection marks and a csv file containing time data and objects detected from each frame of the test recording video. The csv file will be compared with ground truth to get the detection performance value and provide information on the detected illegal objects.

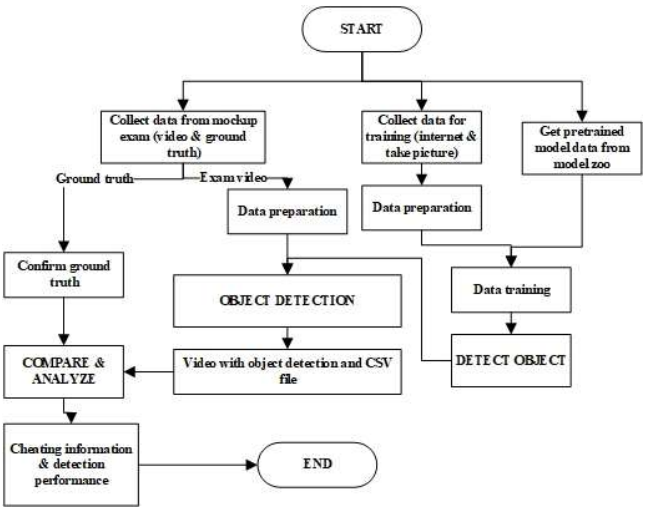

Figure 3. Flowchart system

III. RESULTS AND DISCUSSION

A. Collected Data

This stage aims to provide video data and cheating facts that will be used as test material and ground truth in testing the online exam cheating detection algorithm. This stage involves 15 D3 Informatics Engineering students with the following methods

- Announcement of volunteer registration with an appropriate replacement fee for each student with the obligation to carry out online exam activities recorded using laptop webcam cameras and smartphone cameras.
- The exam consists of 20 questions with a time of 40 minutes where students are required to perform at least 5 external cheating techniques during the exam. External fraud is fraud that is carried out outside of computer activities, for example: taking a smartphone, cheating and others. Students then write down the 5 cheating techniques on a form along with the time span they cheated on the video. One student's cheating data is shown in table II. This data will be used as ground truth for checking the accuracy of the algorithm.
- The recording is sent to the link provided along with a document explaining the time and technique used
- Complete data will be rewarded according to the promised funds

TABEL I. CHEATING RECORD FORM

Name :
Email :
Class :
Phone :

No	Time		Cheating technique
	Start	Finish	
1	01:20	01:33	Holding paper
2	01:38	02:02	Take handphone
3	02:10	02:30	Reading something on the left
4	02:32	02:49	Leaving exam room
5	03:02	03:37	Another person coming
6	03:45	04:03	Take handphone

Furthermore, the video recording of the exam is checked and processed by eliminating more frames at the beginning and end of the test and taking video samples with a shorter duration to make it easier to speed up testing. From the results of data processing, it was found that from 15 participants only 10 data could be used. This is because students edit the recorded data so that it cannot be used in the next process.

B. Data Training

Training is needed to detect objects that are not recognized by the pretrained data model, including mobile phones that are placed in several different positions, especially in front of the laptop screen and above the keyboard as shown in Figure 4. The pretrained data model can only recognize the cellphone when it is on a desk or table. hand.

Figure. 4. Cellphone image in front of monitor not detected by pretrained data model

For that it is necessary to get some sample images of mobile phones or tablets around the monitor or laptop. The process of collecting custom datasets is done by searching the internet and taking pictures using a camera for several different positions. The data collected were 60 images from the internet and 11 images from camera photos.All custom datasets are labeled and annotated according to the object to be recognized in the image. The objects to be trained are mobile phones, tablets, and pens. This custom dataset is then trained and produces a new dataset model. Finally, this new dataset model is evaluated to get the level of accuracy and performance. The results of the evaluation of the overall accuracy of the custom data model are shown in table II. The accuracy value of the pen is low because the training image data contains only 7 pens and overall, it is still low because the amount of data used for training is still small.

TABLE II. OVERALL TRAINING ACCURACY LEVEL

AP	AP50	AP75	APs	APm	AP
32.568	40.736	40.736	nan	22.723	37.046

This custom dataset model is then combined with the pretrained dataset that has been provided by Facebook Research [12] as the baseline dataset model. The zoo model provides a variety of models equipped with data on the accuracy and speed of testing for each data model.We can choose the desired model based on the desired priority. In this case we use the model COCO-Detection/faster_rcnn_R_101_FPN_3x/137851257/model_final_f6e8b1.pkl because it has a high speed of 0.286 and a high accuracy of 42.0.

