

# Towards Generalizable Crop-Agnostic Plant Disease Recognition: An Unsupervised-Learning Approach

Dongmin Choi<sup>1</sup> and Blessina Devakirubai<sup>#</sup>

<sup>1</sup>Chadwick International

<sup>#</sup>Advisor

## ABSTRACT

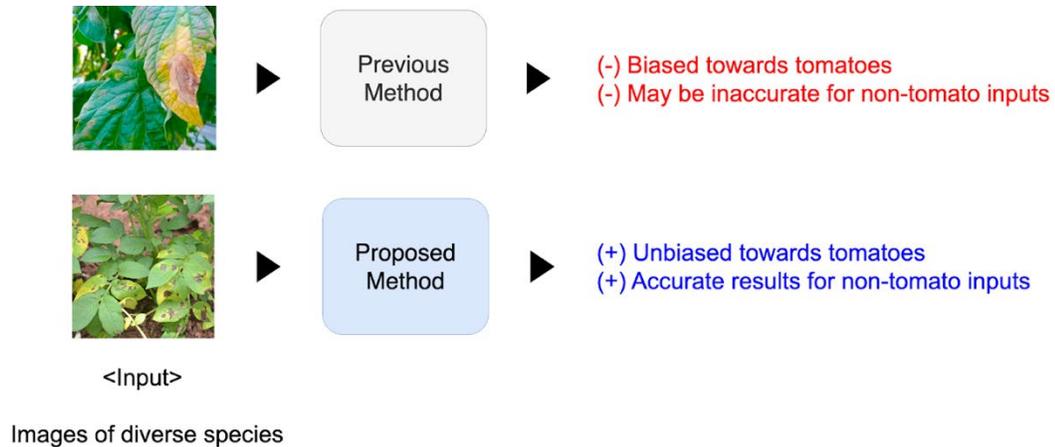
In recent years, machine learning techniques have been incorporated in the agricultural industry to automate plant disease recognition. However, most existing frameworks are constrained to the recognition of certain species' diseases, such as those of tomatoes, due to the severe imbalance in the published dataset. These methods tend to exhibit a strong bias towards specific plants, which require extensive retraining if the model were to accurately classify the disease of other plants. To ensure the system functions in a more realistic context, where images of various plants would be given, it is crucial to develop a generalized system that can recognize a diverse set of plants. To address the aforementioned problem, in this research paper, I propose a novel crop-agnostic plant detection framework. The method leverages contrastive learning, a type of unsupervised learning, to extract discriminative features from plant images regardless of their species. Moreover, the generalizable solution is capable of successfully distinguishing diseases from a dataset with imbalanced class and category. The method achieved an accuracy of 85.88% in the plant village dataset. Through experimental results, it has been demonstrated that the proposed method outperforms existing state-of-the-art methods by a significant margin.

## 1. Introduction

Plant disease detection is a machine learning algorithm sketched for computer vision tasks, particularly to classify the input images into various plant diseases. The research of plant disease detection framework is significant because it may potentially contribute to the preservation of plant's productivity and, more importantly, food security. Along with environmental factors, such as the weather, communicable disease is another factor that influences crop yield. Studying methods to prevent the spread of such diseases in advance will maximize the harvest. Moreover, the research helps protect food security. Note that the drastic change in the demographics of farmers has been observed in the last 10 years. Today, the laboring population in the agricultural industry of Korea is primarily composed of elders over 60 and foreign workers, resulting in 60% of the total farm households experiencing a workforce shortage. With problems like aging of farmers come to issue, the research into plant disease detection will genuinely mitigate issues found in the status quo.

In order to solve these problems, there have been numerous attempts to devise a detection system to automatically classify unhealthy plants from the healthy ones. To begin with, Tm et al. proposed using a modified version of LeNet5 (Tm et al. 2018), achieving an accuracy of 95% on plant village dataset; R. et al. presented ResNet (Karthik et al. 2020), recognizing diseases with a 98% accuracy. The two discoveries were incidents that demonstrated the feasibility of utilizing CNN (Convolutional Neural Network) to classify pathogens. Subsequently, Abbas et al. advanced a few more steps, producing a system in which the artificial intelligence is fed with training samples generated by itself using the C-GAN (Conditional-Generative Adversarial Network) (Abbs et al. 2021) method. While such research focused merely on diagnosing the disease through classification process, Fuentes et al attempted using objection detection to specify the affected portion of the plant.

