Detecting Collectable Resources on Computer Game Based on the Neural Network

Qianhe Shen *

RDFZ Chaoyang Branch School, Beijing, China

* Corresponding Author Email: Shenqianhe@rdfzcygj.cn

Abstract. Image recognition and classification were put in used in many places in recent years thanks to the great progress in machine learning and neural networks. However, seldom evidence indicate applications of image recognition in computer games. Since games are gradually becoming part of peoples life, it is the time to put the focus on those who are unable to enjoy this entertainment. Nowadays computer games often include the element of collecting e.g. (materials or treasure). A program that could automatically mark those collectable objects would significantly improve the gaming experience for those who have disability in their eyes. In this work the game Destiny 2 was used for training and testing and training model chose was Yolov5. 1,500 of images were put into the data set and trained with 350 epochs. The model eventually achieved 0.82 in precision, 0.96 in recall and 0.91 in F1. It also gets the average mAP value of 0.93. These results can strongly prove that this model has the potential to be used in real scenarios.

Keywords: Neural Network, Machine Learning, Computer Vision, Computer Games.

1. Introduction

Image recognition based on the neural networks which is a prevailing used machine learning technique is gradually becoming one of the greatest aspects of artificial intelligence [1-3]. This technique identifies the content in the image by extracting relevant features of the object [4]. Due to its distinctive structure, which resembles the mathematical model of neurons, neural networks are able to perform a relatively accurate recognition when compared to the conventional method. [5]. To train the neural networks for image recognition, there are mainly two stages, namely forward stage and backward stage [4]. During the forward stage the machine basically compares the new data that contains a label with the original ones which is also called forward propagation. In the case of the backward stage the networks calculate the loss of the input image and adjust the weight of networks this process can be called as backward propagation.

As the explosive development of artificial intelligent in recent years, lots of dataset appears like MS COCO, PASCALVOC [6], and push the field of object detection into a new level which allows them to be used in real life application. Since object detection with neural networks brings a range of advantages, a variety of fields have used this techniques includes, transportation, catering and clothing [7]. Yet it is rarely seen object detection utilized in the field of entertainment, especially in computer game.

The neural networks for object detection actually have a huge potential in handling tasks in the field of computer games. Therefore, this research put the scenario in computer games for several reasons. Most computer games contain the element of collection. Players would have to spent much time searching and collecting the rare materials in game. Considering that the whole process is exhausting and frustrating, an object detection program that are able to trace the user's screen and mark the collection is capable to neutralize this problem to a great extent. With deploy of neural network, the computer can detect the object much faster than human eye. Last but not least, the importance of the application of this program to help the disables to have a better gaming experience cannot be ignored. Those who are color blind or unable to distinguish the figure can significantly be benefit from this.

In this regard, this study set up a method to indicate the collectable element in game on the user's screen. The model Yolov5 is used in this research for training the dataset. Compares to the previous

version of Yolov4, Yolov5 come with a much powerful performance in both accuracy and detection speed. Destiny2 is the game selected for further experiment because it includes a common system of collecting materials in game and each type of material have its own 3D model which allows the player to observe it in different angle. Moreover, each type of material has a clearly distinguishable and uniquely fashion in appearance which could represent most looking of materials in different games.

2. The Method

2.1. The preparation and preprocessing of the dataset

The dataset of this project is selected and processed form the video record in game. Videos of the collectable material was recorded in game in the first part of the collection of the dataset. The Video is then processed by the video editing software into specific video clips which included the frame for recognition. The clips were then grouped into three categories by the type of the material it contained. After that a further optimize which mainly involved three set was occurred. First of all, the video clips previously mentioned is input into a python program. which is modified to extract frames of the video clip 60 times per second to create a sequence of images which includes the subject for recognition. Secondly, the sequence of image was import to the image labeling website called makesense [8]. It was able to mark the position of materials with boxes and output a series of text file with the information of the labels. Lastly, the labels and the corresponding images are divided into groups of train set and validation set. In this project 1,031 of images were put into the train set and 441 in validation set. Figure 1 presents the sample images used in the study.

