E-INVIGILATOR: CHEATING DETECTION BASED ON COMPUTER VISION

Kaitong Li, Danmeng Lin

Institute of Systems Science, National University of Singapore, Singapore 119615

ABSTRACT

Nowadays, with the increase in social competition and more examinations, some people always want to achieve the desired results through improper means. This behavior is cheating. In the traditional examination room, the invigilator is responsible for supervising the examination discipline and preventing cheating. However, due to factors such as too long examination time and a large number of candidates, the invigilator sometimes misses some cheating behaviors. So, in order to alleviate this problem, we have made a cheating smart detection project, called E-Invigilator. The intelligent detection system assists the invigilator by analyzing the video surveillance during the examination. The system uses YOLOv3 algorithm to identify students and their location, and detects student actions in this area. Use the Convolutional Neural Network model to divide student actions into three categories: no cheating, peeping and passing notes, and label cheating behaviors in different colors. When the system detects cheating cheating behaviors, it locates the corresponding student and issues a warning to the invigilator. Finally, the system provides an analysis of the examination room discipline, counting the number and frequency of peeping and passing notes in the video unit time. This intelligent cheating detection system not only improves the detection success rate of cheating behavior and reduces the probability of cheating by examinees, but also reduces the number and pressure of invigilators, which greatly helps improve the order of the examination room.

Index Terms— Exam invigilate, cheating detection, video processing, object detection, abnormal behavior detection, Convolutional Neural Network(CNN), YOLO, computer vision, deep learning

1. INTRODUCTION

Fairness and justice are two key things of examinations. Referring to traditional paper-based examinations, invigilators are the fundamental role who monitor examinees' behavior to prevent their cheating. To reduce the manpower cost and improve the integrity, for example, in China, some schools have conducted non-invigilation examinations. Because some people would say that luck can help us out of difficulties, lax invigilation may easily cause examinees' fluke minds to cheat.

In this project, we apply the computer vision techniques we have learnt to achieve the unmanned invigilation, to reduce the manpower cost. We build an efficient and user-friendly anti-cheating system, called E-invigilator. We mainly use object detection and abnormal behavior detection. We use the videos captured by the camera in the classroom as our input. The input video examples are shown in Figure 1. First, we detect every single one of the examinees. As everyone is sitting on their seats, we can only detect the upper body part – head, shoulders, and their hands. Then, based on the examinees we have detected, we capture what cheating behaviors are happening. So, abnormal behavior detection is applied in this section.

Fig. 1. Examination Room Surveillance Video

We define passing notes and peeping others as cheating behaviors. When the system detects cheating behaviors, it will be displayed on the results page. The output of the system is the labels of three cheating behaviors and cheating frequency statistics charts. The system circles the people who have cheating behaviors on the output video and marks the type of cheating. In addition to circling cheaters and marking cheating categories, the system also has another output. It will analyze the two types of cheating, passing notes and peeping at others respectively, and count the occurrences of these two behaviors per unit time. The frequency is displayed on the interface with a histogram, which is convenient for the invigilator to statistically analyze the discipline and order of the examination room, count the periods of high incidence of cheating by students and strengthen the invigilation measures accordingly.

Finally, we evaluate the accuracy of the system. By establishing a confusion matrix between the predicted and actual values of the three behaviors, the precision and recall rates of each frame in the video are calculated. Taking the average of the precision and recall rates of all processed frames in the video, we can get the overall precision and recall rates of the system. A good precision rate is the criterion for a successful system.

2. LITERATURE REVIEW

With the rise of Artificial Intelligence, cheat detection has received mass attention for the past few years. In this section, literature review is divided into the following two parts, based on the structure of this system.

2.1. Human Detection

As deep learning technology is rapidly developing in the recent years, some latest researches explored human detection in this method. Si[1] applied face detection to detect human in a large scale, which is also applicable to student detection in a classroom. Performances of three models was compared in this research - MTCNN, YOLO and AdaBoost algorithm. And at last conclusion that face detection based on MTCNN method achieved the highest accuracy. HOG feature was used in research[2] to do the feature extraction. In this research, the author came up with a head-shoulder model to detect examinees.

While traditional object detection methods mentioned above were applied in many of the researches, Hong[3] implemented the human detection based on deep learning methods using YOLOv2 network and proposed an improved YOLO-D network, achieving better robustness in target detection. To improve the speed and accuracy, Ju et al.[4] proposed an enhanced method based on YOLOV3 network, achieving better performance especially in small object detection. Zhao et al.[5] and Joseph et al.[6] has introduced YOLOv3 network that it uses batch normalization to to improve convergence and prevent overfitting and considers object detection as a regression problem.

2.2. Cheat Detection

As cheat detection is still an immature technology for these years, methods of implementation varies from researches to researches. Literature review of this section mainly covers the following three parts.

