Detection and Application of Apple Leaf Diseases Based on Object Detection

Mingyuan Zhong

School of Information Technology, Beijing Normal University, Zhuhai, Zhuhai, 519087, China

Abstract: *The healthy development of apple leaves is closely related to apple yield. There are three common symptoms of apple leaves: leaf spot, leaf rust and powdery mildew. Apple leaves are very susceptible to these viruses and are highly infectious. If the disease management of apple leaves is improper, it will lead to heavy losses in apple production and directly affect the economic income of farmers. Early detection of apple leaf diseases is the basis for timely prevention, which can inhibit the spread of diseases and minimize serious economic losses. Nowadays, the existing target detection technology can be used in apple leaf detection. Therefore, this paper will use the yolov5 algorithm based on deep learning to detect apple leaves and determine whether apple leaves are diseased. The experimental data comes from the Kaggle AI Challenge competition data set and the public data set on Kaggle. The disease area is manually marked by labeling tools to form a training apple leaf disease data set. In data processing, the Mosaic data enhancement method is used for data expansion, and the image is transferred into the HSV color space for tone transformation, so that the disease features have better highlights and achieve the purpose of data enhancement. The experimental model uses the YOLOv5 s network model. In order to improve the performance of the model, the CBAM attention mechanism is added to the backbone network of YOLOv5 s. The experimental data show that the accuracy (P) of the YOLOv5 s network model is 0.849, the recall rate (R) is 0.788, and the average detection accuracy (mAP) is 0.843. The accuracy (P) of YOLOv5s + CBAM was 0.869, the recall rate (R) was 0.798, and the average detection accuracy (mAP) was 0.848. At the end of the experiment, the web application design based on Flask framework is designed. In the last chapter, this paper looks forward to the future and summarizes the experiment.*

Keywords: YOLOv5; Object detection algorithm; Apple disease

1. Introduction

1.1 Background and Significance of the Research

Diseases of apple trees occur dispersed in different parts of the plant while those diseases are mainly reflected in their leaves. The most predominant apple leaf diseases include leaf rust, leaf spot, and powdery mildew[1]. Diseased leaves would undergo changes in color, streak, and shape. Moreover, different types of diseased leaves show different characteristics in their appearance[2]. Therefore, by precisely analyzing the characteristics of diseased apple leaves, it would be helpful for farmers to accurately grasp the growth condition of apple trees during apple planting through monitoring in real time and effectively preventing diseases of apple trees.

In the traditional process of crop cultivation, most farmers rely on their own ability to judge the cultivation of apples. However, people's ability to judge the disease correctly will gradually decline as their capability is limited. Meanwhile, due to some limitations such as working conditions, emotions, and weather conditions, it would be likely for people to make wrong judgments on the growth condition of apple leaves, leading to the spread of apple diseases. Once the virus spreads, the impact caused by the diseases of apple trees is irreversible[3]. Therefore, such a manual method is difficult for farmers to achieve a high yield of apples.

As an extremely important branch of deep learning, target detection plays a significant role in society development. In the identification of apple virus, target detection could perform well in identifying the specific location of apple disease and accurately analyzing the condition of apple leaves[4]. Therefore, it is an effective and important way to detect diseases in apple leaves by utilizing deep learning methods.

1.2 Research Tasks

In this thesis, we take the three viruses of apple leaf diseases-leaf spot, leaf rust, and powdery mildew-as the research object, and discuss the three steps of preprocessing apple figure data, building a model for detecting apple leaf diseases, and using the model to detect apple leaves.

(1) Construction of apple image data.

(2) Construction of training model.

(3) Design of application model.

2. Theories on Target Detection

2.1 Theories on Performance of Target Detection

2.1.1 Intersection over Union

The Intersection over Union (IoU) function calculates the ratio of the intersection and merger of two bounding boxes. In target detection, it is the ratio of the intersection and the union of the predicted candidate frame and the true frame of the target, and the calculation formula of IoU is shown in Figure 1. It is generally agreed that in a computer detection task, an IoU greater than 0.5 means a correct detection. In particular, if the predicted candidate frame and the true bounding box overlap perfectly, the IoU is 1 as the intersection is equal to the union.