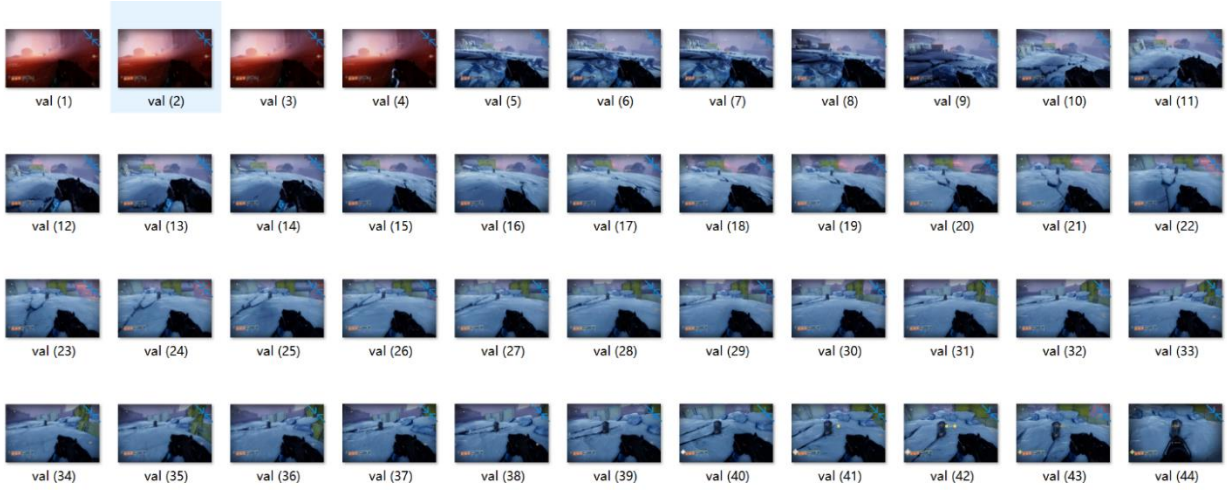


Figure 1. The sample images

2.2. Configuration of the environment to deploy Yolov5

BP neural network is back propagating, mainly composed of three parts: input layer, middle layer and output layer. The number of nodes in the input and output layers is relatively easy to determine, but the determination of the number of nodes in the hidden layer is a very important and complex problem.

After the creation of the dataset the environment was set for the Yolov5 to deploy. In this case the required packages were installed in anaconda and the IDE used is pycharm. Specifically, the version of the package pytorch selected was 11.3 which is match to the device that holds Nvidia GPU for running the project.

2.3. Introduction of Yolov5 and the model chose

The Yolov5 mainly consist of 4 parts: input, neck, backbone and head

input. In this part the model will process and enhance the input images. The model read 4 images each time after that it will randomly rotate, zoom the image and rearrange them together. After that all image will be process into a uniform resolution and input to the next part.

Neck. In this part the image was sliced by passing through serval convolution kernels. An image with default resolution 608×608×3 will eventually be processed into 304×304×32

Backbone. In this stage the model employs a structure of PAN [9]. This structure generally mixes and arrange the image features from previous layers in the network and pass some of the features to next layer.

Head. The last part is used for final detection, it generates the prediction categories and probability with detection box which enclose the object.