2.2.1. *Skin Detection*

Skin detection method was proposed many years ago and has been applied in many researches. Hong[3] conducted skin detection to locate the hands of examinees, by various mathematical morphology methods. While the detection result might be affected by the illumination and even the different skin color among the examinees, YCbCr color space was used in this research to separate the color and the illumination of a frame. The result of skin detection is shown in Figure 2, which is a figure from this research paper[3]. Similar to Hong's research[3], Li[7] applied skin detection method to do the counting of examinees' hands and arms, so as to detect abnormal behaviors. However, this method is not robust to occlusion which would be easily affected by the camera angle.

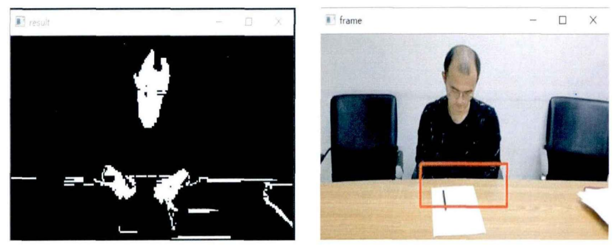

Fig. 2. Result of Skin Detection

2.2.2. *Posture Estimation*

Action detection has become popular with the rising of Artificial Intelligence. It can also be applied in cheat detection. Li[8] proposed a cheat detection method based on AlphaPose. After locating every single examinee, body posture estimation was applied to detect whether the examinee is cheating, by detecting the feature nodes of an upper body and then doing the linking, based on which posture changes and angle changes can be detected by frames. From the literature review, we summarized the strengths and weaknesses of posture estimation method. By doing posture estimation, every person in a frame can be identified and it is possible to perform simple tracking on each person in a video. And it is quite familiar with OpenPose we have experimented in the coursework assignments. While this method may miss some slight body moves and is not performing well in running speed that is not applicable to real-time video detection, it still need improving.

2.2.3. *Machine Learning Methods*

SVM method is very common in many abnormal behavior detection researches. In this research, Xiong[9] classified the dataset into eight types of body postures, then did the feature extraction to train the SVM model. This model is not robust to illumination and occlusion. Once the skin color of an examinee is similar to the background color, the accuracy of feature extraction would be greatly affected. Dai et al.[10] proposed an improved method based on latent SVM, which provides a more efficient method in training dataset mining. After collecting data, AnnotationV2.0 was used to process the dataset to generate training samples, for better model training.

Yu et al.[11] proposed a 3D Convolutional Neural Network method to detect cheat behaviors. This research considered the fact that student behaviors could be observed from two different perspectives using two independent cameras which would simultaneously analyze the data for designing a novel video feature extractor to extract the behavior characteristics of the students in two perspectives separately. While this method requires high complexity and specific devices, it is not applicable to this E-Invigilator system. For a better model, Cheng[11] provides an idea of how to do the image classification. InceptionV3 was used in this research to automatically extract features and then do the classification. Christian[13] has mentioned in his research that the computational cost of Inception is also much lower than VGGNet or its higher performing successors, which makes it more flexible to train the image dataset. In Md et al.'s research[14], they proposed an approach to recognize cooking state using Inception architecture, which could be a reference for this project to do the cheating behavior detection.

3. DATASET

Due to the uniqueness of cheating behavior compared with other human behaviors, very few corresponding data can be found in public data collections. As feature extraction should depend on a large scale of training samples, in order to achieve a better model training, we have established our own image dataset of cheating behaviors.

(a) Passing Notes (b) Peeing at Others

Fig. 3. Examples of Cheating Behavior Samples.

In this system, cheating behaviors containing obvious movement characteristics were classified into two categories: passing notes and peeping at others. These two cheating behaviors can basically cover most of the current cheating and have obvious differences with the students' normal behaviors, which are more conducive to learning and training. Examples of the cheating behavior samples in the image dataset is shown in Figure 3. Besides these two types of behaviors, images of examinees' not cheating were also included in the dataset as the third category. Each of these categories of the dataset contains 150 images for model training.

We also collected eight surveillance videos of the examination room for model testing and input, each is about twenty minutes' long. The frame rate of these video is 24 fps. Frame width is 1920 pixels and frame height is 1080 pixels. While video processing requires both high time cost and RAM cost, several one-minute video cuts were used in the system implementation and testing. As the illumination and contrast ratio will affect the model performance, data pre-processing methods including Histogram Equalization were applied in this system before model training. A screenshot of the surveillance videos is shown in Figure 4.

