Computers, Materials & Continua
DOI: 10.32604/cmc.2022.025714
Article

Tech Science Press

A Study on Small Pest Detection Based on a CascadeR-CNN-Swin Model

Man-Ting Li and Sang-Hyun Lee*

Department of Computer Engineering, Honam University, Gwangsangu, Gwangju, 62399, Korea
*Corresponding Author: Sang-Hyun Lee. Email: leesang64@honam.ac.kr
Received: 02 December 2021; Accepted: 09 February 2022

Abstract: This study aims to detect and prevent greening disease in citrus trees using a deep neural network. The process of collecting data on citrus greening disease is very difficult because the vector pests are too small. In this paper, since the amount of data collected for deep learning is insufficient, we intend to use the efficient feature extraction function of the neural network based on the Transformer algorithm. We want to use the Cascade Region-based Convolutional Neural Networks (Cascade R-CNN) Swin model, which is a mixture of the transformer model and Cascade R-CNN model to detect greening disease occurring in citrus. In this paper, we try to improve model safety by establishing a linear relationship between samples using Mixup and Cutmix algorithms, which are image processing-based data augmentation techniques. In addition, by using the ImageNet dataset, transfer learning, and stochastic weight averaging (SWA) methods, more accuracy can be obtained. This study compared the Faster Region-based Convolutional Neural Networks Residual Network101 (Faster R-CNN ResNet101) model, Cascade Region-based Convolutional Neural Networks Residual Network101 (Cascade R-CNN-ResNet101) model, and Cascade R-CNN Swin Model. As a result, the Faster R-CNN ResNet101 model came out as Average Precision (AP) (Intersection over Union (IoU)=0.5): 88.2%, AP(IoU = 0.75): 62.8%, Recall: 68.2%, and the Cascade R-CNN ResNet101 model was AP(IoU = 0.5): 91.5%, AP (IoU = 0.75): 67.2%, Recall: 73.1%. Alternatively, the Cascade R-CNN Swin Model showed AP (IoU = 0.5): 94.9%, AP (IoU = 0.75): 79.8% and Recall: 76.5%. Thus, the Cascade R-CNN Swin Model showed the best results for detecting citrus greening disease.

Keywords: Cascade R-CNN swin model; cascade R-CNN resNet101 model; faster R-CNN ResNet101 model; mixup; cutmix

1 Introduction

According to the "World Agricultural Organization's research results" on citrus greening disease, trees or leaves infected by the infestation of greening disease should be promptly discarded. Tolerant trees and uninfected seedlings must be used, and transmission cannot be stopped unless all diseased seedlings are found and removed from the orchard. Through these measuresand management, the

loss cost of the citrus orchard can be reduced as much as possible [1]. It is possible to manage such orchards and to manage pests by using artificial intelligence. Deep learning, which currently belongs to the scope of artificial intelligence, is widely used in image recognition and classification research. The convolutional neural network (CNN or ConvNet) [2] has shown excellent performance in classifying damage and diseases of crops such as pears, peaches, apples, grapes, and tomatoes in agricultural research [3].

Based on the related CNN model, researchers developed a detection system for citrus with greening disease [4]. In addition, another researcher developed a system which can confirm images of diseases for 26 plants, including citrus greening disease, and the overall detection accuracy was also high [5]. Another researcher established a model to detect citrus greening disease through a neural network and proposed four classifications for 8 categories of abnormal symptoms, and among them, the accuracy of detecting citrus greening disease was 93.7% [6].

The research direction of this paper is to find a method for the early detection and prevention of pests by examining greening disease of citrus trees using the Cascade Region-based Convolutional Neural Networks (Cascade R-CNN) Model [7]. The difficulty in identifying the existing citrus greening disease is that, as shown in Fig. 1, the greening disease vector pest is too small to identify with a system with low performance and it is very difficult to collect learning data.

Figure 1: The appearance of pests that carry citrus greening disease

The data to be used in this paper needs to be enlarged on the screen to accurately detect the very small pests, and the accuracy is lowered because the amount of collected data is very small. To solve this problem, this paper uses a transformer model which uses a method called 'Self-Attention' [8]. Self-Attention was created to overcome the limitations of recurrent neural network (RNN) [9], which was slow in operation due to difficulty in parallel processing. We want to use the Cascade R-CNN Model to detect citrus infection with greening disease.

