Proceedings of the 3rd International Conference on Biological Engineering and Medical Science
DOI: 10.54254/2753-8818/24/20231164

Study on the automatic detection of pepper TSWV disease based on ResNet

Shibei Ming

Shanghai World Foreign Language Academy, Shanghai, 201100, China

violaming2006@gmail.com

Abstract. This study focuses on the development of a binary classification model for Tomato Spotted Wilt Virus (TSWV) recognition in peppers. A diverse dataset is created for model training, which incorporates TSWV-infected pepper imagery with varying infection severity and regions of different sizes and resolution. The model is trained based on ResNet-18 architecture, resulting in excellent performance of metrics with scores reaching 0.990. Thus, the developed model may have applicational potential for pepper TSWV recognition in real-life conditions.

Keywords: Tomato Spotted Wilt Virus (TSWV); Deep Learning; Plant Disease Detection

1. Introduction

Tomato Spotted Wilt Virus (TSWV) is an orthotospovirus that can infect over 900 monocotyledonous and dicotyledonous plants. First identified as a viral disease after its appearance in Australia in 1915 [1], it has widely spread in temperate, subtropical, and tropical zones and is now present in almost all countries. In China, TSWV is a quarantine plant virus, has infected economic crops such as tobaccos, peppers, tomatoes, and potatoes, causing tremendous economic loss in all parts of China since its first occurrence in Guangzhou in 1989. Typical features of infected plants include lesion, stunting, and ringspots on both leaf and fruit; in early stages, leaves on infected plants usually become deformed and curly, sometimes with chlorosis along the leaf vein, and in later stages, ringspots and mottle often develop on these plants, resulting in necrosis in all parts of infected plants.

Currently, there are no direct cure for TSWV, and diseased plants are usually eradicated from the field to prevent further spread of virus. Therefore, quick scanning of field plants and accurate recognition of diseased plants is crucial. Contrary to traditional biological assays, which are time-consuming and expensive, artificial intelligence (AI) models have appeared as a reliable option, being widely researched throughout these years. AI models can distinguish diseased plants from healthy ones based on classification algorithms, most common ones including logistic regression [2], CNN [3], Neural Networks [4], Support Vector Machines (SVM) [5], Decision Trees [6], k-Nearest Neighbors (k-NN) [7], and Naive Bayes [8].

2. Literature Review

CNN-based models are widely used in the detection of plant diseases. CNNs are effective for classifying images due to their ability to extract local features and learn hierarchical representations [9, 10]. Commonly used architectures of CNN include AlexNet, GoogLeNet, VGGNet, Inception-V3,

DOI: 10.54254/2753-8818/24/20231164

Inception-V4, ResNet, and DenseNets. Zhang et al. used ResNet, recognized as the most effective CNN architecture, GoogeNet and AlexNet, to detect tomato leaf diseases [11]. S. Sladojevic et al. identified 13 common plant diseases based on deep learning, with model average recognition accuract of 96.3% [12]. Cheng et al. used ResNet and AlexNet to identify agricultural pests; While taking comparative experiments with SVM and BP neural networks, they achieved best accuracy of 98.67% by ResNet-101 [13]. Sa et al., by adapting Faster Region-based CNN (Faster R-CNN) through transfer learning, introduced an innovative method for fruit detection, achieving F1 score of 0.83 in a field farm dataset [14].

For more effective detection, these deep learning neural networks are often used along with hyperspectral sensing, a technique used to capture and analyze information from a wide range of electromagnetic wavelengths beside regular RGB and can therefore indicate the presence of disease or stress through detecting unusual spectral signals. Yue et al. presented an innovative hybrid approach for hyperspectral image classification, combining principal component analysis, logistic regression, and deep convolutional neural network algorithms. Their methodology yielded the most favorable outcomes in terms of classification accuracy [15]. To detect diseases in tomato plants before visible symptoms emerge, a method utilizing generative adversarial networks (GANs), a set of deep learning technique that operates on hyperspectral proximal-sensing data, was introduced [16].

Despite the accuracy of hyperspectral-sensing-based systems, they have limitations in wide-range application. Not only requiring a detailed setup for operation, but the hyperspectral camera also tend to be more costly than regular RGB cameras. Furthermore, hyperspectral cameras often sacrifice spatial resolution to achieve superior spectral resolution. Considering these factors, the practical advantages of hyperspectral systems are limited; in many current studies, the observations toward plant disease was reflected through morphological changes, which can also be identified under visible wavelengths [17-20].