Figure1. Calculation formula of IoU

2.1.2 Non Maximum Suppression Based Algorithm

Non Maximum Suppression (NMS) is a technique used in many computer vision tasks. During actual network training, it is likely to make multiple predictions for the same object, meaning that the algorithm may produce multiple prediction candidate frames for the same detected target, which would have a negative effect both on vision and prediction results. NMS algorithm is used to perform a proper filtration from those candidate frames and select the most appropriate one[5], as shown in Figures 2.
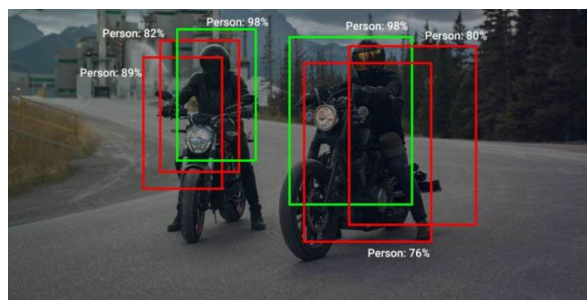

Figure 2. NMS algorithm

2.1.3 Performance Evaluation Metrics

The superiority or inferiority of the detection model should be evaluated by using specific data and scientific indicators to evaluate, so as to correctly reflect the performance of target detection algorithms[6]. Firstly, we introduce several metrics commonly used in the binary classification:

(1) True Positives (TP): The number of positive samples which are correctly predicted as positive samples.

(2) False Negatives (FN): The number of positive samples which are incorrectly predicted as negative samples.

(3) False Positives (FP): The number of negative samples which are incorrectly predicted as positive samples.

(4) True Negatives (TN): The number of negative samples which are correctly predicted as negative samples.

The following formulas are evaluation metrics commonly used in target detection, and some of the formulas would be used to evaluate the performance of experimental results:

(1) Precision (P): The fraction of TP among the samples predicted as positive ones by the model. The formula can be expressed as

$$P = \frac{TP}{TP + FP} \tag{1}$$

(2) Recall (R): The fraction of TP among positive samples, which could be described as

$$R = \frac{TP}{TP + FN} \tag{2}$$

(3) Precision-Recall (PR) curve: It shows the relationship between P and R. In general, the PR curve is plotted by setting Recall as the horizontal coordinate and Precision as the vertical coordinate. In target detection, higher values of P and R, mean a better performance of the model.

(4) Average Precision (AP): AP represents the average precision of a certain category of prediction, which is usually calculated as the area enclosed by the axes and the PR curve. It can also be calculated by finding out the precision rate. Assuming that there are m pictures in category x, then the average precision of this category x could be calculated as

$$P_{x,i} = \frac{TP}{TP + FP} \tag{3}$$

$$AP_x = \frac{1}{n} \sum_{i=1}^{m} P_{x,i} \tag{4}$$

(5) Mean Average Precision (mAP)[7]: the formula calculates the average AP of all the categories. The result is an important criterion to show the performance of the detection model. The formula could be expressed as:

$$mAP = \frac{1}{n} \sum_{i=1}^{n} AP_i \tag{5}$$

2.2 Introduction of YOLOv5

2.2.1 Architecture of YOLOv5 Network

The network architecture of YOLOv5 mainly consists of Focus module, CBL module, CSP module, SPP module, Concat module, and Upsample module. The network architecture of YOLOv5 is shown in Fig 3.

(1) Focus module: The internal structure of the Focus module is shown in Fig 4, which adopts the interlaced sampling splicing structure. Taking an input image with size 3*640*640 as an example, the image is first cut into four 3*320*320 images using the slicing operation, after which the images are spliced through the Concat module[8].
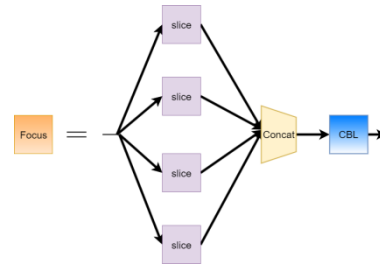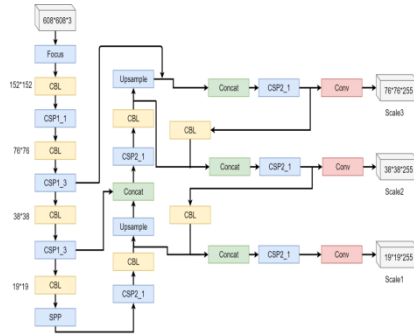
Figure 3. Architecture of YOLOv5 network   *Figure 4. Structure of Focus module*

(2) CBL module: CBL is a convolutional block and consists of three network layers, namely the convolution (Conv) operation, the batch normalization (BN) operation, and the activation function (Leaky-relu)[9]. The structure of the CBL module is shown in Fig 5.