The purpose of this study is to achieve a better result in detecting some small pest targets or some obscured disease targets by using the Cascade R-CNN target detection model based on swin transformer feature extraction network with a small amount of original sample data.

Here it can effectively prevent overfitting and false positives caused by a fixed intersection over union (IoU) threshold that is too high or too low and based on the ImageNet data set, transfer learning and stochastic weight averaging (SWA) [10] are used to achieve higher accuracy. Here it can effectively prevent overfitting and false positives caused by a fixed IoU threshold that is too high or too low and based on the ImageNet data set, transfer learning and stochastic weight averaging (SWA) [11] are used to achieve higher accuracy.

In this study, we can solve Adam W's Loss bounding problem by optimizing the parameter update using stochastic weight averaging (SWA) to improve the stability of the model parameter update

This paper is organized as follows: Section 2 describes the structure of the model proposed in this study; Section 3 describes the configuration of the whole system to be studied; Section 4 describes the experimental results; and finally, Section 5 discusses the conclusion of this paper.

2 Structure of the Swin Transformer Model and Cascade R-CNN Model

2.1 Swin Transformer

Modeling in computer vision has long been dominated by convolutional neural networks (CNNs) [12]. However, the evolution of network architectures in natural language processing (NLP) has taken a different path, where the prevalent architecture is today instead of the transformer [13]. Designed for sequence modeling and transduction tasks, the transformer is notable for its use of attention to model long-range dependencies in the data. Its tremendous success in the language domain has led researchers to investigate its adaptation to computer vision, where it has recently demonstrated promising results on certain tasks, specifically image classification [14] and joint vision-language modeling.

The researched a new vision transformer by 2021, called the swin transformer, which capably serves as a general-purpose backbone for computer vision. Challenges in adapting transformer from language to vision arise from differences between the two domains, such as large variations in the scale of visual entities and the high resolution of pixels in images compared to words in text. To address these differences, researcher propose a hierarchical transformer whose representation is computed with shifted windows [15].

The shifted windowing scheme brings greater efficiency by limiting self-attention computation to non-overlapping local windows while also allowing for cross-window connection [16]. This hierarchical architecture has the flexibility to model at various scales and has linear computational complexity with respect to image size.

Fig. 2a the proposed swin transformer builds hierarchical feature maps by merging image patches (shown in gray) in deeper layers and has linear computation complexity when inputting image size due to computation of self-attention only within each local window (shown in red). It can thus serve as a general-purpose backbone for both image classification and dense recognition tasks. (b) In contrast, previous vision transformers produce feature maps with a single low resolution and have quadratic computation complexity when inputting image size due to computation of self-attention globally [17].

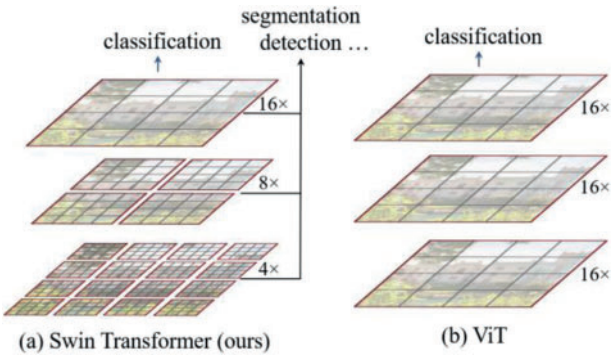

Figure 2: Structure of the swin transformer and ViT

The swin transformer makes it compatible with a broad range of vision tasks, including image classification and dense prediction tasks such as object detection and semantic segmentation.

2.2 Cascade R-CNN

In object detection, an intersection over union (IoU) threshold is required to define positives and negatives [18]. An object detector, trained with a low IoU threshold at 0.5, usually produces noisy detections. However, detection performance tends to degrade when increasing the IoU thresholds. Two main factors are responsible for this: 1) overfitting during training, due to exponentially vanishing positive samples and 2) inference-time mismatch between the IoU, for which the detector is optimal, and those of the input hypotheses. A multi-stage object detection architecture, the Cascade Region-based Convolutional Neural (Cascade R-CNN), is proposed to address these problems. It consists of a sequence of detectors trained with increasing IoU thresholds, to be sequentially more selective against close false positives. The detectors are trained stage by stage, leveraging the observation that the output of a detector is a good distribution for training the next higher quality detector [19].