Still, despite the narrower range of wavelengths RGB-based systems, in the study of Ahmad et al. on detecting crop diseases by DenseNet169-based RGB cameras, the employed model demonstrated exceptional performance with a minimal loss value of 0.0003 and testing accuracies reaching 100%, indicating the great potential of RGB imagery in identifying diseased regions [21].

3. Methodology

3.1. Outline

This study aims to develop a TSWV pepper binary classification model based on RGB images, which has the potential for field application in commercial farming, based on ResNet 18.

3.2. Dataset

3.2.1. Collection Requirements. Considering the objectives of this study, three main aspects of image collection requirements are set to increase the complexity of the model:

3.2.1.1. Sample Diversity. Sample images need to have high diversity to ensure that the training data can provide discriminative information for the model. In turn, the similarity between sample images is kept at minimum to avoid overfitting issues. To this end, different types of sample diversity are considered.

First, for TSWV-infected dataset, images showing a variety of symptoms during different stages of infection should be included. The TSWV disease severity is classified into four stages: in the first stage of infection, leaf deformation and chlorosis along the vein and are observed (figure 1b); then, a few mottles appear on leaf surface, which will later enlarge into connected ringspots (figure 1c); finally, ringspots occupy for more than half of the leaf and may develop into necrosis (figure 1d).

DOI: 10.54254/2753-8818/24/20231164

Figure 1. Healthy pepper leaf (a) and typical TSWV infected leaf samples with symptoms during multiple stages of infection (b) (c) (d).

Second, different organs of peppers are included, which are mainly composed of fruit (figure 2a), stem (figure 2b), and leaves (figure 2c). Chlorosis and ringspots can be the common feature among various organs; nevertheless, stem abnormal distortion and hyperplasia, fruit rotting along the ringspot, and necrosis of leaves can be distinctive features for each organ, which strengthens the model's ability of differentiating TSWV-infected plants from healthy ones.

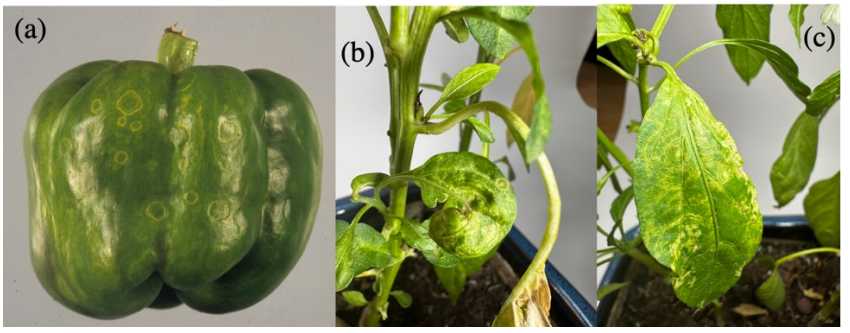

Figure 2. Images showing different organs of peppers.

Third, images of peppers with several shotting angles are included. In general, image categories are set as overhead shots, side shots, and close-up shots to record the morphological characters of peppers, especially TSWV lesion, under various angles.

3.2.1.2. Sample Size. Due to the efficiency of ResNet derived from its residual connections and parameter efficiency compared to other image classification models, it only requires small sample size for effective classification between TSWV and healthy peppers. Based on previous application of ResNet in image classification, a rough sample size of 200-300 is prescribed in this study [11, 22].

3.2.1.3. Sample Resolution. Blurring can reduce the level of detail and discriminability between different regions in the pepper images, thus influencing the efficiency of training. Therefore, a range of requirements need to be set to avoid over-blurring. Yet a highly variable range of image resolution also need to be included in the dataset to develop the model's ability to recognize common blur images. have a large range of image resolution, images with different pixels and sizes will be used. The minimum resolution is 218×218 pixels, the input size of ResNet 18.

3.2.2. Collection Method. Based on the collection requirements explained in 3.2.1, images were mainly collected from home experiment, field observation, and online sources. Currently, the public pepper TSWV datasets online are relatively inadequate, with limited variation among images. Therefore, my dataset will be mainly composed of self-collected images, with online images as a supplement.