Fig. 4. A Screenshot of Surveillance Videos

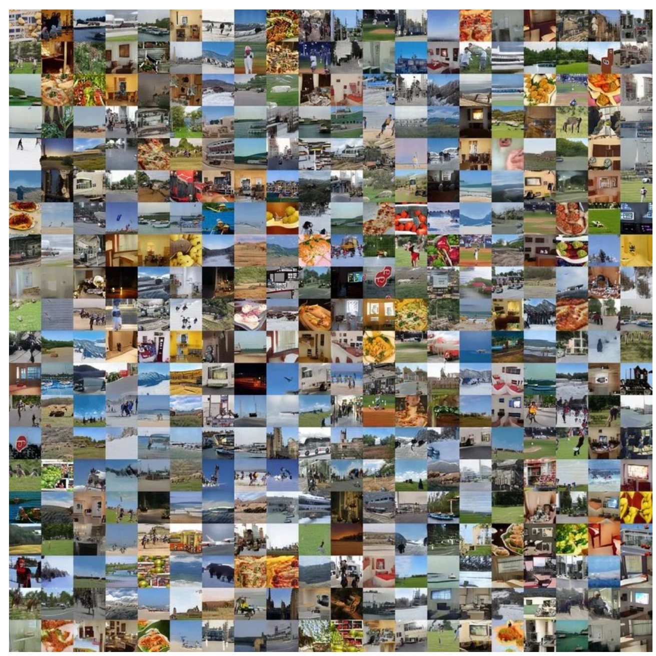

Fig. 5. Examples of COCO Dataset Samples

In object detection module, we used the pre-trained YOLO object detector which was trained on COCO dataset. It is a large-scale object detection, segmentation, and captioning dataset, containing images of 80 object categories. Examples of samples in this dataset are shown in Figure 5.

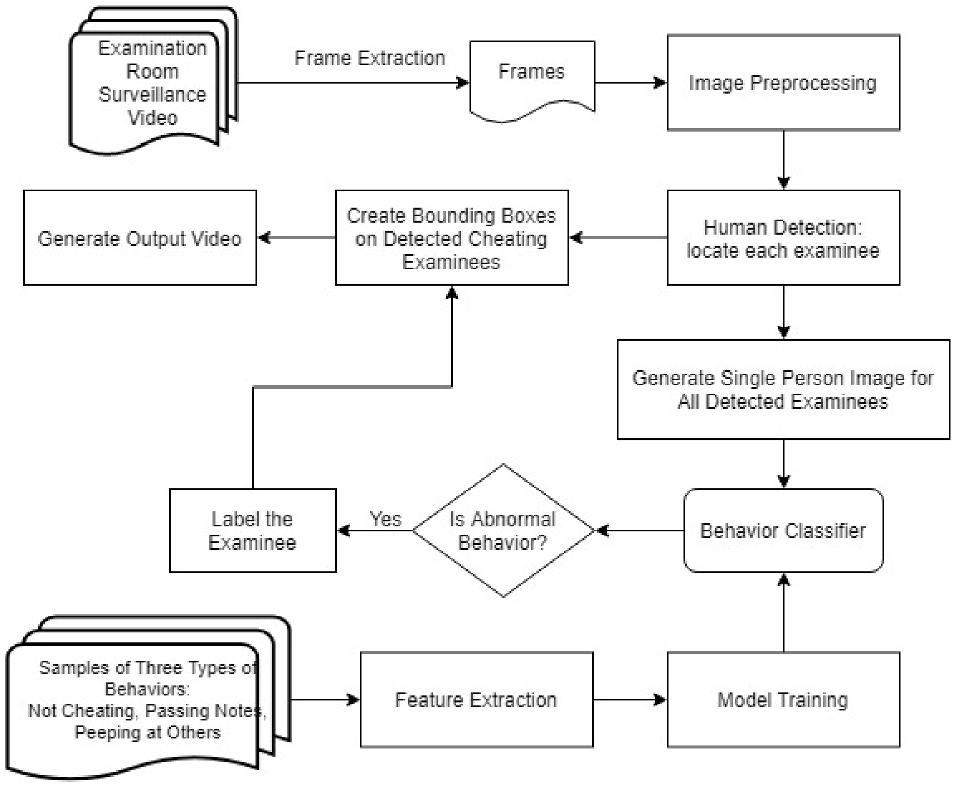

Fig. 6. Flowchart of System Structure

4. PROPOSED SYSTEM

After literature review, we decided to build a web version intelligent system with HTML and Python respectively as the front-end and back-end programming languages. Detailed model structure and how we implemented the system will be described below.

4.1. System Structure

Structure of this system is shown in Figure 6. Input of our system is an examination surveillance video. Output is a processed video inside which bounding boxes are covered on all the examinees detected cheating by frames. And what types of cheating behaviors are detected would be labelled on the bounding boxes.

In this project, as shown in the flowchart, first step of video processing was to extract all the frames and store them into an image array. Human detection would be conducted frame by frame. As illumination and noises are the factors that would affect the performance of object detection, image pre-processing should be done on each raw frame before inputting into the model. Human detection was done by using YOLOv3 network, in which every examinee would be located. Based on the location captured, this system would generate single person images as the input to the cheat behavior detection model.

Cheat behavior detection was applied on the basis of a single person. Thus the input would be single person images cropped from the frame. Then classify the behavior by using Convolutional Neural Network to generate a label. Next step was to create a bounding box labelled with the detected cheating behavior on this detected examinee. At last, integrate all the processed frame into an output video.