The Cascade Region-based Convolutional Neural Network (Cascade R-CNN) is very similar to the Faster Region-based Convolutional Neural Network (Faster R-CNN) and is largely divided into two steps, the first of which is to locate the target and the second being to classify the target. Referring to Fig. 3, the Faster R-CNN [20] classifier is on the left and the Cascade R-CNN classifier is on the right. In Faster R-CNN, pool is a pooling layer [21] for feature maps [22].

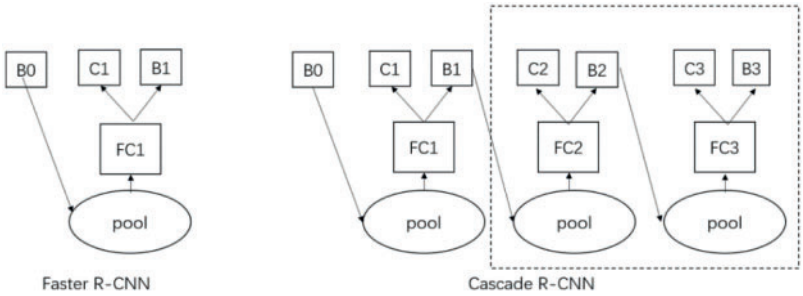

Figure 3: Structure of the searcher of faster R-CNN and cascade R-CNN

FC1 is the fully connected layer, B0 is the boundary box of the candidate region, B1 is the predicted boundary box from the structure, and C1 is the final prediction classification result. The pool of Cascade R-CNN is a pooling layer for the feature map. FC1, FC2, and FC3 represent the complete connected layer, and B0, B1, and B2 represent the bounding box of the candidate region. B3 indicates the predicted bounding box of the structure, C1 and C2 indicate the predicted classification result, and C3 is the result for the final predicted classification. Since the Cascade R-CNN classifier uses the cascade method, better data can be provided to the next classifier because the output value of the previous classifier is used as the input value of the next classifier. For this reason, the classifier has the advantage of showing higher effectiveness [23].

Cascade R-CNN uses three cascade detectors in the classification and regression stages, and by gradually increasing the IoU threshold, the candidate frame is continuously optimized and the detection result becomes more accurate. Additionally, it can effectively prevent overfitting and false positives caused by a fixed IoU threshold that is too high or too low [23].

3 Citrus Greening Bottle Detection System Using the Swin Transformer Model

The transformer used in this paper uses a method called 'Self-Attention'. The transformer's attention was created to overcome the limitations of Recurrent Neural Network (RNN), which

was slow in operation due to difficulty in parallel processing. To translate a given word, it has to be compared against all other words in the sentence. Transformers do not need to process data sequentially like RNNs. This approach is also possible because it allows much more parallelism than RNNs. The transformer, which translates entire sentences in a parallel structure to increase similarity by making associations even with distant words, enhances language comprehension ability when learning deep learning by supplementing the RNN model.

This paper detects the target of greening disease or diseased seedlings in citrus orchards through the transformer model. Here, high-definition cameras or drones are used to collect image data from the citrus orchard. The collected video is converted into a static image by the frame technique, and data is collected through labeling for effective images. Finally, leaves and pests overlapping with citrus greening disease are detected by other target detection models, and the detection performance is compared by comparing the Cascade R-CNN Model, the Cascade R-CNN-ResNet (Residual Network)101 model, and the Faster R-CNN-ResNet101 model.

3.1 Structure of the Proposed System

The proposed system architecture design and components according to the citrus greening disease detection network model architecture design are described.

The citrus greening bottle system architecture is implemented with image acquisition, image enhancement, real-time target detection, data warehouse storage, and web visualization. The process of the overall system architecture is as shown in Fig. 4: (i) the image acquisition part uses a drone equipped with a high-definition camera to capture the citrus orchard and transmits it to the image processing part; (ii) in the image enhancement part, there are image data preprocessing and data enhancement; (iii) the real-time target detection detects the leaves and pests of trees with citrus greening disease in the image; (iv) visualization is performed and risk warning occurs when detecting diseased leaves and pests in the real-time target detection part; and (v) daily data detection results and images are stored in the data warehouse.

