

Application Research of Deep Learning in Crop Leaf Disease Detection

Ran Tao, Qianqian Li*, Yuhang Chen

School of Artificial Intelligence and Software, Kewen College, Jiangsu Normal University, Xuzhou, Jiangsu, China

**Corresponding Author*

Abstract: Crop leaf diseases have become a critical bottleneck in improving agricultural productivity and quality. Traditional manual and laboratory detection methods suffer from inefficiency, environmental and operator-dependent accuracy, and limitations in large-scale field applications. Integrating leaf disease detection with object recognition technology has emerged as a rapidly developing research direction for smart agriculture. The YOLO series of deep learning algorithms demonstrates significant advantages in target recognition. This study utilizes the YOLOv8 algorithm to develop a crop leaf disease identification model. Based on this model, we designed and implemented an intuitive system featuring login/registration, image recognition, and model switching functions. The system adopts a modular and hierarchical architecture, utilizes the PyQt framework for user-friendly interfaces, and employs the SQLite database for data storage. This innovation enhances the efficiency and accuracy of agricultural pest detection, providing robust support for modern farming practices.

Keywords: Deep Learning; Disease Recognition; YOLOv8; Crop Leaf; Recognition System

1. Introduction

Agriculture is the foundation of China's national economy and the root that supports economic and social development. It is related to food security, social stability and the overall development of the country. Leaf diseases seriously affect the growth and yield of crops. Currently, chemical control is the main control method, but there are problems such as excessive use of pesticides. Farmers are prone to misdiagnosis due to factors such as similar

symptoms of pests and diseases and their own lack of professional knowledge. The traditional way of seeking help from agricultural technicians or experts is also inefficient. The accuracy of identifying pests and diseases on crop leaves urgently needs to be improved [1]. With the rapid development of deep learning and image recognition technology, the combined application of deep learning and image recognition technology in the field of crop disease detection provides new research ideas and methods for the detection and identification of crop pests and diseases [2]. It is highly feasible to build an intelligent recognition system in combination with the development of smart agriculture.

This paper takes deep learning [3] as the core technology to construct a crop leaf disease recognition model [4]. Model training and back-end construction are carried out using the Python language and the PyTorch framework. Meanwhile, the front-end UI interface is designed with PySide6, and data is stored in combination with the SQLite database. The model is built through steps such as data preprocessing, model training and environment deployment, experimental analysis and system testing, and ultimately a crop leaf disease recognition system integrating user registration and login, image recognition and model switching functions is realized. It enables farmers to identify pests and diseases in a timely, efficient and convenient manner and apply targeted measures, as well as promptly carry out prevention and control of corresponding crop leaf diseases [5].

2. System Design

2.1 Overall System Function Design

The system functional module consists of three parts: the system management module, the image recognition module, and the model

management module, as shown in Figure 1. The system management module should be responsible for the initialization, configuration and monitoring of the entire system to ensure its stable operation. It should provide log recording and error management functions to enable quick location and repair when problems occur in the system. The image recognition module will serve as the core module of the system, using deep learning algorithms to recognize and analyze the uploaded images of crop leaves. Support processing of "recognized uploaded images", "recognized uploaded folders" and "recognized uploaded videos" to ensure it can handle various input forms. The module should internally include steps such as data preprocessing, model loading, feature extraction, and pest and disease identification to ensure a complete process from input to output. The model management module is responsible for the storage, management and update of deep learning models. Users will be able to specify the model used by the system through "Select Model File". User registration and login as well as user information management will provide user registration and login functions to ensure the security of the system and the privacy of user data. The user information management module is responsible for storing and managing users' personal information, historical records and other data, providing personalized services for users.

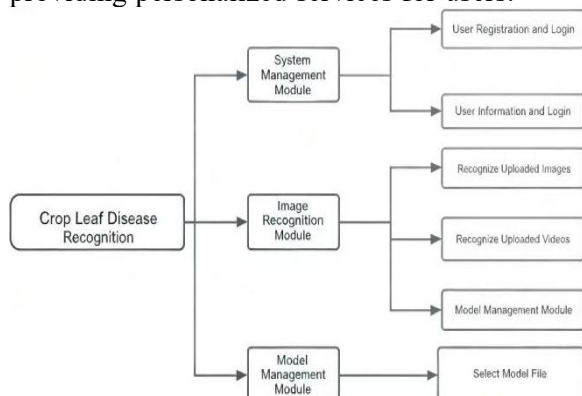


Figure 1. Overall Functional Design Diagram of the Crop Leaf Disease and Pest Identification System

2.1.1 System management module

The user management module mainly consists of two parts, namely registration and login, and user information management. Users who log in to the system for the first time can register according to the instructions. Registered users

can directly log in to the pest and disease identification system. The system will call the user information table in the database and match the information input by the front-end user with the corresponding information in the table. After a successful match and login, users can modify and maintain their personal information, such as changing their profile picture or password. As show in Figure 2.