While processing a frame would take a certain period of time depending on the computer configuration, processing the video frame by frame would be pretty time-consuming. And it is unrealistic to define cheating behaviors on a frame basis. Instead, usually, a specific cheating behavior is done within more than ten frames or even several seconds. For these two reasons, in the system implementation, we chose to process the input video by every 16 frames which is more reasonable.

4.2. Proposed Approach

In this project, approaches we have used mainly cover the following two sections.

4.2.1. Human Detection

During the whole experiment, we have tried two methods to do the human detection. The first one was to the detection by

Table 1. Inceptionv3 CNN Model Summary

Layer(type)	Output Shape	Param #
input_1 (InputLayer)	(None, 299, 299, 3)	0
conv2d (Conv2D)	(None, 149, 149, 32)	864
batch_normalization (BatchNormalization)	(None, 149, 149, 32)	96
activation (Activation)	(None, 149, 149, 32)	32
conv2d_1 (Conv2D)	(None, 147, 147, 32)	9216
batch_normalization_2 (BatchNormalization)	(None, 147, 147, 32)	96
activation_2 (Activation)	(None, 147, 147, 64)	0
max_pooling2d (MaxPooling2D)	(None, 73, 73, 64)	0
conv2d_3 (Conv2D)	(None, 73, 73, 80)	5120
......	......	......
average_pooling2d (AveragePooling)	(None, 35, 35, 192)	0
conv2d_5 (Conv2D)	(None, 35, 35, 64)	12288
conv2d_7 (Conv2D)	(None, 35, 35, 64)	76800
......	......	......
mixed0 (Concatenate)	(None, 35, 35, 256)	0
conv2d_15 (Conv2D)	(None, 35, 35, 64)	16384
......	......	......
mixed10 (Concatenate)	(None, 8, 8, 2048)	0
GlobalPooling (GlobalAveragePooling)	(None, 2048)	0
Dense1 (Dense)	(None, 128)	262272
Dropout1 (Dropout)	(None, 128)	0
Dense2 (Dense)	(None, 128)	16512
Dropout2 (Dropout)	(None, 128)	0
output (Dense)	(None, 3)	387

posture estimation, using OpenPose which is a multi-person technology to jointly detect nodes of human body. But it took a very long process to do the estimation on a video and it was quite unstable. So we switched to the second method - using YOLOv3 network.

(a) Before NMS (b) After NMS

Fig. 7. Result of Before NMS and After NMS.

In the human detection module, we used a pre-trained YOLOv3 network which was trained on COCO dataset to do the detection. YOLOv3 method is one of the most widely used deep learning based object detection methods, which is doing object detection in a regression way. YOLOv3 network generates the probabilities and offsets of bounding boxes directly from the whole image frame with a single feed convolutional neural network.

After frame extraction on the input video, feed the single frame into the model. We set a threshold of 0.5 to the confidence score after doing a set of experiments to filter out weak predictions by ensuring the probability of detection is greater than the minimum probability.Then scale the top-left coordinates of each bounding box as the output of YOLO model was the coordinates of the center points and the size of each bounding boxes including width and height. Based on the information obtained from the model, we can determine the top-left corner of a bounding box. From the output of the YOLOv3 network, we only selected the category of ”person” and filtered out other unnecessary bounding boxes.

Next step was to apply Non-Maximum Suppression algorithm to filter some proposals and suppress weak and overlapping bounding boxes. After several experiments, we chose 0.2 as the IOU threshold. Comparison of the result before applying NMS and after applying NMS is shown in Figure 7.

4.2.2. Cheat Detection

In this project, two types of cheating behaviors were defined - passing notes and peeping at others. Behaviors aside from these two cheating behaviors were all considered normal, which were labelled ”not cheating” in our image dataset. Passing notes behaviors have the most distinguishing characteristics, as examinees usually perform the most body moving range.

To classifier cheating behaviors, InceptionV3 Convolutional Neural Network was applied in the model with 0.2 as the rate of validation split. From the system structure as shown in Figure 8, input shape of the model is (299,299,3) and the output shape is (8,8,2048). Detailed model summary is shown in Table 1. We trained our dataset using Incepetionv3 Convolutional Neural Network, with in total 360 images in the training dataset.

After getting the output from the Inceptionv3 network, to make the feature maps easily interpreted as categories confidence maps, global average pooling was applied to generate one feature map for each corresponding category of the classification task instead of adding fully-connected layers on the top of the feature maps. Take the average of each feature map. And then the output vector is directly fed into the Softmax layer.

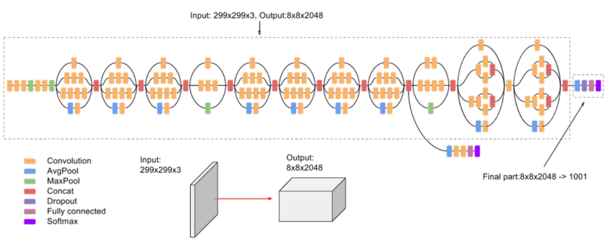

Fig. 8. Structure of Inceptionv3 CNN

Before feeding the images into the model, images would be resized to (299,299). And then do the format transformation from OpenCV format to PIL Image format as the output of image cropping is in OpenCV format while the input of the cheat detection model is in PIL image array format. This process should also refer to the change in color space - from BGR to RGB.