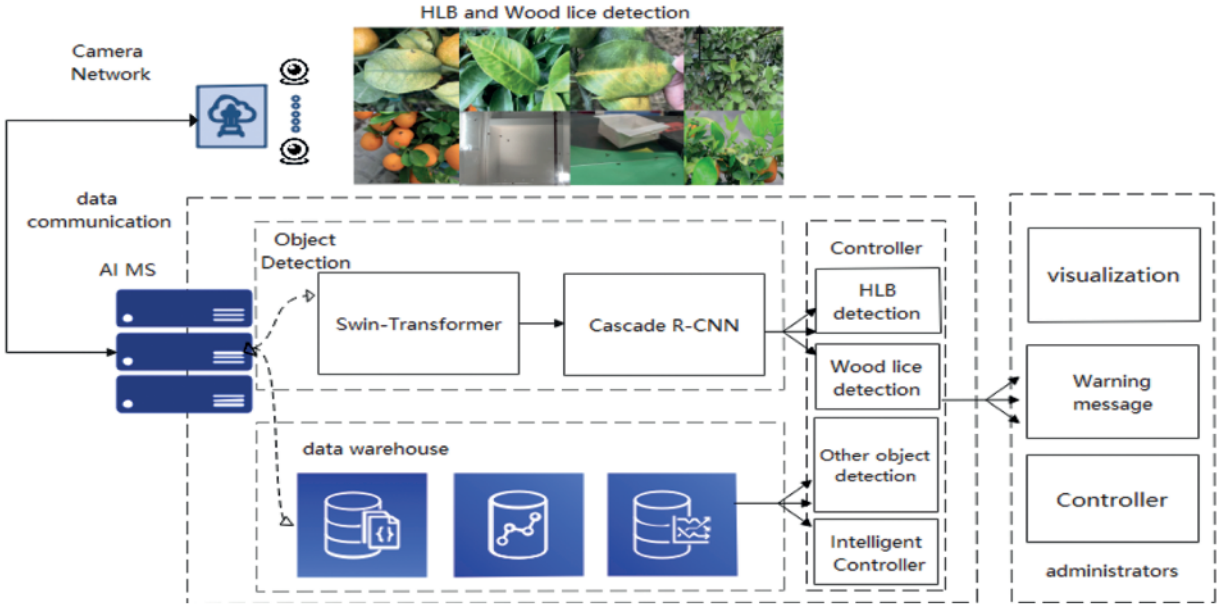

Figure 4: Structure of the proposed system

3.2 Citrus Pest Detection Process Based on Cascade R-CNN and Swin-Transformer Model

Fig. 5 is a learning process for citrus greening disease detection based on Cascade R-CNN Swin. First, we selected a high-quality image from a citrus orchard and built a data set through labeling. The number of original image data is 725, so learning can occur with very little data. However, in this paper, data augmentation methods such as Random Flip, Random Rotate, and Grid mask are used to solve the problem of too little data. The data quantity using the augmentation method will increase the quantity by 250% compared to the original data set.