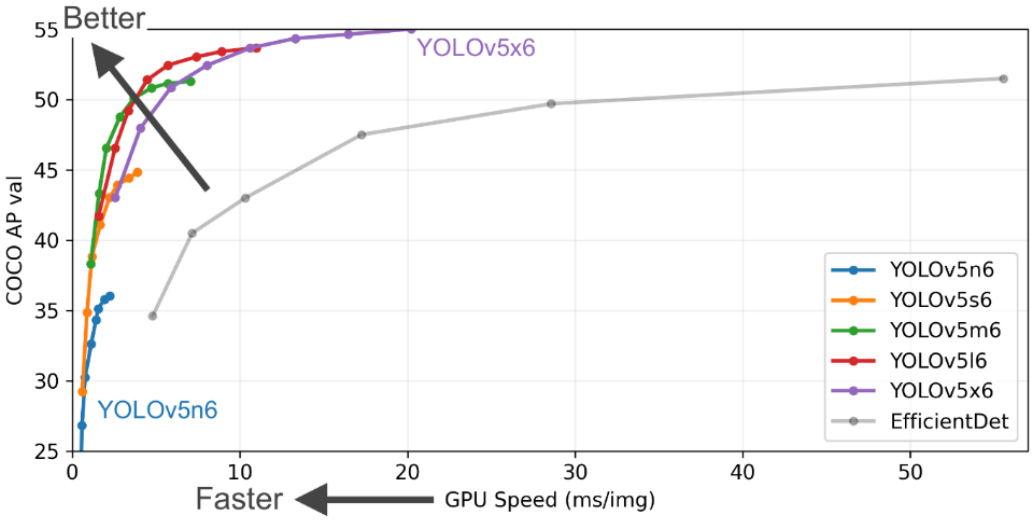


Figure 2. The performance of each model

The type of model selected for the project was yolov5x. It can be interpreted as a more accurate model compares to others from the figure above (Figure 2) [10].

2.4. Approach for training the model

In the previous part all 1,471 of images were made into the dataset. Later those images were formatted with the default resolution of 607×607 before they input to the networks. After that a few more step of adjustment was done with the model. An .yaml file was created to indicate the position of the dataset and connected with file of yolov5x. The file of yolov5x was also edited with 3 classes of object for detection in output.

In the training session the default weight file(yolov5x.pt) was selected for initial weight. The parameter of batch size and workers were optimized for best performance in training. Eventually the epoch of the training was set to 300 in total with the batch size of 6.

2.5. Evaluation standards

The evaluation standards which are commonly used are the accuracy (1), precision (2) and recall (3). The F1 score (4) can combine the value of both precision and recall, which could be used as the final evaluation of the model.

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

$$Precision = \frac{TP}{TP+FP} \quad (2)$$

$$Recall = Sensitivity = \frac{TP}{TP+FN} \quad (3)$$

$$F_1\ Score = \frac{2*Precision*Recall}{Precision+Recall} \quad (4)$$

3. Results and discussion

After the training of 450 epoch with the batch size of 6, the model was successfully trained. Meanwhile, the PR curve of the result was displayed down below.