DOI: 10.54254/2753-8818/24/20231164

Sorted by the sources of images, the collection methods under each scenario are explained below.

3.2.2.1. Home experiment. The main proportion of the experiment is done at home due to relative convenience and minimal environmental concern regarding the TSWV transmission to outside world.

The procedures of planting peppers and collecting images are executed in unified mode.

Capsicum annum L. var. fasciculatum Irish., XiangLa 108 in specific, is chosen as the subject of this experiment due to its high adaptivity to environment. Total of 60 peppers seedlings were purchased from local farm fair and were planted in general garden soil with a layer of peat on the bottom; the 60 peppers were separated into 12 groups randomly, with every 5 peppers planted in one plastic 50cm*50cm*20cm square box. After the initial adaptation of 1 week, the peppers were taken photo every 4 days, with 1 overhead shot (figure 3a), 4 close-up shots (figure 3b), and 1 side shot (figure 3c) taken for each box. After 3 weeks, all the plants will be TSWV-inoculated through mechanical inoculation: slight amount of quartz sand was spread on the surface of plant leaves, and after dropping infected juice on leaves, the leaves would be rubbed with fingers along the stem. This operation was repeated 3 times for each leave, with 3 true leaves per pepper chosen for inoculation. 3 days after inoculation, 5 leave samples were chosen randomly from 12 boxes of peppers and sent to Nanjing Agricultural University for PCR test; all samples were tested as TSWV positive. After inoculation, at least 20 days of observation is mandatory, as visible symptoms of TSWV usually appear 10-14 days after infection on peppers.

Figure 3. Sample images for image collection.

3.2.2.2. Field Observation. Field observation were done at several places: Shared vegetable plots, green belts in local communities, farmland in rural areas, etc. Images from these sources represent the most natural scenarios of pepper growth or TSWV outbreak.

In shared vegetable plots, 10 pepper seedlings were planted and taken photo every week; also, for TSWV-infected peppers in green belts and farmlands, regular observations were also taken at about once per week. In some photos, the background noise was kept imitating the complex environment in field.

3.2.2.3. Online Sources. Online data is collected mainly to further supplement the current dataset by providing symptoms that are not present in the home experiment and field observation. The online images were collected from Invasive.Org (https://www.invasive.org/index.cfm) and Eppo Global Database (https://gd.eppo.int/taxon/TSWV00/photos).

3.2.2.4. Summary. During data collection, sample diversity, size, and resolution are the three main factors considered.

DOI: 10.54254/2753-8818/24/20231164

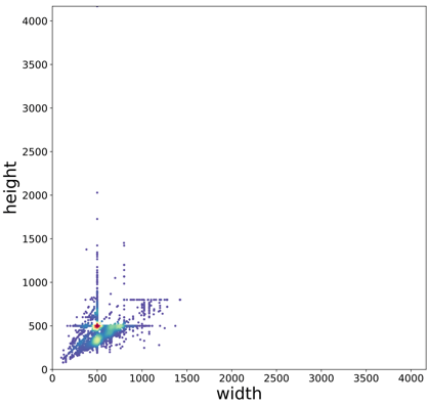

Figure 4. Height and Width scatter point graph.

3.3. Algorithm

3.3.1. ResNet

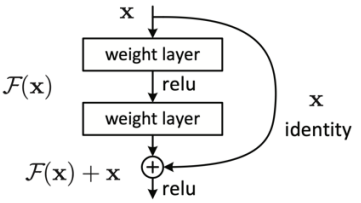

Figure 5. Residual Connection in ResNet [23].

The main feature of ResNet is the use of Residual Connection to solve the problem of gradient disappearance and model degradation in deep network training. In traditional convolutional neural networks, increasing the number of layers leads to the problem of gradient disappearance, which makes the network difficult to train. However, the residual connection introduces skip connection, which passes the input features directly to the subsequent layer, so that the network can learn the residual information better. With residual connections, ResNet can build very deep networks. Many ResNet architectures with different number of layers have been developed, such as ResNet18, ResNet34, ResNet50, ResNet101, and ResNet152.

3.3.2. Dataset. The collected data is randomly divided into train group and test group in a ratio of 4:1. Then, data augmentation is performed to expand the dataset, in form of horizontal and vertical flip (Table 1). Meanwhile, the images are zoomed to fit the fixed input size of ResNet18 (218×218 pixels). Currently, there are 385 photos available for training and 98 photos for testing (Table 1).