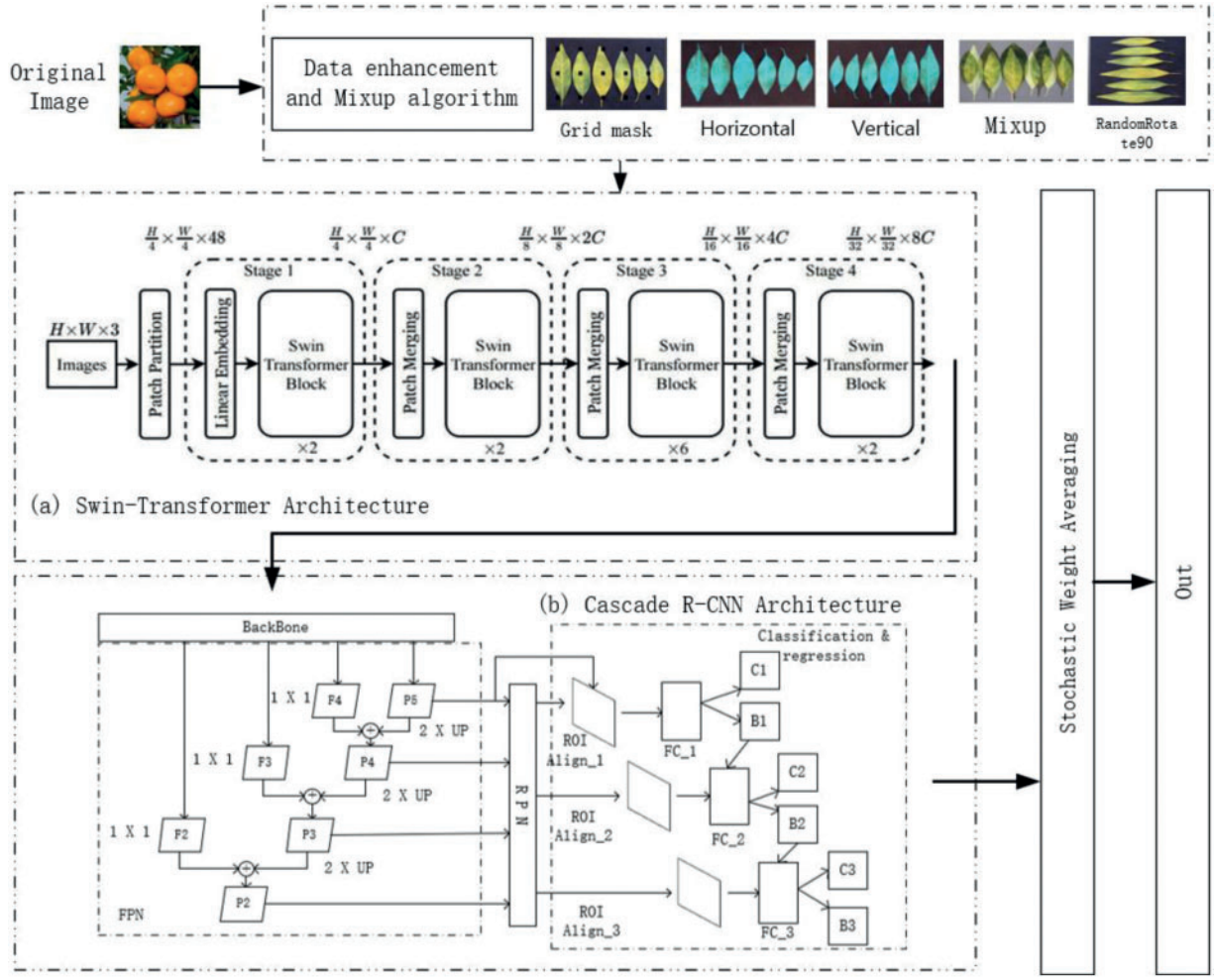

Figure 5: Learning process for citrus greening disease detection based on the cascade R-CNN and swin transformer

Cascade R-CNN-Swin is based on the Transformer, the feature extraction network of the Swin Transformer Neural Network, which uses the backbone network to extract the object features and learns from the Cascade R-CNN Model.

Fig. 5a shows the overall structure of the swin transformer. It is largely divided into patch partition, linear embedding, swin transformer, and path merging and consists of 4 stages (x2/x2/x6/x2) which are written under each stage indicating the number of swin transformer blocks. Since 2 encoders

are attached to each block, 1/1/3/1 block are actually repeated by grouping them as a set. H/4 x W/4 x C, written on each stage, is patch x patch x channel, where 48 is obtained as the initial patch size x channel (4x4x3), and C uses 96 in the base model swin-transformer. The swin transformer block replaced the vision transformer's Multi-head Self-Attention (MSA) with Window Multi-head Self Attention (W-MSA) and Shifted Window Multi-head Self Attention (SW-MSA). The reason for the replacement is that MSA is a standard of the self-attention process of the transformer, but if it is used in an image, its cost is very high because each pixel goes through a process of referencing the entire pixel value on the image. Here, W-MSA divides the image into 4 windows and performs self-attention for each window. SW-MAS is a shifted window MSA, which shifts by half the size of the window sizes wH and wW [23].

Fig. 5b shows the classification and regression process of the Cascade R-CNN citrus greening disease detection model. The region proposal is generated with the detector trained with IoU = 0.5 of the main and regression parts of the Cascade R-CNN Citrus Greening Disease detection model, and IoU = 0.6 is trained.

Therefore, the final output value is derived by learning the detector with IoU = 0.7 as the result of the detector.

It is composed of three stages, and it is said that more stages would adversely affect performance. The cascade structure is not only applied to the train, but the cascade structure shown in Fig. 5b is also used for inference.

4 Implementation

4.1 Development Environment

In this study, the development environment for the experiment was developed using Python's 3.7 version, and the PyTorch-based MMDetection API was used for the artificial intelligence library. In the training and test environments, the OS was Windows 10, the CPU was i9-9900k, the RAM was 128 GB, and the GPU was NVIDIA RTX 6000. The detailed development environment is shown in Tab. 1.