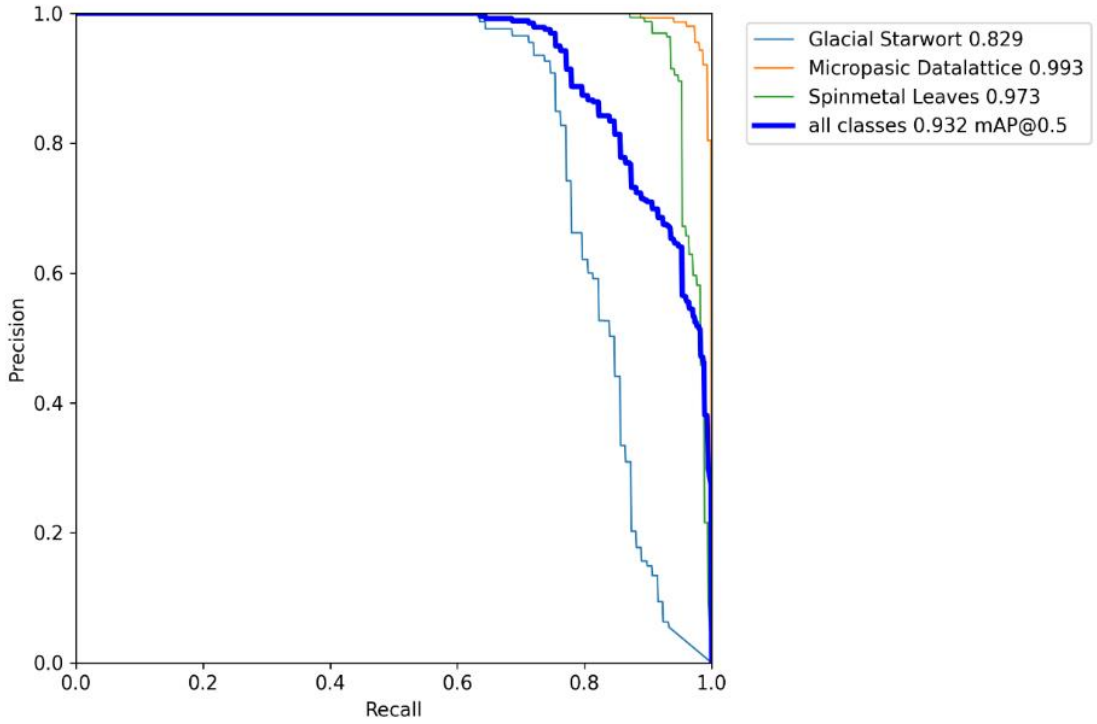


Figure 3. The precision and recall of the model.

This Figure 3 includes the PR curve of each individual materials (Glacial Starwort, Micropasic Datalattice, Spinmetal Leaves) indicated by different colors and the PR curve of all classes with the deep blue line. The area covered by the curve with x and y-axis is the mAP value. The graph also calculated this number and marked it on the right. It was obvious that the mAP of all class reached to an average of 0.93. Among all results, the detection of Micropasic Datalattice got the best outcome which was around 0.99 in mAP. Then follows the Spinmetal Leaves with 0.97 mAP which was approached to the previous one. The Glacial Starwort got the worst result (at 0.82 mAP). Several assumptions of the material that it had no obvious feature and its color merged with the background were made to demonstrate the unpleasant outcomes. On the other hand, the model reached its best mAP value with the average of all classes at 0.97 of the 105th epoch.

The Figure 4 below reveal the actual performance of the model in the real game play. The boxes were used to indicate the material that was detected. It marked the position of the object on the screen and distinguish them by different colors and the label on top of the box. The number besides the label showed the confidence of the model. In the practical test, above 70% of the time in total the model were able to recognized the correct material. However, detection in Micropasic Datalattice was not as capable as the other two, which was significantly different to the theoretical result in the training session. The images of Micropasic Datalattice in dataset which was not sophisticated enough or lack of quantity may be the main reason which give rise to the phenomenon of overfitting in this model.