As sometimes it was ambiguous to distinguish the peeping behaviors with normal behaviors, we have tried to enlarge the data size of ”not cheating” category by adding more samples. But it did not go better for the result so we remained the previous version of dataset.

5. EXPERIMENTAL RESULTS

5.1. Human Detection

We have tried the posture estimation method using OpenPose on our examination surveillance videos. While it took very long to process a video and the detection was quite unstable, it was tougher to process cheat detection. As shown in Figure 9 and Figure 10, for the left side two girls, it is not accurate to link their arms together. Besides, as examinees were all at the desk, what were captured by the CCTV camera were their upper-body parts. Thus sometimes this model could miss their hands. Considering all these reasons, we abandoned this methods.

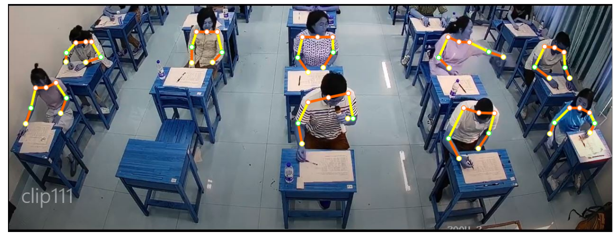

Fig. 9. Result of Posture Estimation (1)

Fig. 10. Result of Posture Estimation (2)

The confidence score threshold and IOU threshold are two key parameters in human detection using YOLOv3 network. After couples of experiments, we chose 0.5 as the confidence score threshold and 0.2 as the IOU threshold. However, it was still hard ensure detecting all the examinees in a frame as barrel distortion would happen because of the camera. As shown in Figure 11, barrel distortion is a type of deviation from rectilinear projection. So for examinees sitting right at the corner of the frame, they sometimes would be missed by the detection model. An example of this issue is shown in Figure 12.Besides, YOLOv3 network has the weakness in small object detection. So it still needs improving.

5.2. Cheat Detection

We use precision rate and recall rate to experiment with our system. We choose this two parameters because they are the main parameters for evaluating system availability. The precision rate is for the prediction result. It indicates how many of the samples whose predictions are positive are truly positive samples. The recall rate is for the original sample, and it indicates how many positive examples in the sample are predicted correctly.

For this system, we classify the sample into three categories, so it is a multi-classification problem. Three categories of peeping, passing notes and no cheating each has its own precision rate and recall rate, as shown in the green column on the right side of Figure 17. As shown in the chart, P represent peeping behaviors, PN represents passing notes

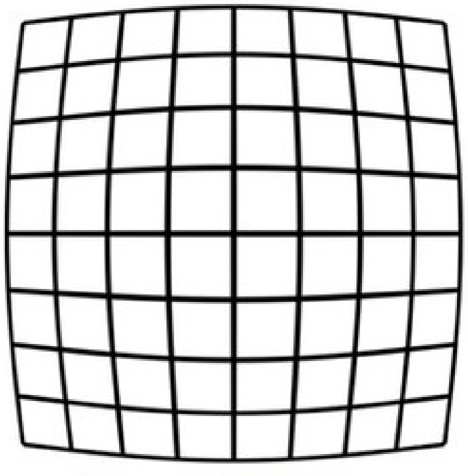

Fig. 11. Barrel Distortion

Fig. 12. An Issue of YOLOv3 Network Caused by Barrel Distortion

behaviors and NC represents not cheating. This system processes an video by every 16 frame. So the first column in 17 stands for serial number of each 16 frames.

We use a 40-second test video to evaluate the experimental results. An example of the screenshot of the video is shown in Figure 13. And Figure 14 shows an example of a processed frame. As shown in this figure, three examinees are detected peeping at others and four examinees are detected passing notes. One of the passing notes detected is an erroneous detection - the man who was using the phone should have been classified to ”no cheating” according to the rules of this system. Figure 16 shows the statistics result analysis of peeping in test videos. Figure 15 shows the statistics of peeping and passing notes made by the system for test videos.

The precision rate is the ratio of people cheating detected to the actual cheating. The recall rate is the proportion of people who did not cheat but was detected cheating, and those who cheated but were not detected cheating, to people who were detected cheating.