Table 1: Development environment

Division	Specification
operating system (OS)	Ubuntu 18.04
central processing unit (CPU)	intel i9 9900 K
graphics processing unit (GPU)	NVIDIA RTX6000
Memory	128 GB
Storage	Samsung M.2 1TB

4.2 Image Reinforcement Learning

Histogram equalization generally increases the overall color sharpness of an image when the image is represented by a narrow range of intensity values. By evenly applying the overall color sharpness, it is possible to flatten the intensity of the histogram.

Therefore, an area with low color sharpness can achieve high color sharpness in the surrounding area, and good results are obtained in light or dark images. The histogram equalization image enhancement result is shown in Fig. 6.

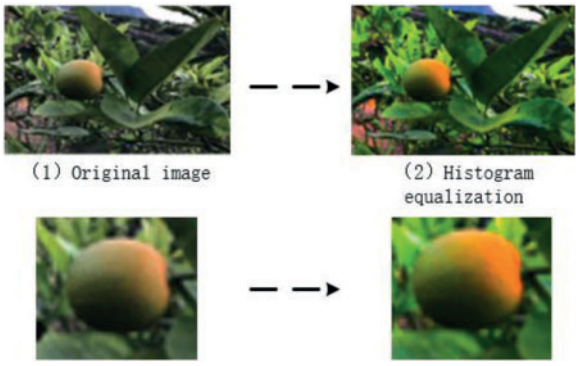

Figure 6: Comparison of the differences before and after histogram equalization filter processing

Tab. 2 describes the performance results obtained by training the model for detecting greening-diseased citrus using the existing data set and the histogram averaged data set. As a result of the detection of citrus fruits with greening disease using the existing data set, Average Precision (AP) (IoU = 0.5) was 82.2%, Average Precision (AP) (IoU = 0.75) was 60.4%, and Recall was 68.2%.

Table 2: Experimental results of the enhanced image

Division	AP (IoU = 0.5)	AP (IoU = 0.75)	Recall
Existing data set	82.2%	60.4%	68.2%
Histogram Normalized DataSet	86.0%	63.3%	71.5%

The results of detecting citrus with greening disease using the histogram normalization treatment data set improved AP (IoU = 0.5), AP (IoU = 0.75), and Recall by 3.85%, 2.96%, and 3.32%, respectively. Experimental results show that image enhancement can increase the diversity of features and improve the accuracy of training results based on the original data. These results show that the performance of the neural network model is linearly and positively related to the number of training samples.

4.3 Data Augmentation

Random Flip, Random Rotate 90, and Grid data augmentation methods used to make the network model have various characteristics of the data set before training to detect greening-diseased citrus. In addition, the stability of the model is improved by establishing a linear relationship between samples by utilizing the Mixup algorithm [24].

The results of detecting citrus with greening disease using the existing data set are AP (IoU = 0.5): 85.4%, AP (IoU = 0.75): 63.7%, and Recall: 71.1%. Using the data set processed by the data augmentation method, the detection of greening-diseased citrus fruits improved to AP (IoU = 0.5), AP (IoU = 0.75), Recall: 3.33%, 4.21%, and 3.97%, respectively. Tab. 3 shows the evaluations after using the existing data set.

Table 3: Evaluation after using an existing data set

Division	AP (IoU = 0.5)	AP (IoU = 0.75)	Recall
Existing data set	85.4%	63.7%	71.1%
Histogram Normalized Data Set	88.7%	67.0%	75.3%

4.4 Results of the Detection of Citrus Infected with Greening Disease

The learning loss can be confirmed in Tab. 4 with the results of the training process to detect citrus with greening disease using the Cascade R-CNN Model used in this study.

Table 4: Evaluate after using an existing data set

Iter	Loss	Loss_bbox	Loss_cls
0	1.078	0.067	0.220
2,500	0.696	0.034	0.035
5,000	0.636	0.039	0.033
		~	
15,000	0.510	0.036	0.020
17,500	0.482	0.040	0.017

When training to detect citrus with greening disease with the Cascade R-CNN Model, the Loss, Loss_bbox, and Loss_cls values were checked for each Iter. As a result, it was measured as 0 Iter: 1.078, 5,000 Iter: 0.636, and 17,500 Iter: 0.382. The robustness of the network appeared according to the change in the loss value according to the training. For Loss_bbox and Loss_cls they are 0.067 and 0.220, respectively, with 0 Iter. At 2,500 Iter, 0.034 and 0.035, at 15,000 Iter, 0.036 and 0.020, and finally, at 17,500 Iter, it showed stable low errors around 0.040 and 0.017, respectively.