C. Implementation, testing and system evaluation

The object detection process on the test video recording must be done twice, namely using pretrained datasets and custom datasets. Both data models succeeded in giving marks and labels to objects detected on the test recording video as shown in Figure 5.

Figure 5. Object detection results using pretrained data model (a) which is able to recognize many objects including the cellphone on the table. (b) Custom dataset is able to detect cellphones placed on laptops

The detection process also generates data in the form of a csv file containing frame number data, time and detected objects before exam began shown in table II and during the exam in table III. The csv file is then analyzed to mark frames suspected of fraud. csv file consisting of frame number, time, detected objects and their similarity level (not shown in table II and III). These detected objects can be more than one in one frame. For example, from table II, we can see that there are 4 objects detected by camera before the exam began, namely laptop, people, mice dan toys. During the exam, at sixth second, camera detected another object namely keyboard as shown in table III.

TABEL II. OBJECT DETECTED BEFORE EXAM

Time	Object Detected			
	1	2	3	4
00:00.0	laptop	person	mouse	toy
00:00.1	laptop	person	mouse	toy
00:00.1	person	laptop	mouse	toy
00:00.1	person	laptop	mouse	toy
00:00.2	laptop	person	mouse	toy
00:00.2	person	laptop	mouse	toy
00:00.2	laptop	mouse	person	toy
00:00.3	laptop	person	mouse	toy
00:00.3	mouse	person	laptop	toy
00:00.3	laptop	person	mouse	toy
00:00.4	laptop	person	mouse	toy
00:00.4	person	mouse	laptop	toy
00:00.4	person	laptop	mouse	toy
00:00.5	person	laptop	mouse	toy
00:00.5	person	laptop	mouse	toy
00:00.5	person	mouse	laptop	toy
00:00.6	person	laptop	mouse	toy

TABEL III. OBJECT DETECTED DURING EXAM

Time	Object Detected				
	1	2	3	4	5
00:06.0	person	mouse	laptop	toy	
00:06.0	person	laptop	mouse	toy	
00:06.0	person	mouse	laptop	toy	
00:06.1	person	mouse	laptop	toy	
00:06.1	person	mouse	laptop	toy	
00:06.1	person	laptop	mouse	toy	
00:06.2	person	laptop	mouse	toy	
00:06.2	person	mouse	laptop	toy	
00:06.2	person	mouse	laptop	toy	keyboard
00:06.3	person	laptop	mouse	toy	keyboard
00:06.3	person	laptop	mouse	toy	
00:06.4	person	laptop	mouse	toy	
00:06.4	person	laptop	mouse	toy	
00:06.4	person	laptop	mouse	toy	
00:06.5	person	laptop	mouse	toy	bird
00:06.5	person	laptop	mouse	toy	

The algorithm for marking the frames in the test video is shown by the flowchart in Figure 6. This algorithm compares the video during the exam with a video 3 minutes before the exam starts because in the early minutes, the supervisor will check the room and only allow certain items to be placed near the examinee. So that when there are different objects that appear when the exam starts, it is suspect that there is potential for cheating.