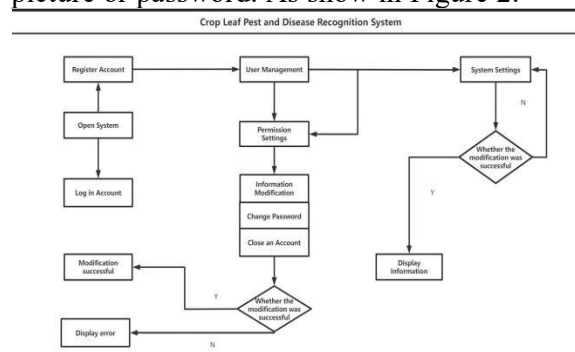


Figure 2. System Management Module Diagram

2.1.2 Image recognition module

Click "Start" in the system to identify and upload images of crop leaf diseases, and display the detection results on the page to feed back to the customers. The recognition module enables users to upload local image files and send them to the back end for processing. The images of crop leaf diseases are mainly identified by calling the improved YOLOv8 detection model in the experiment, and then the recognition results will be displayed on the page, as shown in Figure 3.

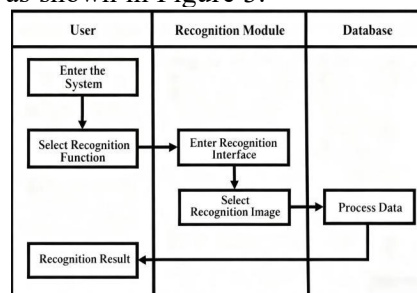


Figure 3. Diagram of the Image Recognition Module

2.1.3 Model management module

In the system, click on "Model Switch" and select the model file you need from the folder. After the selection is completed, the model switch is done. The switch can also be used to identify pests and diseases, as shown in Figure 4.

2.2 Database Design

According to the system requirements analysis,

the crop leaf pest and disease detection system designs three tables based on the SQLite database to store the tables needed during the system development and operation process, namely the user information table, the pest and disease detail table, and the identification information table. The user information table is used to store the personal information of registered users [6,7].

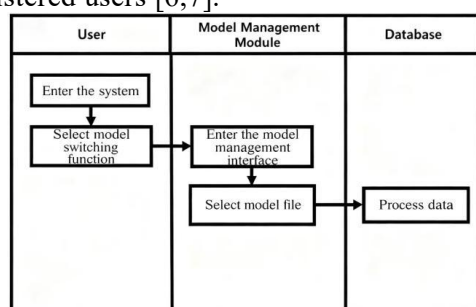


Figure 4. Diagram of the Model Management Module

Its relationship pattern is represented as (user id, user name, user password, user avatar), as shown in Table 1.

Table 1. User Information

Name	Type	MajorKey(Y/N)	Void(Y/N)
user_id	bigint	Y	N
user_name	varchar	N	N
password	varchar	N	N
user_image	varchar	N	Y

The pest and disease detail table is used to store relevant information about pest and disease details. Its relationship model representation (pest and disease id, pest and disease name) is shown in Table 2.

Table 2. Details of Pests and Diseases

Name	Type	MajorKey (Y/N)	Void (Y/N)
id	bigint	Y	N
disease_name	varchar	N	N

The identification information table is used to store relevant information during the system's identification process. Its relationship pattern is represented as (user id, user-uploaded image, recognition details, recognition time), and its table structure is shown in Table 3.

Table 3. Identification Information

Name	Type	MajorKey(Y/N)	Void(Y/N)
user_id	bigint	Y	N
updatepic	file	N	N
txt	text	N	N
date	datetime	N	N

3. System Implementation

3.1 Model Training

3.1.1 Data preprocessing

3.1.1.1 Dataset sources and division

At present, there are mainly two ways to obtain crop disease datasets: utilizing public datasets and self-collecting on-site. Due to objective limitations such as geographical conditions and professional knowledge, this paper only adopts the former. After research, open-source datasets in the agricultural field are relatively scarce. The existing public datasets include PlantVillage [8] (containing 38 categories and 54,303 images, involving crops such as peppers and tomatoes) and AI Challenger (containing 27 categories and 36,258 images). Involving crops such as apples and tomatoes, IP102 (including 102 types of pests, over 75,000 images, involving crops such as rice and wheat), etc. This paper selects the open-source dataset provided by the AI Studio (PaddlePaddle) platform, and screens out 15 common crops such as apples, sweet peppers, blueberries, corn, and potatoes, as well as related leaf diseases, to provide data support for the subsequent model training.