Figure 5. Structure of CBL module

(3) CSP Module: CSP module consists of CBL module, Convolutional Layer (CONV), Residual Component, and Bottleneck Module. Two CSP architectures are designed in YOLO V5. A CSP1_X architecture is used in Backbone network and another CSP2 _X architecture is used in Neck. These two CSP architectures further enhance the integration ability of the network features[7]. The CSP module structure is shown in Fig 6 and Fig 7.
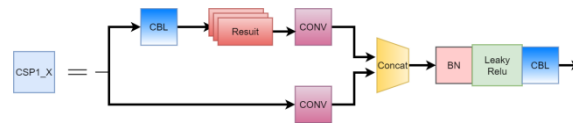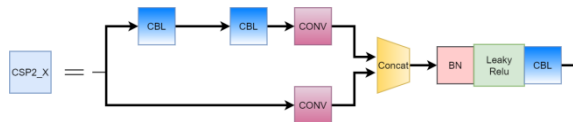

Figure 6. CPS1_X module structure

Figure 7. CSP2_X module structure

(4) Spatial Pyramid Pooling (SPP) module: SPP layer is mainly used to extract deeper image features while keeping the size of the input and output fixed[9]. The structure of the SPP module is shown in Fig 8.
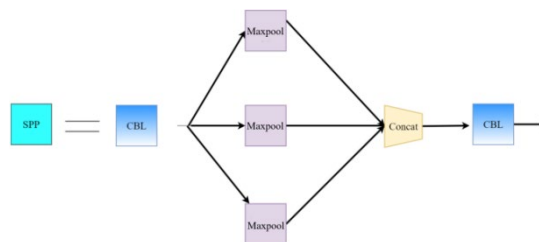

Figure 8. SPP module structure

2.2.2 Loss Function of YOLOv5

The loss function for target detection usually consists of two parts, namely bounding box regression loss and classification loss.

In the target detection task, YOLOv5 utilizes generalized intersection over union (GloU)[7] as the loss function for bounding box regression. GIoU is an improved method of IoU, which introduces the degree of completeness X of the predicted and real frames on the basis of IoU. In particular, the degree of completeness X is calculated by using the ratio of the intersection area of the smallest closed

rectangle constructed by the predicted frame and the real one to the corresponding union area. Let A be the predicted frame and B be the real frame. Denote C as the intersection area of the smallest closed rectangle constructed by the predicted frame and the actual frame, then the formula of GloU can be derived as

$$Glou = IoU - \frac{|C \setminus (A \cup B)|}{|C|} \tag{6}$$

Similarly, the loss function GloULoss could be described as

$$GlouLoss = 1 - GloU \tag{7}$$

The Binary cross-entropy loss function (BCELF) is used as the classification loss function for the training phase of YOLOv5 uses[9]. In summary, the complete loss function for YOLOv5 consists of the following three parts, namely bounding box regression loss, confidence prediction loss, and classification loss. The specific formula could be described as

$$Loss(obj) = GIoULoss + \sum_{i=0}^{S \times S} \sum_{j=0}^{B} 1_{ij}^{obj} [C_i \log(C_i) + (1 - C_i) \log(1 - C_i)] \tag{8}$$

$$- \sum_{i=0}^{S \times S} \sum_{j=0}^{B} 1_{ij}^{noobj} [C_i \log(C_i)$$

$$+ (1 - C_i) \log(1 - C_i)]$$

$$+ \sum_{i=0}^{S \times S} \sum_{j=0}^{B} 1_{ij}^{obj} \sum_{c \in classes} [p_i(c) \log(p_i(c))$$

$$+ (1 - p_i(c)) \log(1 - p_i(c))]$$

3. Apple Leaf Disease Detection Model Based on YOLOv5

3.1 Problem Analysis

Our experiments aim to accurately classify the type of apple leaf diseases and localize the diseased parts. First, the dataset in YOLO format needs to be drawn manually and preprocessed. Second, the data are put into a network model so that a YOLO based model would be trained to detect apple leaf diseases. Third, the test dataset is then put into the model for testing. The performance of the network model could be determined through metrics such as accuracy, mAP, and confusion matrix.