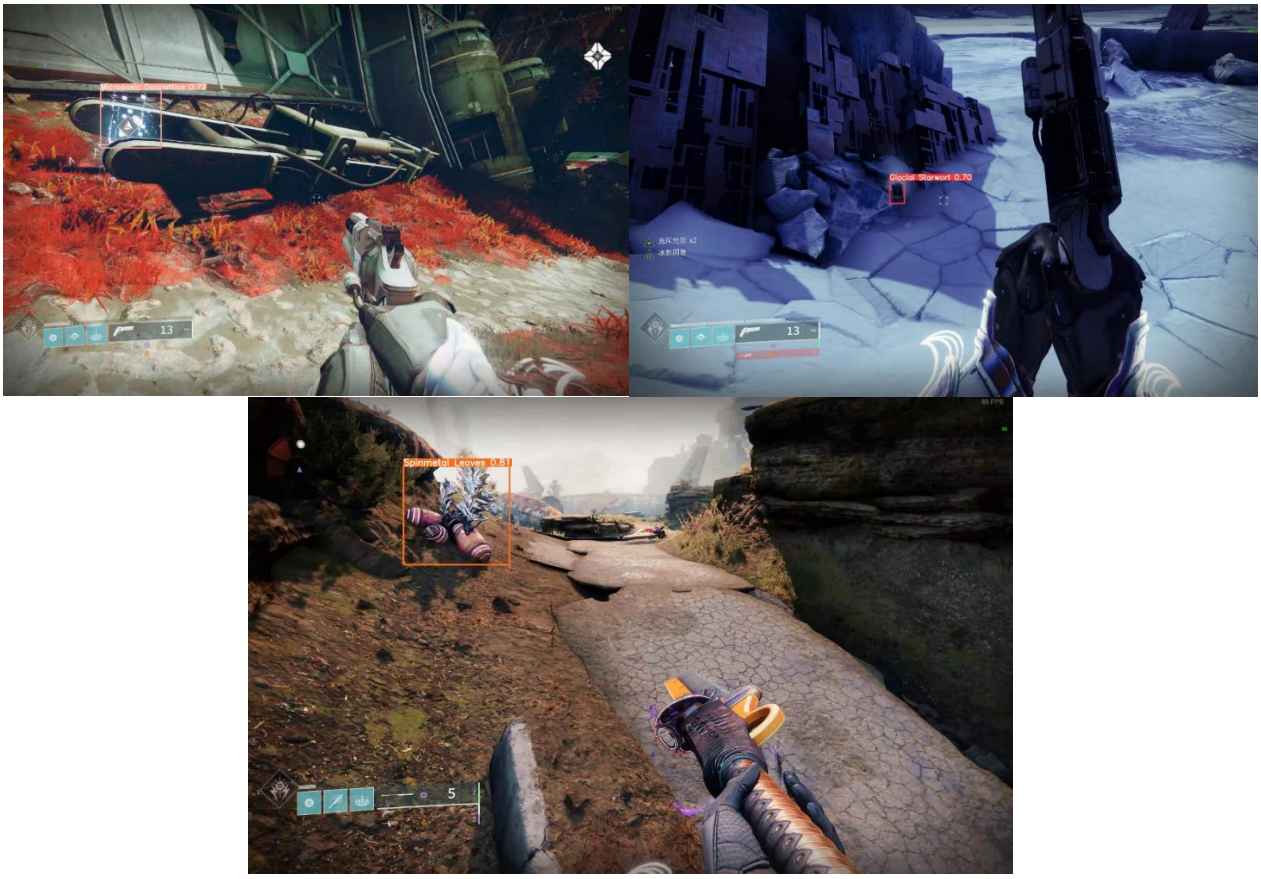


Figure 4. The detected result in the real game play.

4. Conclusions

In conclusion, this work proposed a way to achieve the material detection in games by using the neural network. In this work the model Yolov5x and datasets of the game Destiny2 were created to train the network. The result of this model reveals that its high accuracy where able to handle the task of recognizing game materials in most situations though the result may be affected by the appearance and the complexity of the material. Apart from this, several further improvements where proposed. Firstly, more images with greater variety in backgrounds and lighting could be put into the datasets to reduce the overfitting of this model. Moreover, the model currently includes 3 classes of material, which is not enough for most games. It is necessary to increase the categories of collectable materials. Lastly, an application interface could be deployed to replace the plain code for detection. This allows the process for detection to be simpler and more user-friendly.

References

[1] Tian Y et al. An improved YOLO Nano model for dorsal hand vein detection system [J]. Medical & Biological Engineering & Computing (2022): 1-13.

[2] Dixit A et al. Brain Tumor Detection Using Fine-Tuned YOLO Model with Transfer Learning [J]. Artificial Intelligence on Medical Data. Springer, Singapore, 2023. 363-371.

[3] Yu Q. Improved denoising autoencoder for maritime image denoising and semantic segmentation of USV [J]. China Communications, 17(3), 46-57, 2020.

[4] Traore B B et al. Deep convolution neural network for image recognition [J]. Ecological Informatics 48 (2018): 257-268.

[5] Abraham A, Artificial neural networks [J]. Handbook of measuring system design (2005).

[6] Wu W et al. Application of local fully Convolutional Neural Network combined with YOLO v5 algorithm in small target detection of remote sensing image [J]. PloS one 16.10 (2021): e0259283.

[7] Zhang Z, Research on the application of artificial intelligence in image recognition technology [C]. Journal of Physics: Conference Series. Vol. 1992. No. 3. IOP Publishing, 2021.

[8] Makesense [R]. 2022, https://www.makesense.ai/

[9] Liu S et al. Path aggregation network for instance segmentation [C]. Proceedings of the IEEE conference on computer vision and pattern recognition. 2018.

[10] Github. Yolov5 [R]. 2022, https://github.com/ultralytics/yolov5