Fig. 13. A Frame of Image in the Test Video

Fig. 14. An Example of the Processed Frame

We calculate multi-classification problem by using the formula 1 to formula 6.The parameters in the formula are defined according to Table 2.

$$Precision(peeping) = \frac{a}{a+d+g} \tag{1}$$

$$Recall(peeping) = \frac{a}{a+b+c} \tag{2}$$

$$Precision(passingnotes) = \frac{b}{b+e+h} \tag{3}$$

$$Recall(passingnotes) = \frac{d}{d+e+f} \tag{4}$$

$$Precision(nocheating) = \frac{c}{c+f+i} \tag{5}$$

$$Recall(nocheating) = \frac{g}{g+h+i} \tag{6}$$

We took 60 frames of images from the test video, performed confusion matrix statistics on these 60 images, and calculated the precision rate and recall rates, as shown in Figure 17. Table 3 is an example of applying the formula to the first image, and table 4 is the sum of each parameter after we apply the formula to the 60 images in the test video.

Table 2. Confusion Matrix of Multi-classification Problem

Confusion Matrix 1		Predicted		
		peeping	passing notes	no cheating
Actual	peeping	a	b	c
	passing notes	d	e	f
	no cheating	g	h	i

Table 3. Confusion Matrix of the First Frame

Confusion Matrix 1		Predicted		
		peeping	passing notes	no cheating
Actual	peeping	2	0	1
	passing notes	0	2	0
	no cheating	0	2	20

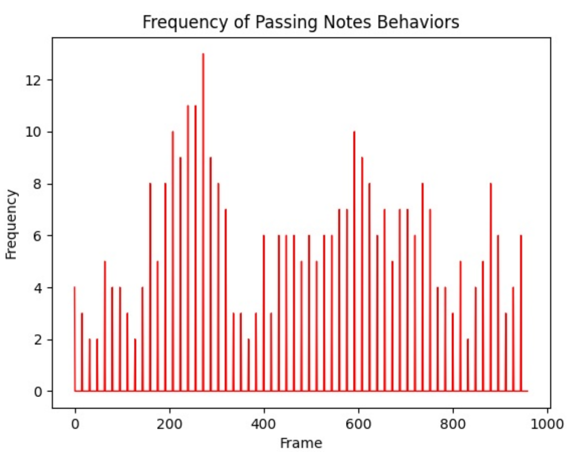

Fig. 15. Statistics Analysis of Passing Notes in Test Video

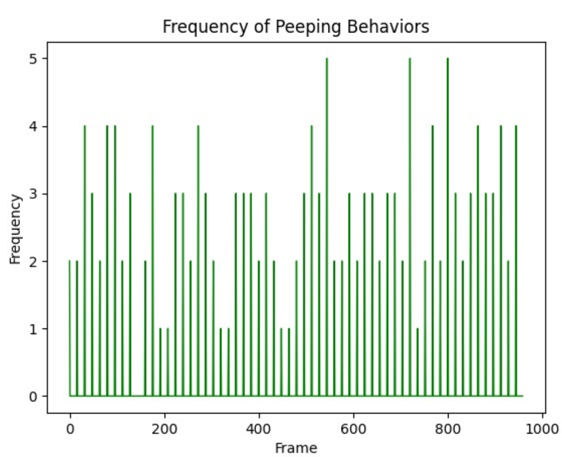

Fig. 16. Statistics Analysis of Peeping in Test Video

As can be seen from the results,the system has a high accuracy rate for the passing notes behaviors, because the feature of extending the arms are so obvious. But the difference between peeping and no cheating is limited to the direction of the head, so these two types are sometimes confused as one category. That is the direction we will continue to improve in the future.

6. CONCLUSIONS AND FUTURE WORK

6.1. Conclusion

In this project, we developed an intelligent system where we can do the cheat detection based on the examination room surveillance video captured by the CCTV cameras. It was a process during which we were exploring knowledge and techniques of computer vision. This project also gave us an insight on how to do the video analysis which we had never done before. Thus it was a challenge for us at the early stage when we decided what we were going to do and how we were going to implement it.

After going through the internet and some related researches, it is clear to see that cheat detection technology is still at an immature stage. And it is quite hard to find a complete cheat detection system in the market. So this project is not only a discovery but also an innovation although still have many aspects to improve.

6.2. Future work

For future work, if we have more time, more team members, or more computational resources, we would collect more data to improve the model, increasing the accuracy of the model. Furthermore, we can apply the system to more scenarios, such as detecting whether students are serious in class, etc.

Detailed future work is shown below.

1) Improve the object detection model, to make it more applicable to small object detection. And improve the

Table 4. Confusion Matrix of All Frames

Confusion Matrix 60		Predicted		
		peeping	passing notes	no cheating
Actual	peeping	104	29	90
	passing notes	12	253	51
	no cheating	48	63	970

robustness to various situation, such as frames in low contrast and frames with people occusion.

2) Improve the cheat detection model. Especially for peeping behaviors, more samples with distinguishing characteristics should be added into the dataset and improve the feature extraction, to make it more precise in distinguishing peeping behaviors and normal behaviors.

3) Conduct cheat detection on more detailed categories, such as using phones, talking to others, and etc. And even we can detect who is passing the notes and who is receiving the notes. Label all the examinees and then evaluate whether this examination is in good order.

7. CONTRIBUTIONS

Detailed task allocation to our team members in this project is shown below.