3.2 Experimental Dataset

The experimental dataset is obtained from the AI Challenge competition dataset and the public dataset on the website Kaggle, containing raw images of apple leaf diseases, and a data file recording the kind of apple leaf diseases corresponding to each image of apple leaf. Three common diseases are included in the dataset, namely leaf spot, leaf rust, and powdery mildew. Among the image dataset, 3023 images are selected as the training set, and 399 images are selected as the test set in proportion. The images are labeled manually by using image labeling tools and then the corresponding YOLO file data is generated.

The data for our experiment are labeled on the make-sense website, which is officially recommended by yolov5, and the labels labeled in more than 3,000 images totaled more than 12,000 labels.

3.2.1 One of the Categories in Dataset- Leaf Spot

Among apple leaf diseases, one of the diseases is named as leaf spot, also known as apple gray spot. The main cause of this disease is Venturia inaequalis parasitizing on the surface of apple tree leaves and fruit. Leaf spot would cause small, yellowish-brown spots on apple leaves about 1-5 mm in diameter, which would gradually increase in size and turn dark brown with tiny black spots visible in the center.

3.2.2 One of the Categories in Dataset- Leaf Rust

Another kind of common apple leaf disease is apple leaf rust, which is one of the common diseases occurring on apple leaves. The specific symptoms of this disease are the reddish brown, orange-yellow, or yellow-brown spots on apple leaves, which usually appear on the underside of the leaves first and then spread to the top of the leaves displaying round or irregularly shaped spots. The main symptoms of leaf rust are shown in Fig 9.

Figure 9(a) Leaf spot   Figure 9(b) Leaf Rust   Figure 9(c) Powdery mildew

3.2.3 One of the Categories in Dataset- Powdery Mildew

Another disease of apple leaves is named as powdery mildew. Such fungus would usually produce when the weather is humid and the temperature is favorable. This pathogen would enter the plant through plant stomata or wounds on the plant surface and casually forms mycelium on the plant tissues, which would lead to water and nutrient deficiencies in the corresponding area, and cause the leaves to become dead white[10].

3.3 Data Preprocessing

3.3.1 Mosaic Data Enhancement

Mosaic data enhancement would randomly select four different images and combine them together to form a new training sample after scaling and randomly cropping. The four images can be images of different categories or different images of the same category. The size and position of each image are also random[7]. Thus, the labeling information of these four images also needs to be combined accordingly, i.e., the object locations and categories should be relabeled on the new spliced images. For the current dataset in our experiments, after data enhancement, an image can be fused with the background information of four different images as well as different apple leaf spots. Moreover, more small spots would be added to the image after performing random scaling, which would be helpful in detecting diseases with small leaf spots[9].

3.3.2 HSV Color Space

HSV, standing for hue, saturation, and value, is a model to describe colors. In YOLOv5 algorithm, the detection of targets with specific colors could be achieved by using HSV color space. By converting the RGB images to HSV color space, the target with specific colors could be filtered out more easily. In addition, HSV color space in YOLOv5 could also deal with changes in lighting. Since the color information in the HSV model has a low correlation with lighting variations, the application of HSV color space in target detection could mitigate the effect of lighting variations on the detection results, thus improving the detection robustness[4].

3.4 Adding Multi-attention Mechanism Module

Attention mechanisms are one of the key techniques in neural networks for learning the relative importance of different parts of an input and modeling the relationships between different parts. It has been widely used in fields such as natural language processing and computer vision[11]. Currently, there are two types of attention mechanisms, namely channel attention mechanisms represented by squeeze and excitation (SE), and attention mechanisms based on convolutional block attention module (CBAM) by combining channel attention mechanism and spatial attention mechanism. CBAM module combines both spatial and channel attention mechanisms, allowing the network to understand specific content in the image more comprehensively and accurately. Thus, it has performed well in many computer vision tasks[9]. This approach improves the efficiency of the attention mechanism while maintaining accuracy. Therefore, the CBAM attention mechanism module would be added in our experiment. Fig 10 shows the schematic of CBAM implementation is shown. Fig 11 shows the channel attention module while
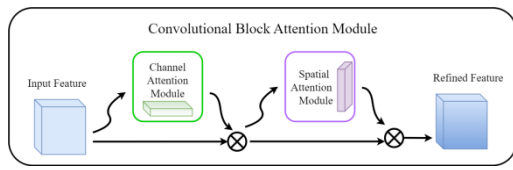
Fig 12 shows the spatial attention module.