Table 1. Dataset Overview

Name	Pepper (*Capsicum annum L. var. fasciculatum Irish.*, XiangLa 108)		
Classification	0:Health 1:TSWV		
Number	Genre	Training Set	Test Set
	Health	192	48
	TSWV	193	50
Data Augmentation	True; Horizontal flip and vertical flip		

DOI: 10.54254/2753-8818/24/20231164

3.3.3. Model Training.

3.3.3.1. Method of transfer learning. As the types of images used in this experiment isn't highly similar to ImageNet, the dataset used by ResNet18, while they still share some similarities as there are close-to-life images both in my dataset and ImageNet, the original Resnet18 model will be fine-tuned through Adam optimization function to adjust parameters in fully connected layer.

3.3.3.2. Parameters and functions. In general, the images and labels of the training set will be captured and fed to the model, so the model can perform a forward prediction, obtain the predictions for the current batch, and then compare the predictions with the labels to calculate the cross-entropy loss function for the current batch. The settings of hyperparameters and methods are explained in the following.

Methods:

Cross-Entropy loss function

Cross-entropy loss is a measure of dissimilarity between predicted and actual probability distributions. It quantifies the difference between predicted probabilities and the true class labels. By minimizing cross-entropy loss during training, a model learns to accurately predict class probabilities.

In case of Binary Classification, cross-entropy is given by:

$$l = -(y\ log(p) + (1 - y)\ log\ (1 - p)\ \ [24]$$

p: the predicted probability

y: the indicator (0 or 1 in the case of binary classification)

Adam Optimizer

Adam Optimizer is an algorithm for stochastic optimization. It has less memory requirement and high efficiency.

StepLR

This method adjusts the learning rate of each parameter group by gamma every step_epoch.

$$lr = lr * gamma\ [21]$$

gamma (float) – Multiplicative factor of learning rate decay.

Hyperparameters:

$Batch\ size = 32$

$Epochs = 30$

3.3.4. Statistical Analysis

3.3.4.1. Loss (Cross Entropy Loss). Cross-entropy loss is a measure of dissimilarity between predicted and actual probability distributions. It quantifies the difference between predicted probabilities and the true class labels. By minimizing cross-entropy loss during training, a model learns to accurately predict class probabilities.

in the case of Binary Classification, cross-entropy is given by:

$l = -(y\ log(p) + (1 - y)\ log\ (1 - p))$ [24]

p: the predicted probability

y: the indicator (0 or 1 in the case of binary classification)

3.3.4.2. Confusion Matrix. All metrics are calculated based on confusion matrix. TP (true positive) is the number of actual TSWV images that is predicted as TSWV by the model; FP (false positive) is the number of healthy images that is predicted as TSWV by the model; FN (false negative) is the number of TSWV images that is predicted as healthy by the model; TN (true negative) is the number of healthy images that is predicted as healthy by the model.

DOI: 10.54254/2753-8818/24/20231164

Table 2. Confusion Matrix.

	Predicted TSWV	**Predicted Healthy**
Actual TSWV	TP	FN
Actual Healthy	FP	TN

The following metrics are used in this study to evaluate the performance of the TSWV pepper binary classification model:

3.3.4.3. Accuracy. Accuracy is a measure of accuracy of classification model to predict the overall indicator. It represents the proportion of the number of samples correctly predicted by the model to the total number of samples. In TSWV recognition, accuracy suggests what proportion of peppers are correctly identified by the model.

$$Accuracy = \frac{TP + TN}{TP + FP + FN + TN}$$

3.3.4.4. Precision. Precision is a measure of how accurately a model predicts a positive outcome. It indicates how many of the samples predicted as positive by the model are true positive samples. In this case, the precision reflects what proportion of peppers predicted as TSWV by the model are actual TSWV samples.

$$Precision = \frac{TP}{TP + FP}$$

3.3.4.5. Recall. The recall rate is the ability to measure model correctly identify are samples. It indicates how many true positive samples the model has successfully found. In particular, recall value indicates what proportion of TSWV peppers are correctly detected by the model.

$$Recall = \frac{TP}{TP + FN}$$

3.3.4.6. F1-score. F1 score is an evaluation metric that takes into account both precision and recall. It measures the performance of the model by calculating the harmonic mean of precision and recall. The F1 score can take into account the prediction ability of the model for positive and negative samples at the same time and has good robustness in imbalanced data sets.

$$F = \frac{(a^2 + 1) \times precision \times Recall}{a^2 \times Precision + Recall}$$

$$F_1 = \frac{2 \times precision \times Recall}{Precision + Recall} \ (a = 1)$$

4. Results

This study uses accuracy curve, loss curve, recall curve, precision curve and F1 score curve to analyze the performance of the model, as shown in Figure 6-11.