The Citrus Greening Disease detection model used and trained Faster R-CNN or CascadeR-CNN, and the compared values of the performance results according to the model are shown in Tab. 5.

Table 5: Development environment

Model	Backbone	AP (IoU = 0.5)	AP (IoU = 0.75)	Recall
FasterR-CNN	ResNet101	0.882	0.628	0.682
CascadeR-CNN	ResNet101	0.915	0.672	0.731
CascadeR-CNN	Swin-Transformer	0.949	0.798	0.765

AP (IoU = 0.5): 88.2%, AP (IoU = 0.75): 62.8%, and Recall: 68.2% of the Citrus Greening Disease detection model using the Faster R-CNN-ResNet101 Model. When the CascadeR-CNN-ResNet101 Model was used, the performance of the model for detecting greening-diseased citrus was approximately 91.5%, 67.2%, and 73.1% for AP (IoU = 0.5), AP (IoU = 0.75), and Recall, respectively. On the other hand, the result of comparing the Cascade R-CNN Model with the Swin-Transformer

backbone and the Cascade R-CNN Model with the ResNet101 backbone was as high as 3.4% for AP (IoU = 0.5) and as high as 12.6% for AP (IoU = 0.75). Recall was also as high as 3.4%.

5 Conclusion

This paper used the Cascade R-CNN Model to detect greening disease in citrus fruits.

In addition, the backbone of the model was selected as a feature extraction model using the Swin Transformer Neural Network based on the Transformer. Thus, it was possible to have the effect of expanding the range of the reception field of each network layer by extracting more detailed and multi-scale features.

In this paper, in terms of model research, the FasterR-CNN Model was selected to compare the performance of the Cascade R-CNN Model, and the backbone of the two algorithms was set to ResNet-101 for comparison. Cascade R-CNN, a Swin-Transformer backbone, was compared with Cascade R-CNN, a ResNet101 backbone.

As a result of the study, Cascade R-CNN showed AP (IoU = 0.5): 91.5%, AP (IoU = 0.75): 67.2%, and Recall: 73.1% in detecting greening-diseased citrus.

The Cascade R-CNN Model showed AP (IoU = 0.5): 3.3% higher, AP (IoU = 0.75): 4.4%, and Recall: 4.9%, than the Faster R-CNN Model, whereas the Cascade R-CNN Model, a Swin-Transformer backbone, when compared with the Cascade R-CNN Model, which is the ResNet101 backbone, was AP (IoU = 0.5): 3.4% and AP (IoU = 0.75): 12.6% higher. Recall was also as high as 3.4%.

Through this, the Cascade R-CNN Model, a Swin-Transformer backbone, showed higher performance in detecting greening disease than the Faster R-CNN Model, which is the ResNet101backbone, and the Cascade R-CNN, which is the ResNet101backbone.

Therefore, the superiority of the Cascade R-CNN Model, the Swin-Transformer backbone proposed in this paper, was demonstrated. Based on the results of this study, we intend to further develop a model necessary for agriculture.

Funding Statement: This research was supported by the Honam University Research Fund, 2021.

Conflicts of Interest: The authors declare that they have no conflicts of interest to report regarding the present study.

References

[1] M. Z. U. Rehman, F. Ahmed, M. A. Khan, U. Tariq, S. S. Jamal *et al.,* "Classification of citrus plant diseases using deep transfer learning," *Computers, Materials & Continua*, vol. 70, no. 1, pp. 1401–1417, 2022.

[2] H. Farman, J. Ahmad, B. Jan, Y. Shahzad, M. Abdullah *et al.,* "Efficient net-based robust recognition of peach plant diseases in field images," *Computers, Materials & Continua*, vol. 71, no. 1, pp. 2073–2089, 2022.

[3] F. N. Al-Wesabi1, A. A. Albraikan, A. M. Hilal, M. M. Eltahir, M. A. Hamza *et al.,* "Artificial intelligence enabled apple leaf disease classification for precision agriculture," *Computers, Materials & Continua*, vol. 70, no. 3, pp. 6223–6238, 2022.

[4] Z. Liu, X. Xiang, J. H. Qin, Y. Tan, Q. Zhang *et al.,* "Image recognition of citrus diseases based on deep learning," *Computers, Materials & Continua*, vol. 66, no. 1, pp. 457–466, 2021.