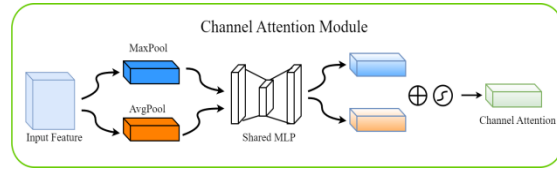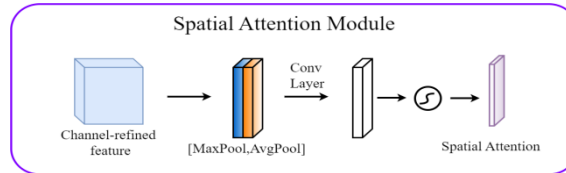Figure 10. Schematic of CBAM implementation   Figure 11. Channel attention module

Figure 12. Spatial attention module

3.5 Experiment and Results Analysis

3.5.1 Model Parameters and Experimental Environment

The network used in our experiment is yolov5s with a training epoch of 150 rounds and a batch size of 16 images. The environment for this experiment is as follows. The GPU we used is NVIDIA GeForce RTX 3080. The development language is Python 3.7. The GPU is accelerated by CUDA 11.4, and the software we used is PyCharm Community Edition 2021.3.1.

3.5.2 Analysis of Precision, Recall, and MAP

The results are shown in Table 1. In the training model based on YOLOv5s, the overall precision in the test set with a size of 399 images is 84.9% while the recall is 78.8%. The corresponding mAP is 84.3%. Among them, the precision of leaf rust is 84.2% while the recall is 83.9%. The corresponding mAP is 88.1%. Besides, the precision of the leaf spot is 88.7% while the recall is 79.1%. The corresponding mAP is 86.5%. Moreover, the precision of powdery mildew is 82.0% while the recall is 73.3%. The corresponding mAP was 78.3%.

After adding the CBAM module to the training model based on YOLOv5s, the corresponding results are demonstrated in Table 2. One could find that the overall precision is 86.9% while recall is 79.8%, and mAP is 84.8% on the same test set. Among them, the precision of the leaf rust is 89.7% while the recall rate is 84.8%. The corresponding mAP is 90.4%. The precision of leaf spot is 87.8% and the recall rate is 81.2%. The corresponding mAP is 87.7%. The precision of powdery mildew is 86.4% while the recall rate is 74.8%. The corresponding mAP is 74.8%. By observing the results, one could find that there is a significant improvement in the overall mAP.

Table 1. Experimental results of YOLOv5s

Types	Precision(P)	Recall(R)	mAP
Overall	84.9%	78.8%	84.3%
Leaf rust	84.2%	83.9%	88.1%
Leaf spot	88.7%	79.1%	86.5%
Powdery mildew	82.0%	73.3%	78.3%

Table 2. Experimental results of YOLOv5s+CBAM

Types	Precision(P)	Recall(R)	mAP
Overall	86.9%	79.8%	84.8%
Leaf rust	89.7%	84.8%	90.4%
Leaf spot	87.8%	81.2%	87.7%
Powdery mildew	83.2%	73.5%	76.1%

3.5.3 Analysis of PR Curve

The PR curve demonstrates the trade-off between precision and recall. The performance of the model could be judged by the highest point in the PR curve, i.e., the point at which the precision and recall are optimally balanced, which is generally in the upper right corner of the graph. The PR curve for YOLOv5s is shown in Fig 13, and the PR curve for YOLOv5s+CBAM is shown in Fig14 By

observing two figures, one could find that the highest point of the overall curve for YOLOv5s+CBAM is higher than that for YOLOv5s.
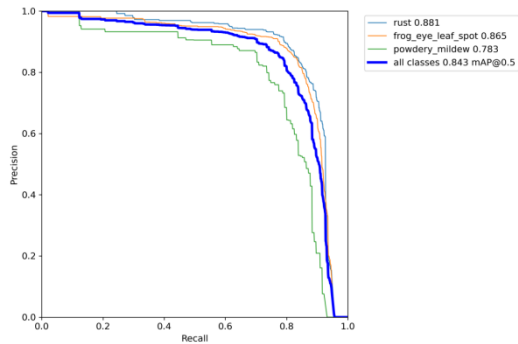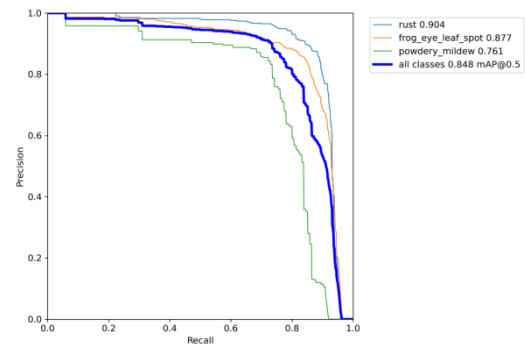