1) Li Kaitong is responsible for the integration of data sets, the establishment of object detection and cheating classification models, the realization of the combination between the front-end web page in HTML language and the back-end system implemented in Python language, and the improvement of web content.

2) Lin Danmeng is responsible for collecting examination room surveillance video, building up the frame of the web page user interface, pre-processing the dataset and making statistical analysis of data.

8. REFERENCES

[1] Si B.M., Research on Face Detection in Large Scale, Donghua University, 2020.

[2] Ding M.M., Research on Intelligent Monitoring Method Based on Examination Surveillance Video, University of Science and Technology of China, 2018.

[3] Hong Z., Research and Implementation of Detection Method for Abnormal Behavior of Examination Based on Video Surveillance, Beijing University of Posts and Telecommunications, 2019.

[4] Ju M.R., Luo H.B., Wang Z.B., et al., Improved YOLOV3 Algorithm and Its Application in Small Target Detection, Acta Optica Sinica, 2019, 39(7):1-8.

[5] Zhao L.Q., Li S.Y., Object Detection Algorithm Based on Improved YOLOv3, Electronics, 2020, 9:1-11.

[6] Joseph R, Ali F., YOLOv3: An Incremental Improvement, Computer Vision and Pattern Recognition, 2018.

[7] Li H., Research of Video-Based Abnormal Behavior Recognition in Exam, Tianjin University of Finance and Economics, 2012.

[8] Li M.X., Wang B.D., Dai W.B., et al., An Intelligent Cheat Detection Method in Standard Examination Room Environment, Intelligent Technology, 2020, 10:213-218.

[9] Xiong S. Research in Application of Abnormal Behavior Detection Method Based on SVM in E-Examination Room, Journal of Hubei University of Education, 2013, 30(8):42-46.

[10] Dai J.B., Long M.L., Zhao H.W., et al., Algorithm of the Exam Abnormal Behavior Detection, Journal of Jilin University (Engineering and Technology Edition), 2012, (42):236-240.

[11] Yu M.X., Jin X., Li X.D., et al., Abnormal Behavior Recognition Based on 3D Convolutional Neural Network with Dual Stream Videos, Journal of Beijing Electronic Science and Technology Institute, 2018, 26(4):60-71.

[12] Cheng W., Chen D.L., Lin H., et al., Pulmonary image classification based on Inception-v3 Transfer Learning Model, IEEE Access, 2017, PP(99):1-1.

[13] Christian S., Vincent V., Sergey I., et al., Rethinking the Inception Architecture for Computer Vision, Computer Vision and Pattern Recognition, 2015.

[14] Md S.S., Ahmad B.J., Rafsanjany K., Cooking State Recognition from Images Using Inception Architecture, Computer Vision and Pattern Recognition, 2019.