However, the trained network for methods mentioned previously is geared particularly to suit the classification of tomato diseases, meaning it is confined to the detection of certain plants. Consequently, the network proved to be ineffectual in recognizing a wide range of plants. Hence, they could hardly be implemented in a real-world situation as shown in figure 1.



**Figure 1.** Difference between previous and proposed methods.

Therefore, in this paper, I propose a novel crop-agnostic plant disease framework in order to enable the classification of diseases for all types of plants. The system will be primarily composed of two steps: unsupervised learning and supervised learning. An unsupervised approach exploits contrastive learning to train the CNN, so it can capture robust visual representations despite the prevalence of tomato images in published datasets. In the supervised learning, the preprocessed CNN will be further trained to distinguish between various plant diseases. Ultimately, when an image of a plant is given, the system will return whether the crop is infected, and if so, the specific name of the disease will be displayed.

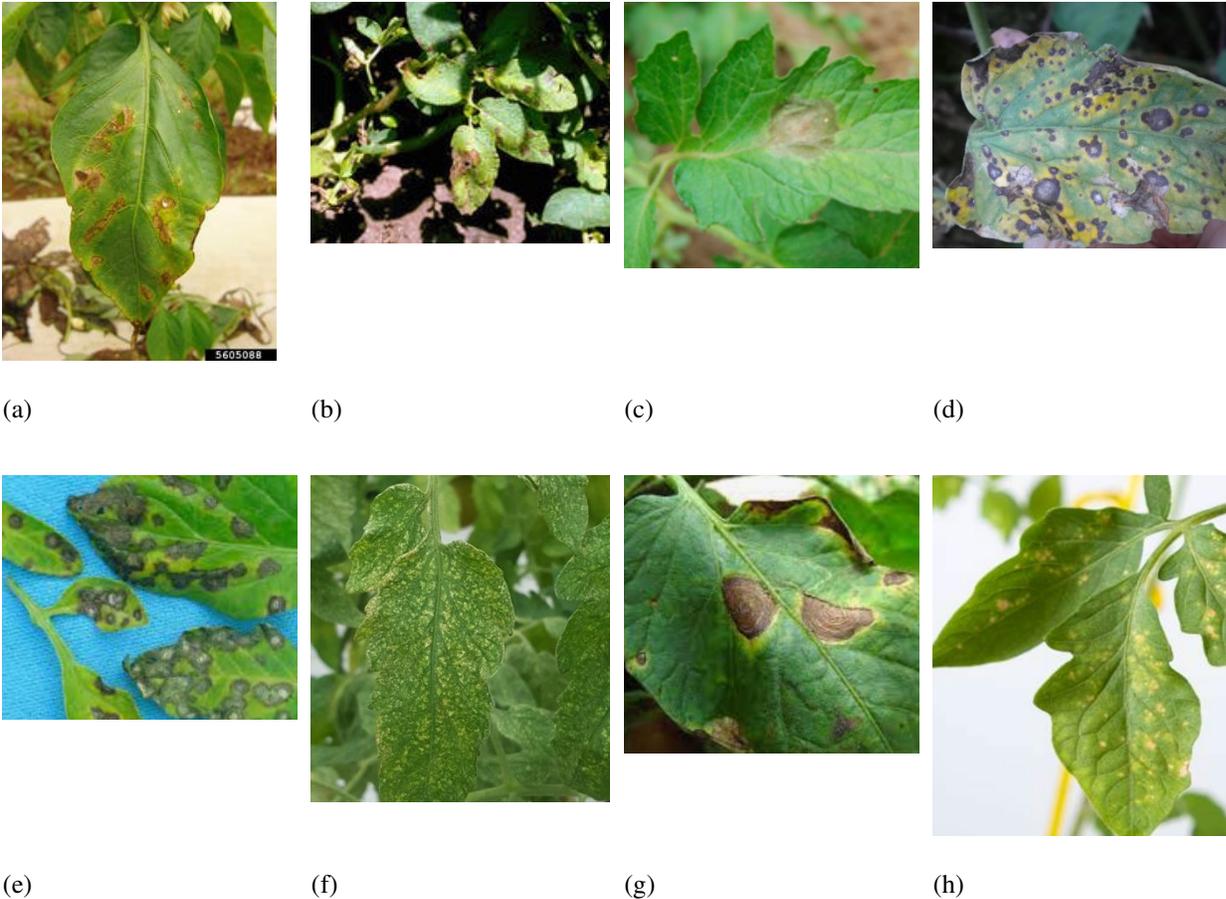
The contributions of this paper are as follows. To the best of my knowledge, I propose the first generalizable crop-agnostic disease recognition framework. I propose utilizing contrastive learning to prevent the feature extractor from being biased against certain species. I prove the superiority of the proposed methods over conventional frameworks through various systematic experiments.