Figure 13. PR curve for YOLOv5s   Figure 14. PR curve for YOLOv5s+CBAM

3.5.4 Analysis of F1 Score

The F1 score is the harmonic average of precision and recall. A higher F1 score means better model performance. The F1 score of YOLOv5s is shown in Figure 15 and the F1 score of YOLOv5s+CBAM is shown in Figure 16. From the two figures, one could find that the F1 score of YOLOv5s+CBAM is higher than the F1 score of YOLOv5s under the same confidence level.
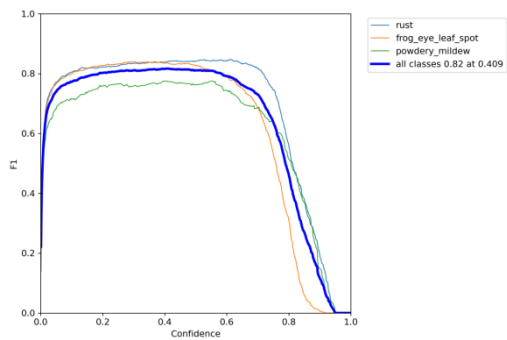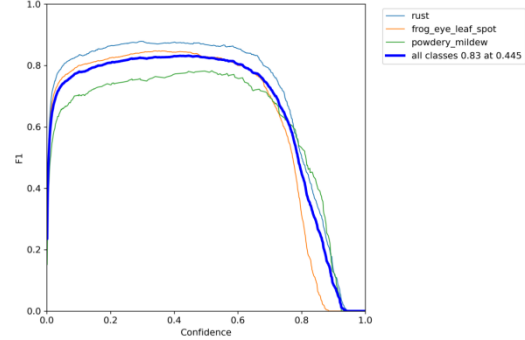

Figure 15. F1 score curve of YOLOv5s   Figure 16. F1 score curve of YOLOv5s+CBAM

3.5.5 Confusion Matrix

In our experiment, the confusion matrix of YOLOv5s is shown in Fig 17 and the confusion matrix after adding the attention mechanism is shown in 18. By observing the diagonal line, one could find that the classification accuracy would increase significantly after the attention mechanism is added. The accuracy of predicting leaf rust increases from 0.89 to 0.90 while the accuracy of predicting leaf spot increases from 0.85 to 0.87, and the accuracy of predicting powdery mildew increases from 0.77 to 0.79.(As shown in figure 18)
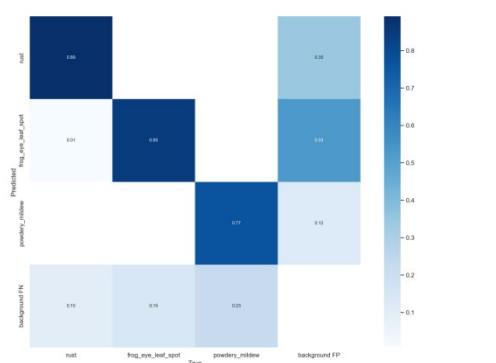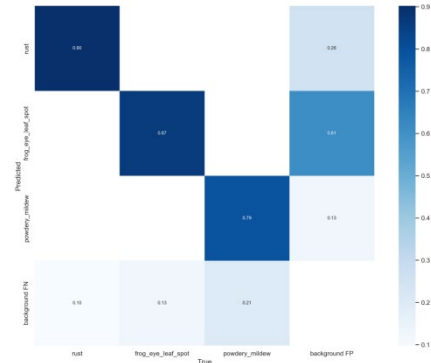

Figure 17. Confusion matrix of YOLOv5s   Figure 18. Confusion matrix of YOLOv5s+CBAM

4. Web Applications Based on Flask Framework

4.1 Introduction to Flask

Flask is a web framework based on Python and could be used to build web applications and APIs. The main purpose of the Flask framework is to build simple web applications quickly, which is ideal for small projects. Utilizing the Flask framework to display the detection results of YOLOv5 network on a web page could not only achieve quick access through a browser, but also achieve a visualization of the detection results of the YOLOv5 network. In this way, it would be easy for a user to better observe and understand the detection results as well as the specific purpose of the application. In addition, it would also be easy to interact with users through the web interface to enhance the user experience. Therefore, based on the above advantages, our experiment would use the Flask framework to display the apple leaf diseases detected by the YOLOv5s+CBAM network model with better performance metrics on a Web page.