Actual -> Predicted	P -> P	P -> PN	P -> NC	PN -> P	PN -> PN	PN -> NC	NC -> P	NC -> PN	NC -> NC	Precision_P	Recall_P	Precision_PN	Recall_PN	Precision_NC	Recall_NC
1	2	0	1	0	2	0	0	2	20	1.00	0.67	0.50	1.00	0.95	0.91
2	1	0	0	0	2	0	1	1	22	0.50	1.00	0.67	1.00	1.00	0.92
3	2	0	1	0	1	0	2	1	20	0.50	0.67	0.50	1.00	0.95	0.87
4	2	0	1	0	2	0	1	0	21	0.67	0.67	1.00	1.00	0.95	0.95
5	2	0	0	0	3	0	0	2	20	1.00	1.00	0.60	1.00	1.00	0.91
6	4	0	2	0	2	1	0	2	16	1.00	0.67	0.50	0.67	0.84	0.89
7	2	0	1	0	2	0	1	2	19	0.67	0.67	0.50	1.00	0.95	0.86
8	1	0	1	0	2	1	1	1	20	0.50	0.50	0.67	0.67	0.91	0.91
9	2	0	2	0	1	0	1	1	20	0.67	0.50	0.50	1.00	0.91	0.91
10	1	1	2	0	3	0	0	0	20	1.00	0.25	0.75	1.00	0.91	1.00
11	2	1	0	0	7	1	0	0	16	1.00	0.67	0.88	0.88	0.94	1.00
12	2	1	1	0	4	1	2	0	16	0.50	0.50	0.80	0.80	0.89	0.89
13	1	2	1	0	6	1	0	0	16	1.00	0.25	0.75	0.86	0.89	1.00
14	1	1	1	0	8	2	0	1	13	1.00	0.33	0.80	0.80	0.81	0.93
15	2	2	0	0	7	3	1	0	12	0.67	0.50	0.78	0.70	0.80	0.92
16	2	2	1	0	9	0	1	0	12	0.67	0.40	0.82	1.00	0.92	0.92
17	1	1	2	0	9	0	1	1	12	0.50	0.25	0.82	1.00	0.86	0.86
18	2	2	0	1	10	0	1	1	10	0.50	0.50	0.77	0.91	1.00	0.83
19	2	1	1	0	6	1	1	2	13	0.67	0.50	0.67	0.86	0.87	0.81
20	1	0	1	0	6	0	1	2	16	0.50	0.50	0.75	1.00	0.94	0.84
21	1	1	0	0	5	0	0	1	19	1.00	0.50	0.71	1.00	1.00	0.95
22	1	1	3	0	2	0	0	0	20	1.00	0.20	0.67	1.00	0.87	1.00
23	2	1	2	0	1	0	1	1	19	0.67	0.40	0.33	1.00	0.90	0.90
24	2	1	1	0	1	0	1	0	21	0.67	0.50	0.50	1.00	0.95	0.95
25	2	0	1	0	2	1	1	1	19	0.67	0.67	0.67	0.67	0.90	0.90
26	2	0	1	0	5	0	0	1	18	1.00	0.67	0.83	1.00	0.95	0.95
27	1	0	1	0	2	1	2	1	19	0.33	0.50	0.67	0.67	0.90	0.86
28	1	1	1	0	4	1	1	1	17	0.50	0.33	0.67	0.80	0.89	0.89
29	1	0	2	0	4	0	0	2	18	1.00	0.33	0.67	1.00	0.90	0.90
30	1	1	3	0	3	0	0	2	17	1.00	0.20	0.50	1.00	0.85	0.89
31	0	0	2	0	4	1	2	1	17	0.00	0.00	0.80	0.80	0.85	0.85
32	0	1	3	1	2	1	2	2	15	0.00	0.00	0.40	0.50	0.79	0.79
33	2	0	3	1	5	1	1	0	14	0.50	0.40	1.00	0.71	0.78	0.93
34	2	0	3	1	5	1	0	1	14	0.67	0.40	0.83	0.71	0.78	0.93
35	1	1	2	2	6	2	2	0	11	0.20	0.25	0.86	0.60	0.73	0.85
36	1	1	3	1	6	2	0	0	13	0.50	0.20	0.86	0.67	0.72	1.00
37	2	1	2	0	6	3	0	0	13	1.00	0.40	0.86	0.67	0.72	1.00
38	1	0	2	1	10	2	1	0	10	0.33	0.33	1.00	0.77	0.71	0.91
39	2	0	1	0	8	2	0	0	14	1.00	0.67	1.00	0.80	0.82	1.00
40	2	0	2	0	4	1	1	4	13	0.67	0.50	0.50	0.80	0.81	0.72
41	1	0	2	1	4	2	1	2	14	0.33	0.33	0.67	0.57	0.78	0.82
42	2	0	1	0	6	1	0	1	16	1.00	0.67	0.86	0.86	0.89	0.94
43	0	0	1	2	4	2	1	1	16	0.00	0.00	0.80	0.50	0.84	0.89
44	1	0	2	0	7	0	2	0	15	0.33	0.33	1.00	1.00	0.88	0.88
45	2	1	1	0	5	0	0	1	17	1.00	0.50	0.71	1.00	0.94	0.94
46	4	0	0	0	5	2	1	1	14	0.80	1.00	0.83	0.71	0.88	0.88
47	1	1	0	0	5	2	0	2	16	1.00	0.50	0.63	0.71	0.89	0.89
48	1	1	1	0	4	2	1	2	15	0.50	0.33	0.57	0.67	0.83	0.83
49	3	0	2	0	3	1	1	1	16	0.75	0.60	0.75	0.75	0.84	0.89
50	2	0	3	0	2	1	0	2	17	1.00	0.40	0.50	0.67	0.81	0.89
51	4	1	0	0	1	0	1	1	19	0.80	0.80	0.33	1.00	1.00	0.90
52	3	0	3	0	3	0	0	2	16	1.00	0.50	0.60	1.00	0.84	0.89
53	0	0	3	0	2	2	2	0	18	0.00	0.00	1.00	0.50	0.78	0.90
54	2	0	2	1	3	1	0	1	17	0.67	0.50	0.75	0.60	0.85	0.94
55	3	0	2	0	4	0	0	1	17	1.00	0.60	0.80	1.00	0.89	0.94
56	3	1	2	0	6	1	2	1	11	0.60	0.50	0.75	0.86	0.79	0.79
57	2	0	1	0	4	1	1	2	16	0.67	0.67	0.67	0.80	0.89	0.84
58	4	0	1	0	2	2	3	1	14	0.57	0.80	0.67	0.50	0.82	0.78
59	1	0	3	0	2	0	1	2	18	0.50	0.25	0.50	1.00	0.86	0.86
60	3	0	3	0	4	1	1	2	13	0.75	0.50	0.67	0.80	0.76	0.81
	104	**29**	**90**	**12**	**253**	**51**	**48**	**63**	**970**	**0.63**	**0.47**	**0.73**	**0.80**	**0.87**	**0.90**

Fig. 17. Result of Cheating Data Analysis