Chapter 2 presents the related works of this research. Chapter 3 delves deeper into the proposed method. Chapter 4 features the results of different experiments. Lastly, Chapter 5 remarks a conclusion.

## 2. Related Work

### 2.1 Plant Disease

Plant diseases are disorders produced by pathogens, ranging from fungus to bacteria, that impose detrimental effects on the health of plants (Campbell et al. 1990). The following is a summary providing a holistic view of visual characteristics of each plant disease category. To begin with, Bacteria spot is a disease caused by different strains of bacteria, namely the *Xanthomonas Vesicatoria* (Jones et al. 2010).



**Figure 2.** Well-known plant diseases categories and their visual characteristics.  
 (a): Bacteria Spot, (b): Early Blight, (c): Late Blight, (d): Leaf Mold  
 (e): Septoria Leaf Spot, (f): Two-spotted spider mite, (g): Target Spot, and (h): Mosaic Virus

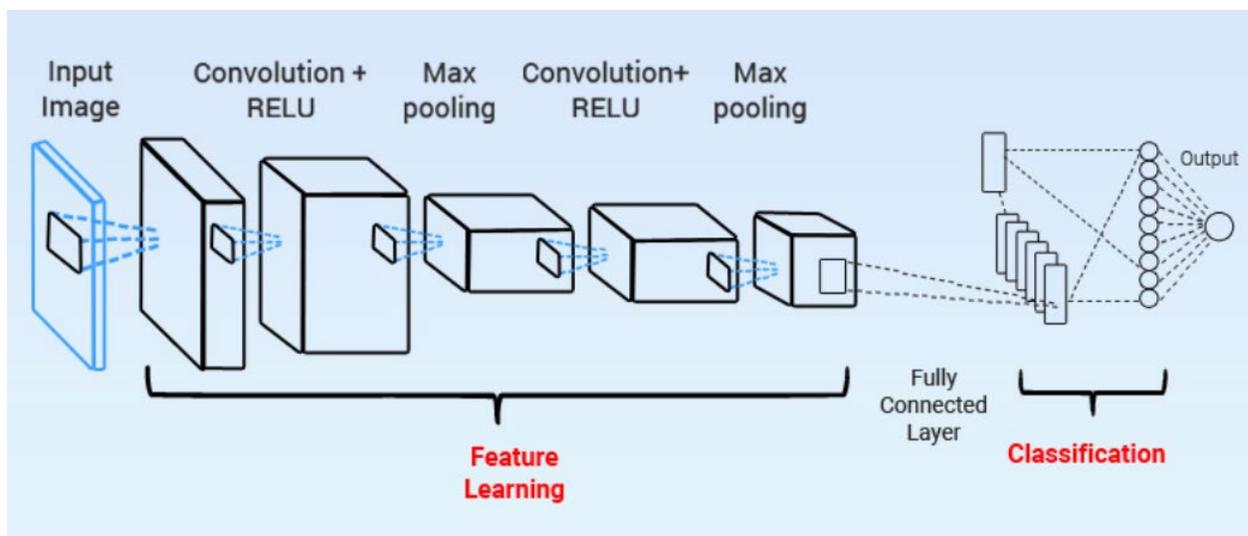
As shown in figure 2, the infected leaves commonly develop brown circular spots encompassed by a yellow halo. At the same time, small holes take place at the center of the leaves. Next, Early blight is a disease produced by a fungal pathogen named *Alternaria Solani* (Chaerani et al. 2006). The disease affects mostly the lower leaves, but it can also infect the stems and fruits. The small, dark spots developed at the initial stage progressively grow and turn yellow. Third, the Late blight is developed because of a water mold named *Phytophthora Infestan* (Arora et al. 2014). The disease initially changes portions of leaves into a pale-green, but at a later stage, the infected area becomes darker and develops an oily texture. Furthermore, the Leaf mold is caused by a fungus named *Passalora Fulva*, and it is only pathogenic on tomatoes. The disease creates dark, encysted resting structures on the plant debris, leading to the potential development of new spores. Next, Septoria leaf spot is transmitted from the fungus *Septoria Lycopersici*, and it emerges primarily when plants are setting fruit on the older, lower leaves and stems. Little, round, wet patches that measure 1.5 mm to 3 mm in diameter represent the initial symptoms. Subsequently, the centers of those dots change from gray to tan, with a dark-brown border. Moreover, Target spot is caused by fungal pathogen *Corynespora Cassicola*. Some of the spots on the plants enlarge up to 10mm and show characteristics of rings. In addition, Leaf spots turn from yellow or light green to black or dark brown. Next is Mosaic virus, produced by the potyvirus. As the name suggests, it can be identified through a mosaic pattern on the leaves. Lastly, Yellow leaf curl virus is transmitted from the Geminivirus. The chlorotic leaves roll upwards and bend downwards compared to ordinary leaves. On the other