3.1.1.2 Dataset preprocessing

A total of 5,150 images of pests and diseases were selected from the above dataset and proportionally divided into a training set (4,420 images), a validation set (366 images), and a test set (364 images). The training set is used for the model to learn disease features and enhance recognition ability through iterative optimization. The validation set is used to adjust the model's hyperparameters and compare and analyze the performance of different parameter combinations to select the optimal configuration. The test set is used to evaluate the model's generalization ability on unknown data and ensure its robustness in practical applications. In the preprocessing stage, automatic direction correction is first carried out to eliminate the variations introduced by the differences in shooting angles, enabling the model to focus on learning the characteristics of diseases rather than direction noise. Secondly, all images were uniformly adjusted to a resolution of 800×800 pixels to ensure the consistency of input data and improve the processing efficiency of the model. Through statistical analysis of the dataset distribution map and the bounding box distribution map, two major problems were found: First, the number of labels varies significantly, which can easily lead to

overfitting of the model and a decline in generalization ability; Second, the disease labeling boxes are concentrated in the center of the image, reflecting deviations in data collection. To address these issues, data augmentation techniques such as image rotation, scaling, and color adjustment are adopted to increase sample diversity; At the same time, consider balancing the number of category samples through resampling techniques or adjusting the loss function to mitigate the impact of category imbalance. In addition, a refined annotation method is adopted in the dataset annotation. Precise bounding boxes are marked for each disease area and category labels are provided. The annotation quality is strictly controlled to ensure that the model can learn accurate and reliable disease identification features.

3.1.2 Performance comparison of YOLOv8

This paper evaluates and compares the performance of the YOLOv8, YOLOv7, and YOLOv5 models in the task of target detection of crop leaf diseases. To achieve this goal, control variables were used to train and test the three models using the same dataset respectively. The training parameters of the models were set, with no changes except for the training rounds (epochs). Three groups of experiments were set, and the epoch parameters were set to 100, 80, and 60 respectively. The experimental results will directly compare the performance of different models based on key indicators. The results are shown in Table 4.

According to the experimental results, the two evaluation metrics mAP and F1-Score of the YOLOv8 model generally lead the YOLOv7 model and the YOLOv5 model. The mAP score of the YOLOv8 model is generally higher than that of the YOLOv7 model and the YOLOv5 model. This indicates that it outperforms the YOLOv7 model and the YOLOv5 model in terms of overall detection accuracy. As an indicator of overall performance, a higher mAP score means that the model has better detection ability and higher reliability. Another important indicator is that the F1-Score of the YOLOv8 model is generally higher than that of the YOLOv7 model and the YOLOv5 model. This further confirms its advantages in detection performance. From the results of this experiment, it can be concluded that the

performance of YOLOv8 is superior to that of the YOLOv7 model and the YOLOv5 model in the target detection task of crop leaf diseases.

Table 4. Comparison of Model Experiment Results

epoch	Model name	mAP	F1-Score
100	YOLOv8	0.636	0.60
	YOLOv7	0.576	0.55
	YOLOv5	0.629	0.60
80	YOLOv8	0.611	0.59
	YOLOv7	0.552	0.52
	YOLOv5	0.633	0.60
60	YOLOv8	0.640	0.60
	YOLOv7	0.499	0.42
	YOLOv5	0.603	0.57

3.1.3 Model training and testing

YOLO models usually need to be deployed in a specific training environment. Before model training, the corresponding deep learning environment and the dependent libraries required by YOLOv8 should be installed and deployed properly. After the data preprocessing is completed, a dataset in YOLO format is obtained, and the environment required for model training is also deployed. The preparatory work for model training is basically completed. Place the dataset in the data folder for model training and modify the path information of the corresponding dataset in the model training code. Run the model training code and wait for the training to complete.

3.2 Function Implementation

3.2.1 System login module implementation

The system provides registration and login management functions based on SQLite. For first-time users, the system will guide them to complete the registration process through a dedicated registration interface. During this process, users need to enter their usernames and passwords, which will be securely stored in the SQLite database by the system. Once the registration is successful, users can easily log in to the system through the login interface with the username and password they set previously. This design enhances the security of the system, providing a strong guarantee for users' information security, laying a solid foundation for adding more personalized functions in the future, and reserving the possibility for the expansion of system functions.

3.2.2 Implementation of system image recognition function

On the main interface, the system supports multiple input methods, including images, videos, real-time cameras, and batch files. Users can easily select pictures or videos of plant leaf diseases to be detected by clicking the corresponding buttons, and even start the camera for real-time detection. When conducting disease detection, the system will immediately display the results and securely store these detection records in the database for subsequent review and analysis. This design not only enhances the convenience of user operation but also ensures the integrity and traceability of the detection data.