The accuracy curve is calculated from the prediction results under different thresholds. It shows the accuracy of the model under different classifier output thresholds. In general, the steeper the rise of the accuracy curve, the better the performance of the model in the classification task. From figure 8, the accuracy of this experiment shows an increasing trend and gradually tends to be stable, finally reaching 99%.

The loss curve shows the change in the value of the loss function during the training of the model. The loss function is used to measure the difference between the prediction of the model and the true label. Through the loss curve, it can be known whether the model converges during training and whether underfitting or overfitting issues present. It can be seen that the loss curve of the model shows a decreasing trend, which indicates that the model is learning and gradually optimizing (Figure 6 and Figure 7).

DOI: 10.54254/2753-8818/24/20231164

Recall rate curve describes the model under different thresholds are cases of recall (positive samples). Recall is the proportion of samples that the model correctly predicts as positive. The recall rate curve model to better capture the positive cases.

Accurate rate curve describes the model under different thresholds are cases of prediction accuracy. The accuracy curve can show how the accuracy of the model changes for different thresholds.

F1 score, combining precision rate and recall rate, reflects the overall performance of the model. The F1-score curve aids the balance between recall and precision, because when there is a mutual constraint between the two, it needs to find a balance point to achieve the best overall performance. According to table 3, the final f1 scores of the model reached 0.990.

Table 3. Results of Metrics.

Model	Accuracy	Precision	Recall	F1-Score
Model_V1	0.990	0.990	0.990	0.990

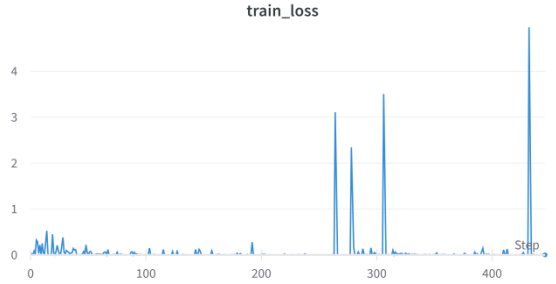

Figure 6. train_loss.

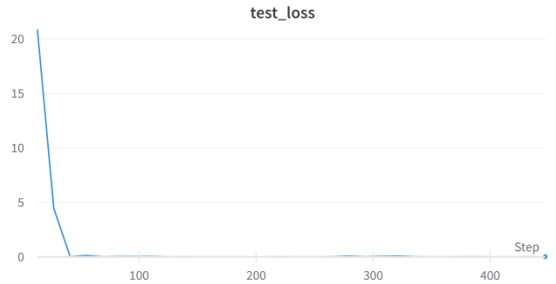

Figure 7. test_loss.

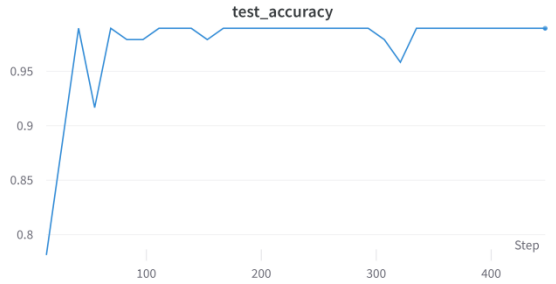

Figure 8. test_accuracy.

DOI: 10.54254/2753-8818/24/20231164

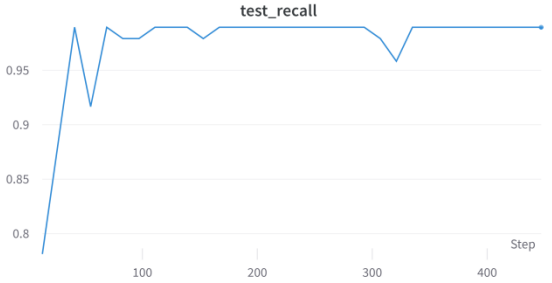

Figure 9. test_recall.