hand, Two-spotted spider mite is a pest that consumes parts of the leaves of a tomato plant. The mites erode the plant itself, producing tiny white and yellow spots that cause the leaves to have a blotched and dotted appearance. As a result, the grown plants may develop contorted leaves and flowers.

As mentioned earlier, plant diseases frequently exhibit distinct and recognizable visual patterns. The main objective of this research is to develop a systemic plant disease classifier based on their specific visual characteristics.

## 2.2 Image Classification

Image classification is a crucial computer vision task that has garnered considerable interest in the research field due to its abundant real-world application possibilities. Image classification systems are generally developed using CNN (Convolutional Neural Network) architectures. The process involves training a CNN model on a large dataset of labeled images to learn and extract meaningful features from the images. These learned features are then used to make predictions on new, unseen images.



**Figure 3.** Example of image classification network (Wang et al. 2017)

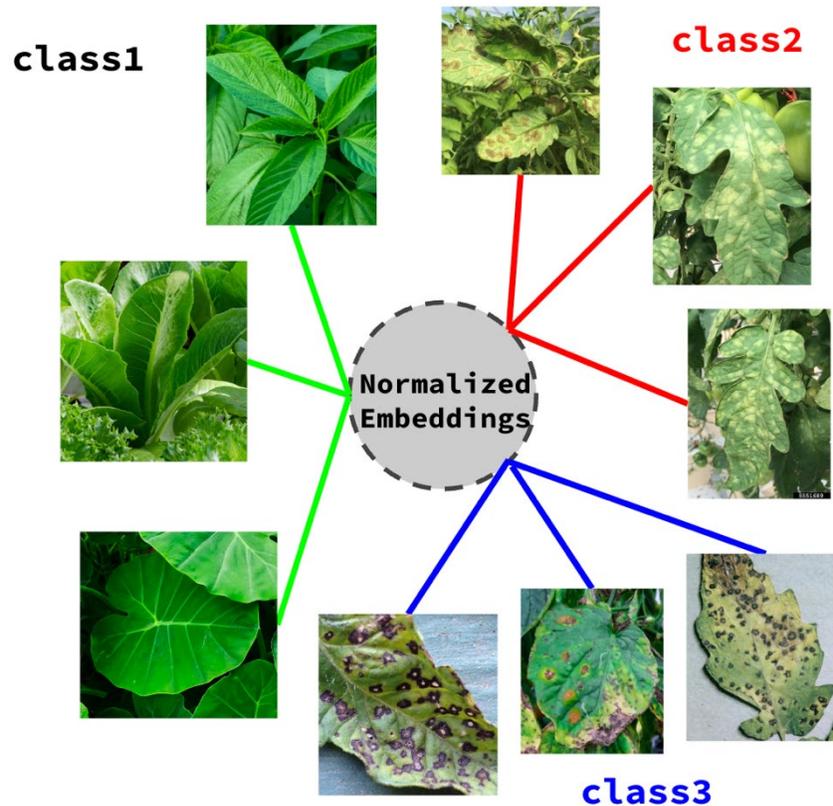
Figure 3 illustrates an example of the architecture of a CNN. The network takes digital images as input and extracts significant features, which are then employed by the classifier to determine the corresponding category or class of the input image. There are numerous practical applications of image classification, such as object recognition in autonomous vehicles, medical diagnosis, facial recognition in security systems, content-based image retrieval in search engines, and quality control in manufacturing processes.