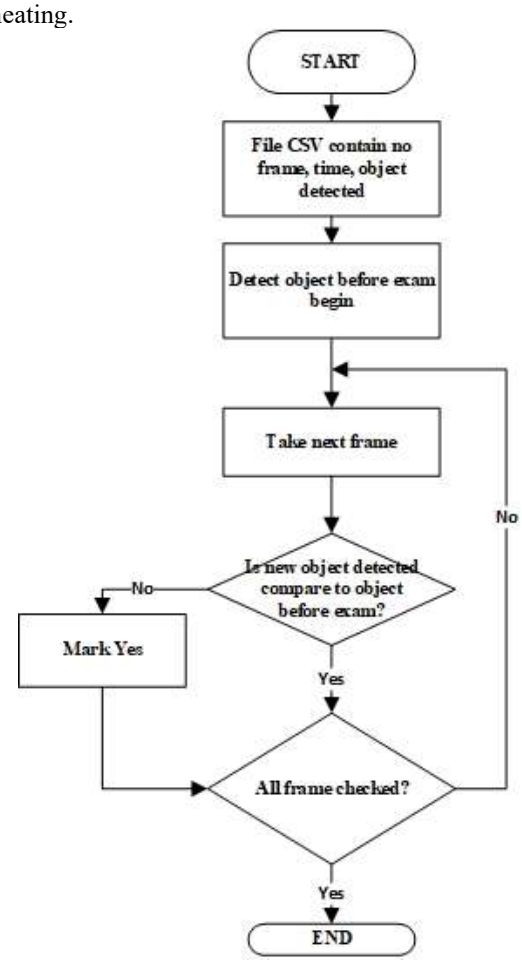

Figure 6. Fraud detection algorithm

Result of this algorithm shown in table IV where 'yes' means that object is detected before the exam. If the column in object detected say 'no', it can be signed as high potency of cheating at this time because new object appeared that it is not existed before the exam. For the purpose of accuracy measurement of the algorithm, we do manual checking the video at frame which marked as cheated. We marked it FP (False Positive) if the warning is false and TP (True Positive) if the warning is right. We also check statement from examinee about time where they did cheated action and compare it with the result of the algorithm. If by manual checking of the video, we detected a

cheating action but algorithm failed to detected it, then we marked it as FN (False Negative).

TABEL IV. OBJECT DETECTED DURING EXAM

Time	Object Detected					
	1	2	3	4	5	Manual check
00:06.0	yes	yes	yes	yes		TP
00:06.0	yes	yes	yes	yes		TP
00:06.0	yes	yes	yes	yes		TP
00:06.1	yes	yes	yes	yes		TP
00:06.1	yes	yes	yes	yes		TP
00:06.1	yes	yes	yes	yes		TP
00:06.2	yes	yes	yes	yes		TP
00:06.2	yes	yes	yes	yes		TP
00:06.2	yes	yes	yes	yes	no	FP
00:06.3	yes	yes	yes	yes	no	FP
00:06.3	yes	yes	yes	yes		TP
00:06.3	yes	yes	yes	yes		TP
00:06.4	yes	yes	yes	yes		TP
00:06.4	yes	yes	yes	yes		TP
00:06.4	yes	yes	yes	yes		TP
00:06.5	yes	yes	yes	yes	yes	TP
00:06.5	yes	yes	yes	yes		TP

The quality and quantity of training samples, the input image, the model parameters, and the accuracy threshold requirements all affect how accurate the object detection model is. Several parameters are used to measure object detection accuracy, namely:

- Precision is defined as the proportion of true positives to all positive predictions.

$$presisi = \frac{TP}{TP+FP} \quad (1)$$

- Recall is defined as the proportion of actual (relevant) objects to true positives.

$$presisi = \frac{TP}{TP+FN} \quad (2)$$

- True Positive (TP) — There is a warning, as anticipated by the model, and it is true.
- False Positive (FP) — The model incorrectly projected that there would be a warning.
- False Negative (FN) — The model incorrectly projected that there would be no warning.

After the data preparation process, only 6 videos can be used because the validation before the exam has failed so that all videos fail. The recapitulation of video processing results is shown in table V.

TABLE V. FRAUD DETECTION RESULTS

Video	Duration	Frame	Size MB	TP	FP	FN
Exam 1	4:22	7,889	556	4	0	2
Exam 2	20.54	37,355	906	3	3	0
Exam 3	18.41	28,034	1,009	3	1	0
Exam 4	9:04	14,183	1,004	5	1	0
Exam 5	15:16	27,356	1,290	6	0	1
Exam 6	22:17	78,112	3,520	2	0	0
Total	1:47:13	192.929	8,285	23	5	3

The accuracy obtained using equations 1 and 2 is a precision value of 0.884615385, which means that there are 11.5384615% of the algorithm giving a warning that there is fraud, but actually there is no cheating. While the recall value is equal to 17.8571429% of events where the algorithm does not give a warning when in fact there is fraud.

IV. CONCLUSION

Based on the discussion of the previous chapters, it can be concluded that the accuracy of detecting online exam fraud using a side camera is a precision value of 0.884615385, which means that there are 11.5384615% of the algorithm giving a warning that there is cheating, which in fact there is no cheating. While the recall value of 0.821428571 means that there are 17.8571429% events where the algorithm does not give a warning when in fact there is fraud.

ACKNOWLEDGEMENTS

This work is supported by Research Grant No. 2677.35/PL36/PG/2021 of the State Polytechnic of Banyuwangi, manage by LPPM State Polytechnic of Banyuwangi.