Figure 10. test_precision.

Figure 11. test_f1-score

5. Conclusion

In this study, I cultivated peppers and other plants to build a new dataset that include the morphological characteristics of TSWV-infected plants. Three types of main plant organs, stem, leaves, and fruits, are collected; TSWV dataset is composed of images of TSWV plant with four categories of infection severity to increase the diversity of disease dataset; images have abundant sample sizes with series of resolution; also, to approach images shot in real life, various images shotting angles are presents in the dataset. Considering the above features of my dataset, it comprises multifaceted visual information about TSWV plants and is valuable for future TSWV AI recognition studies, especially under the circumstances that many current TSWV plant datasets only include images of individual diseased leaves.

During model training, such diverse dataset helps to improve the complexity of my model. In order to cope with data insufficiency, data augmentation is applied; moreover, a model suitable for small dataset, ResNet-18, is chosen. Consequently, the training gained good results, with the final metrics all reaching 0.990. Therefore, this model is applicable to TSWV recognition.

In conclusion, this study developed a RGB TSWV pepper binary classification model based on ResNet 18. It introduces a self-built plant dataset that include diverse real-life images of TSWV-infected plants, which provides valuable data for future TSWV recognition studies.

By increasing the complexity and discriminative ability of the model, it finally achieved high scores in terms of metrics; thus, this model has potential for field application in commercial farming.

References

[1] Samuel G, Bald JG, Pittman HA. In: *Investigations on "Spotted Wilt" of Tomatoes.* Green HJ, editor. Government Printer; Melbourne, Australia: 1930. p. 120.

[2] Chang, C.-L., & Hsu, M.-Y. (2009). The study that applies artificial intelligence and logistic regression for assistance in differential diagnostic of pancreatic cancer. *Expert Systems with Applications*, *36*(7), 10663–10672. https://doi.org/10.1016/j.eswa.2009.02.046

[3] Sakshi, & Kukreja, V. (2021). A retrospective study on handwritten mathematical symbols and expressions: Classification and recognition. *Engineering Applications of Artificial Intelligence*, *103*, 104292. https://doi.org/10.1016/j.engappai.2021.104292

[4] Kukreja, V., Kumar, D., Kaur, A., Geetanjali, & Sakshi. (2020). Gan-based synthetic data augmentation for increased CNN performance in vehicle number plate recognition. *2020 4th International Conference on Electronics, Communication and Aerospace Technology (ICECA)*. https://doi.org/10.1109/iceca49313.2020.9297625

[5] Byvatov, E., Fechner, U., Sadowski, J., & Schneider, G. (2004). Comparison of support Vector Machine and artificial neural network systems for drug/Nondrug classification. *ChemInform*, *35*(5). https://doi.org/10.1002/chin.200405237

[6] Chen, W., Li, Y., Tsangaratos, P., Shahabi, H., Ilia, I., Xue, W., & Bian, H. (2020). Groundwater spring potential mapping using artificial intelligence approach based on kernel logistic regression, random forest, and alternating decision tree models. *Applied Sciences*, *10*(2), 425. https://doi.org/10.3390/app10020425

[7] Zhang, K., Wu, Q., Liu, A., & Meng, X. (2018). Can deep learning identify tomato leaf disease? *Advances in Multimedia*, *2018*, 1–10. https://doi.org/10.1155/2018/6710865

[8] Chen, W., Shirzadi, A., Shahabi, H., Ahmad, B. B., Zhang, S., Hong, H., & Zhang, N. (2017). A novel hybrid artificial intelligence approach based on the rotation forest ensemble and Naïve Bayes Tree Classifiers for a landslide susceptibility assessment in Langao County, China. *Geomatics, Natural Hazards and Risk*, *8*(2), 1955–1977. https://doi.org/10.1080/19475705.2017.1401560

[9] Datta, P., & Sharma, B. (2017). A survey on IOT architectures, Protocols, security and Smart City based applications. *2017 8th International Conference on Computing, Communication and Networking Technologies (ICCCNT)*. https://doi.org/10.1109/icccnt.2017.8203943