4.2 Basic Architecture of the Project

Fig 19 shows the steps to display the detection results on a Web page using the Flask framework. First, a front-end page is programmed, in which the user can click the upload button to upload a picture of an apple leaf. After that, JavaScript code is invoked in the front-end page to send a POST request to send the uploaded image to an interface provided by the Flask framework. Through the interface, the YOLOv5s+CBAM model is used to make predictions and return the detection results to the front-end, which contains the object's category, the object's confidence level, and the object's coordinates. Finally, the detection results are visualized on the front-end page.
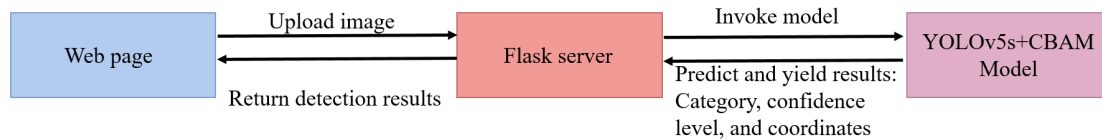

Figure 19. Project architecture

4.3 Web Page Display

The project consists of five pages, namely a home page, three introduction pages for the corresponding three diseases, and a page for detection, which is the most important one.

4.3.1 Home Page

The home page is used to provide guidance and could be divided into two parts. Fig 20 shows the first part of the home page, which contains a welcome font and a button that can be clicked to go to the image detection page. As shown in Fig 21, the second part of the home page is a brief introduction to apple leaf diseases. Furthermore, by clicking on Learn More, a detailed description of a particular apple leaf disease would be given.
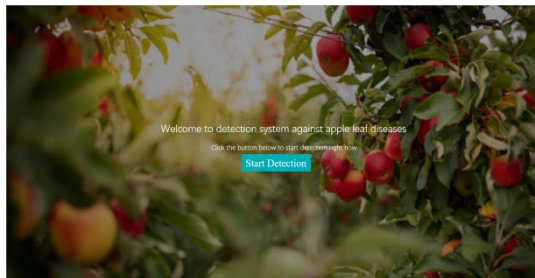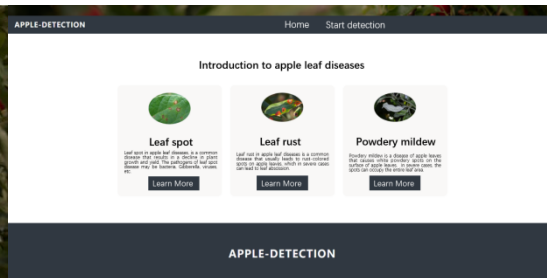

Figure 20. Web home page - part 1   *Figure 21. Web home page - part 2*

4.3.2 Introduction to Apple Leaf Diseases

The introduction pages provide detailed descriptions of the three common apple leaf diseases, namely leaf spot, leaf rust, and powdery mildew. These three diseases also correspond to the three categories in the dataset.

4.3.3 Detection Page

The detection pages are shown in Fig 22-23. The left part shows the image to be detected and the user can click on the upload image button to upload the image. The right part is the page to display the detection results, where the detected viruses are boxed out while the name of the virus and the confidence level is displayed as well. At the bottom of the page, the specific values of the detection result are also displayed, including the detection category, the confidence level, and the coordinates of the image.
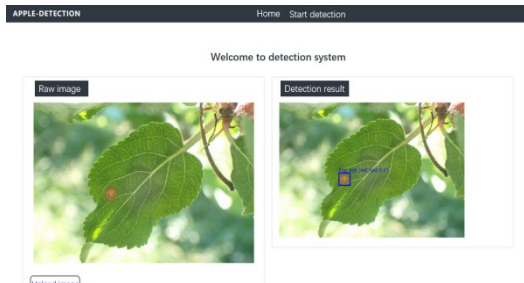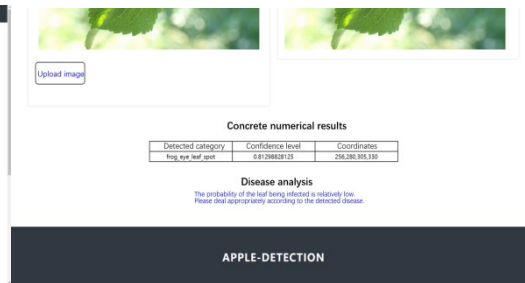