3.2.3 Implementation of system model switching function

This system also features the function of one-click switching of YOLOv8 models. Users only need to click the "Switch Model" button on the interface to easily select different versions of the YOLOv8 model for plant leaf disease detection. Meanwhile, the system also comes with a rich dataset, enabling users to retrain the model based on actual needs to adapt to the detection requirements in different scenarios, thereby ensuring the accuracy and reliability of the detection results.

4. System Testing

There are three functional tests to be conducted, namely: user login and registration function, pest and disease identification function, and model management function [9].

4.1 User Login and Registration Function

After starting the system, enter the user login interface. You need to input your username and password, click the login button to log in or press the Enter key to log in. If the user does not exist, it will prompt the user to register. Unregistered users click "Register" to be redirected to the user registration interface. Upload the user avatar, enter the username and password, and then enter the verification code. After all are successfully entered, click "Register". The user's registration is complete and they can be redirected to the user login interface to perform the user login operation. After logging out of the system and reopening it, you can still log in with your current username and password.

4.2 Pest and Disease Identification Function

After a user logs in successfully, they will enter the main page of the system. On the left side of the system, there are function ICONS for selection. Selecting image recognition will pop up the file selection interface. Select the image that needs to be recognized and upload it to the system. The system will detect the image, circle the detected area with boxes of different sizes, and mark the name and confidence level of the recognized label at the top. An image may detect multiple regions. The list at the bottom of the system's main interface will arrange all detected labels and their confidence levels in order of their confidence levels from high to low [10].

4.3 Model Management Function

On the main interface, you can select model management, and a file selection interface will pop up. Find the interface for storing model weights to select different versions of pre-trained models and test the detection effects of different models. After switching, the pest and disease identification function can still be used.

5. Summary and Outlook

This paper combines deep learning with crop disease identification and completes a crop leaf disease identification system based on a deep learning model. The main functions of the system include user registration and login, image recognition, model switching, etc., which are implemented through the PyQt framework, ensuring the friendliness of user interaction. In addition, the system's backend adopts the PyTorch deep learning framework and combines it with the YOLOv8 algorithm for model training and disease identification, demonstrating powerful performance. This system enables farmers to efficiently and conveniently identify pests and diseases, promptly prevent and control crop leaf diseases, reduce farmers' losses, and contribute to maintaining national food security, the stability of the agricultural ecosystem, and the sustainable development of agriculture.

Acknowledgements

This paper was supported by the "Jiangsu Provincial College Students' Innovation and Entrepreneurship Training Program (Project No.: S202513988017)".

References

- [1] Zhou Yang, Yang Puyun, Zhao Zhonghua, et al. Characteristics of Occurrence and Control of Major Crop Diseases and Pests in China in 2016 and Key Points of Control Work in 2017. *China Plant Protection Guide*, 2017, 37(05):60-64.
- [2] Wang Zhehao. Research on Tomato Leaf Disease Recognition and Classification Based on Convolutional Neural Network. Wuhan Polytechnic University, 2023.
- [3] Tan Zhifeng, He Zhilei, Zhang Lei, et al. Review on the Application of Deep Learning Technology in Crop Pest and Disease Identification. *Hebei Agriculture*, 2024, (01):30-32.
- [4] Zhou Jianglong, Wang Tianyi, Li Lun, et al. A lightweight crop leaf disease recognition model Based on Deep learning. *Jiangsu Agricultural Sciences*, 2024, 52 (16): 230-238.
- [5] Xu Xuefeng. Analysis of the Current Situation and Control Techniques of Crop Pests and Diseases in the New Era. *Hebei Agriculture*, 2024, (01):71-72.
- [6] Research on Corn Leaf Pest and Disease Identification System Based on Improved YOLOv5s Hu Mengjiao Hebei Normal University of Science and Technology 2023.
- [7] Xu Hongmei. Analyzing the Application of SQLite in Python. *Neijiang Science and Technology*, 2023, 44(04):60-61.
- [8] Dong Qin, Yang Guoyu. Crop Pest and Disease Identification Based on Optimized CBAM and Improved YOLOv5. *Applications of Computer Systems*, 2023, 32(07):261-268.
- [9] Tao Zhi, Kong Jianlei, Jin Xuebo, et al. Design of an APP System for Crop Pest and Disease Image Recognition Based on Deep Learning . *Computer Applications and Software*, 2022, 39(03):341-345.
- [10] Zhu Deming, Cheng Xiangping, Qiu Yijian, et al. Research Progress of Crop Image Recognition Technology Based on Deep Learning . *Jiangxi Science*, 2025, 43 (01): 154-161.