[10] Li, Q., Cai, W., Wang, X., Zhou, Y., Feng, D. D., & Chen, M. (2014). Medical image classification with Convolutional Neural Network. *2014 13th International Conference on Control Automation Robotics & Vision (ICARCV)*. https://doi.org/10.1109/icarcv.2014.7064414

[11] Yue, J., Zhao, W., Mao, S., & Liu, H. (2015). Spectral–spatial classification of hyperspectral images using deep convolutional neural networks. *Remote Sensing Letters*, *6*(6), 468–477. https://doi.org/10.1080/2150704x.2015.1047045

[12] Sladojevic, S., Arsenovic, M., Anderla, A., Culibrk, D., & Stefanovic, D. (2016). Deep Neural Networks based recognition of plant diseases by Leaf Image Classification. *Computational Intelligence and Neuroscience*, *2016*, 1–11. https://doi.org/10.1155/2016/3289801

[13] Cheng, X., Zhang, Y., Chen, Y., Wu, Y., & Yue, Y. (2017). Pest identification via deep residual learning in complex background. *Computers and Electronics in Agriculture*, *141*, 351–356. https://doi.org/10.1016/j.compag.2017.08.005

[14] Sa, I., Ge, Z., Dayoub, F., Upcroft, B., Perez, T., & McCool, C. (2016). DeepFruits: A Fruit Detection System using Deep Neural Networks. *Sensors*, *16*(8), 1222. https://doi.org/10.3390/s16081222

[15] Wang, D., Vinson, R., Holmes, M., Seibel, G., Bechar, A., Nof, S., & Tao, Y. (2019). Early detection of tomato spotted wilt virus by hyperspectral imaging and outlier removal auxiliary

classifier generative adversarial nets (or-ac-gan). *Scientific Reports*, *9*(1). https://doi.org/10.1038/s41598-019-40066-y

[16] *STEPLR*. StepLR - PyTorch 2.0 documentation. (n.d.). https://pytorch.org/docs/stable/generated/torch.optim.lr_scheduler.StepLR.html

[17] Khan, M. J., Khan, H. S., Yousaf, A., Khurshid, K., & Abbas, A. (2018). Modern trends in Hyperspectral Image Analysis: A Review. *IEEE Access*, *6*, 14118–14129. https://doi.org/10.1109/access.2018.2812999

[18] Kuswidiyanto, L. W., Noh, H.-H., & Han, X. (2022). Plant disease diagnosis using deep learning based on Aerial Hyperspectral Images: A Review. *Remote Sensing*, *14*(23), 6031. https://doi.org/10.3390/rs14236031

[19] Lowe, A., Harrison, N., & French, A. P. (2017). Hyperspectral image analysis techniques for the detection and classification of the early onset of plant disease and stress. *Plant Methods*, *13*(1). https://doi.org/10.1186/s13007-017-0233-z

[20] Moghadam, P., Ward, D., Goan, E., Jayawardena, S., Sikka, P., & Hernandez, E. (2017). Plant disease detection using hyperspectral imaging. *2017 International Conference on Digital Image Computing: Techniques and Applications (DICTA)*. https://doi.org/10.1109/dicta.2017.8227476

[21] Ahmad, A., Aggarwal, V., Saraswat, D., El Gamal, A., & Johal, G. S. (2022). GeoDLS: A deep learning-based corn disease tracking and location system using RTK geolocated UAS imagery. *Remote Sensing*, *14*(17), 4140. https://doi.org/10.3390/rs14174140

[22] Li, X., & Rai, L. (2020). Apple leaf disease identification and classification using Resnet models. *2020 IEEE 3rd International Conference on Electronic Information and Communication Technology (ICEICT)*. https://doi.org/10.1109/iceict51264.2020.9334214

[23] Şahan, S., Polat, K., Kodaz, H., & Güneş, S. (2007). A new hybrid method based on fuzzy-artificial immune system and -NN algorithm for breast cancer diagnosis. *Computers in Biology and Medicine*, *37*(3), 415–423. https://doi.org/10.1016/j.compbiomed.2006.05.003

[24] Maheshkar, S. (2021, September 10). *What is cross entropy loss? A tutorial with code*. W&B. https://wandb.ai/sauravmaheshkar/cross-entropy/reports/What-Is-Cross-Entropy-Loss-A-Tutorial-With-Code--VmlldzoxMDA5NTMx