Figure 22. Detection page-part1   *Figure 23. Detection page-part2*

5. Conclusion and Future Work

In this thesis, focusing on the problem of apple leaf disease detection, we propose a detection algorithm based on YOLOv5, which uses artificial intelligence technology to automatically identify apple leaf diseases. Our main contributions are as follows:

Dataset Collection: The dataset comes from the apple leaf disease images on the AI Challenger competition on Kaggle and the public dataset on Kaggle. We manually label more than 3,000 images with more than 12,000 labels by using the YOLO officially recommended website, and generate the corresponding YOLO format file. The apple diseases are divided into three categories in specific, namely leaf spot disease, leaf rust disease, and powdery mildew disease.

Processing the dataset to improve the detection accuracy of the model: Mosaic data enhancement is used in processing the dataset to generate more data images for training. Meanwhile, the HSV color space is used to convert the RGB images so as to filter out the target objects of a specific color more conveniently. Hence, the detection accuracy of apple leaf diseases could be effectively improved.

Add attention mechanism: In order to correctly and efficiently detect apple leaf diseases, we add the CBAM attention mechanism to the backbone network of YOLOv5s. By observing the performance evaluation metrics and the confusion matrix, one can find that the precision, mAP, and other important metrics are improved after the attention mechanism is added.

Web program design: In order to visualize the experimental results, we design a Web page based on the Flask framework, where users could directly obtain the detection results after uploading the apple leaf images to the Web page. By displaying the Web page and the experimental results, it is demonstrated that the basic task of detecting apple leaf diseases is completed in this thesis.

References

[1] Liu Y. Research on apple leaf disease detection method based on improved Faster R-CNN[D]. Xi 'an: Shaanxi University of Science and Technology, 2022.

[2] Ding R. Research on apple leaf disease recognition and segmentation based on convolutional neural network[D]. Shandong Agricultural University, 2022.

[3] Xu C. Measurement of green total factor productivity and analysis of influencing factors in China's main apple producing areas[D]. Tai 'an: Shandong Agricultural University, 2022.

[4] Li J, Zhu X, Jia R, Liu B, Yu C. Apple-YOLO: A Novel Mobile Terminal Detector Based on YOLOv5 for Early Apple Leaf Diseases[C]//2022 IEEE 46th Annual Computers, Software, and Applications Conference (COMPSAC). Los Alamitos, CA, USA: IEEE, 2022: 352-361.

[5] Song X, Mariano V Y. Image-based Apple Disease Detection Based on Residual Neural Network and Transfer Learning[C]//2023 IEEE 3rd International Conference on Power, Electronics and Computer Applications (ICPECA). Shenyang, China: IEEE, 2023: 365-369.

[6] Firdous S, Akbar S, Hassan S A, Khalid A, Gull S. Deep Convolutional Neural Network-based Framework for Apple Leaves Disease Detection[C]//2023 4th International Conference on Advancements in Computational Sciences (ICACS). Lahore, Pakistan: IEEE, 2023: 1-6.

[7] Liu Y-C. Improved algorithm for target detection based on YOLO series[D]. Changchun: Jilin University, 2021.

[8] Bonkra A, Bhatt P K, Kaur A, Kamboj S. Scientific Landscape and the Road Ahead for Deep Learning: Apple Leaves Disease Detection[C]//2023 International Conference on Artificial Intelligence and Smart Communication (AISC). Greater Noida, India: IEEE, 2023: 869-873.

[9] Ding K, Li X, Guo W, Wu L. Improved object detection algorithm for drone-captured dataset based on yolov5[C]//2022 2nd International Conference on Consumer Electronics and Computer Engineering (ICCECE). Guangzhou, China: IEEE, 2022: 895-899.

[10] Chakraborty S, Paul S, Rahat-Uz-Zaman M. Prediction of Apple Leaf Diseases Using Multiclass Support Vector Machine[C]//2021 2nd International Conference on Robotics, Electrical and Signal Processing Techniques (ICREST). DHAKA, Bangladesh: IEEE, 2021: 147-151.

[11] B. K S B, G S, Kishore M, S R, A. D J. Real-time Pothole Detection using YOLOv5 Algorithm: A Feasible Approach for Intelligent Transportation Systems[C]//2023 Second International Conference on Electronics and Renewable Systems (ICEARS). Tuticorin, India: IEEE, 2023: 1678-1683.