REFERENCES

[1] Haotian Li, Min Xu, Yong Wang, Huan Wei, Huamin Qu. 2021. A Visual Analytics Approach to Facilitate the Proctoring of Online Exams. 2021. arXiv:2101.07990v1 [cs.HC] 20 Jan 2021

[2] Darwin L. King and Carl J. Case. 2014. E-cheating: Incidence and trends among college students. Issues in Information Systems 15, 1 (2014), 20–27.

[3] Jeroen Janz, Community for Learning Innovation, https://www.erasmusmagazine.nl/ en/2020/09/10/eur-considers-using-second-camera-in-proctored-online-exams. Diakses pada 24 Maret 2021

[4] Chia Yuan Chuang, Scotty D. Craig, and John Femiani. 2017. Detecting probable cheating during online assessments based on time delay and head pose. Higher Education Research & Development 36, 6 (2017), 1123–1137. https://doi.org/10.1080/07294360.2017. 1303456

[5] Ahmad Khawaji, Fang Chen, Jianlong Zhou, and Nadine Marcus. 2014. Trust and cognitive load in the text-chat environment: the role of mouse movement. In the 26th Australian Computer-Human Interaction Conference on Designing Futures - the Future of Design, OZCHI '14, Sydney, New South Wales, Australia, December 2-5, 2014. ACM, New York, NY, USA, 324–327. https://doi.org/10.1145/ 2686612.2686661

[6] Darwin L. King and Carl J. Case. 2014. E-cheating: Incidence and trends among college students. Issues in Information Systems 15, 1 (2014), 20–27.

[7] Gennaro Costagliola, Vittorio Fuccella, Massimiliano Giordano, and Giuseppe Polese. 2009. Monitoring online tests through data visualization. IEEE Transactions on Knowledge and Data Engineering. 21, 6 (2009), 773–784. https://doi.org/10.1109/TKDE.2008.133

[8] Xuanchong Li, Kai-min Chang, Yueran Yuan, and Alexander Hauptmann. 2015. Massive open online proctor: Protecting the credibility of MOOCs certificates. In the 18th ACM Conference on Computer Supported Cooperative Work & Social Computing, CSCW 2015, Vancouver, BC, Canada. Association for Computing Machinery, New York, NY, USA, 1129–1137. https://doi.org/10.1145/ 2675133.2675245

[9] Gosia Migut, Dennis Koelma, Cees G. M. Snoek, and Natasa Brouwer. 2018. Cheat me not: Automated proctoring of digital exams on bring-your-own-device. In the 23rd Annual ACM Conference on Innovation and Technology in Computer Science Education, ITiCSE 2018, Larnaca, Cyprus. ACM, New York, NY, USA, 388. https://doi.org/10.1145/3197091.3205813

[10] Yousef Atoum, Liping Chen, Alex X. Liu, Stephen D. H. Hsu, and Xiaoming Liu. 2017. Automated online exam proctoring. IEEE Transactions on Multimedia 19, 7 (2017), 1609–1624. https://doi.org/10.1109/TMM.2017.2656064

[11] Shaoqing Pen, Kaiming He, Ross Grishick, and Jian Sun. Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks. arXiv:1506.01497v3 [cs.CV] 6 Jan 2016

[12] Facebook research. Detectron model zoo. https://github.com/facebookresearch/detectron2/blob/main/MODEL_ZOO.md, 2021

[13]Joseph Roth, Xiaoming Liu, and Dimitris N. Metaxas. 2014. On continuous user authentication via typing behavior. IEEE Transactions on Image Processing 23 (2014), 4611–4624.

[14] Joseph Roth, Xiaoming Liu, Arun Ross, and Dimitris N. Metaxas. 2015. Investigating the discriminative power of keystroke sound. IEEE Transactions on Information Forensics and Security 10, 2 (2015), 333–345. https://doi.org/10.1109/TIFS.2014.2374424

[15] Erica Southgate, Karen Blackmore, Stephanie Pieschl, Susan Grimes, Jessey McGuire, and Kate Smithers. 2019. Artificial intelligence and emerging technolo- gies in schools. https://docs.education.gov.au/system/files/doc/other/aiet_final_report_august_2019.pdf

[16] Z. Cao, G. Hidalgo, T. Simon, S. -E. Wei and Y. Sheikh, "OpenPose: Realtime Multi-Person 2D Pose Estimation Using Part Affinity Fields," in IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 43, no. 1, pp. 172-186, 1 Jan. 2021, doi: 10.1109/TPAMI.2019.2929257.