[5] L. E. Ehler, "Integrated pest management (IPM): Definition, historical development and implementation, and the other IPM," *Pest Management Science*, vol. 62, no. 9, pp. 787–789, 2016.

[6] X. Du and X. Xiang, "Research on prevention of citrus anthracnose based on image retrieval technology," *Journal of Information Hiding and Privacy Protection*, vol. 2, no. 1, pp. 11–19, 2020.
[7] Y. Zhu and C. and Ma, J. Du, "Rotated cascade R-CNN: A shape robust detector with coordinate regression," *Pattern Recognition*, vol. 96, pp. 106964, 2019.
[8] Y. Li, J. Liu and S. Shang, "WMA: A multi-scale self-attention feature extraction network based on weight sharing for VQA," *Journal on Big Data*, vol. 3, no. 3, pp. 111–118, 2021.
[9] Y. Shen, Y. Li, J. Sun, W. K. Ding and X. J. Shi, "Hashtag recommendation using LSTM networks with self-attention," *CMC Computers, Materials & Continua*, vol. 61, no. 3, pp. 1261–1269, 2019.
[10] A. Majid, M. A. Khan, M. Alhaisoni, N. Hussain and U. Tariq, "An integrated deep learning framework for fruits diseases classification," *CMC Computers, Materials & Continua*, vol. 71, no. 1, pp. 1387–1402, 2022.
[11] N. Gul, S. Ahmed, A. Elahi, S. M. Kim and J. Kim, "Optimal cooperative spectrum sensing based on butterfly optimization algorithm," *CMC Computers, Materials & Continua*, vol. 71, no. 1, pp. 369–387, 2022.
[12] S. H. Lee, "A study on classification and detection of small moths using CNN model," *CMC Computers-Materials & Continua*, vol. 71, no. 1, pp. 1987–1998, 2022.
[13] G. N. Chandrika, K. Alnowibet, K. S. Kautish, E. S. Reddy and A. F. Alrasheedi, "Graph transformer for communities detection in social networks," *CMC Computers-Materials & Continua*, vol. 70, no. 3, pp. 5707–5720, 2022.
[14] Z. Deng, B. Zhou, P. He, J. Huang and O. Alfarraj, "A Position-aware transformer for image captioning," *CMC Computers, Materials & Continua*, vol. 70, no. 1, pp. 2065–2081, 2022.
[15] Z. Liu, H. Hu, Y. Lin, Z. Yao, Z. Xie *et al.,* "Swin transformer V2: Scaling up capacity and resolution," 2021. [Online]. Available: https://arxiv.org/abs/2111.09883.
[16] Z. Liu, Y. Lin, Y. Cao, H. Hu, Y. Weiet *et al.,* "Swin transformer: Hierarchical vision transformer using shifted windows," 2021. [Online]. Available: https://arxiv.org/abs/2103.14030.
[17] C. Papageorgiou and T. Poggio "A trainable system for object detection," *International Journal of Computer Vision*, vol. 38, no. 1, pp. 15–33, 2000.
[18] Y. Zhu, C. Ma and J. Du, "Rotated cascade R-CNN: A shape robust detector with coordinate regression," *Pattern Recognition*, vol. 96, pp. 106964, 2019.
[19] R. Meng, S. G. Rice, J. Wang and X. Sun, "Rotated cascade R-CNN: A shape robust detector with coordinate regression," *CMC Computers, Materials & Continua*, vol. 55, no. 1, pp. 1–16, 2018.
[20] L. Li, S. Li and J. Su, "Rotated cascade R-CNN: A shape robust detector with coordinate regression," *CMC Computers, Materials & Continua*, vol. 69, no. 2, pp. 2355–2366, 2021.
[21] J. Chen, Z. Zhou, Z. Pan and C. Yang, "Instance retrieval using region of interest based CNN features," *CMC Computers, Materials & Continua*, vol. 1, no. 2, pp. 87–99, 2019.
[22] J. C. Chen, Z. Zohu, Z. Pan and C. N. Yang, "Instance retrieval using region of interest based CNN features," *Journal of New Media*, vol. 1, no. 2, pp. 87–99, 2019.
[23] Z. Cai, and N. Vasconcelos, "Cascade R-CNN: Delving into high quality object detection," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 6154–6162, 2018.
[24] Guo, H. Mao, Y. Zhang and R. Chong, "Mixup as locally linear out-of-manifold regularization," *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 33, no. 1, pp. 3714–3722, 2019.