In this research, the plant disease detection system is regarded as an image classification approach, as the proposed method's input and output align with those of a typical image classification system. The detailed process of the proposed method, including its underlying assumptions and implementation, will be thoroughly presented in Chapter 3.

## 2.3 Contrastive Learning

Contrastive learning is a type of unsupervised learning that lays the essential groundwork for maintaining an accurate recognition rate. More specifically, its objective is to train the machine learning models to capture features that are

only relevant to the task at the moment by contrasting similar and dissimilar samples. This learning metric is frequently used for computer vision tasks, especially for its ability to pinpoint the most useful representations of input data, including images and videos.



**Figure 4.** Example of contrastive learning

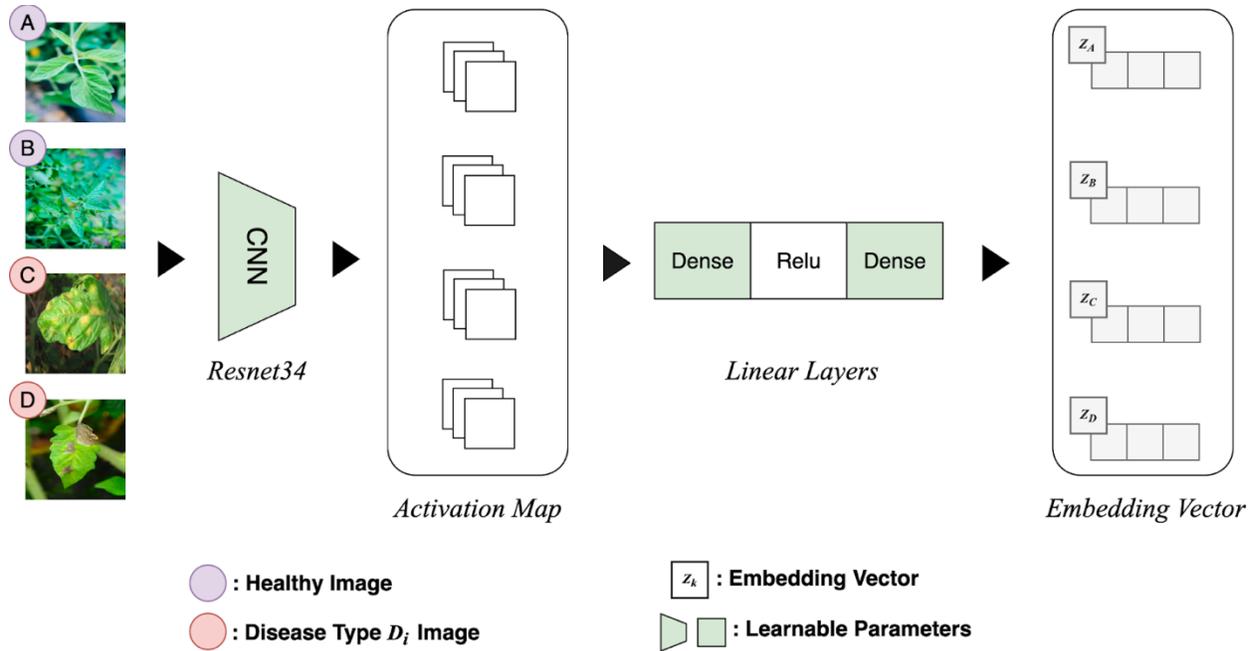
Figure 4 illustrates the concept of contrastive learning applied to the plant disease dataset. In this context, machine learning models are trained to extract similar features or representations for the same plant disease categories. The ultimate objective of this process is to cluster the features or representations of the same category in the feature space, which aids in enhancing accuracy during classification.

This paper focuses on researching and discovering how contrastive learning can effectively address class and category imbalance issues in the plant disease dataset. The mathematical reasoning and approach will be elucidated in Chapter 3, while the effectiveness of the proposed method will be evaluated through extensive experiments in Chapter 4.

### 3. Proposed Method

The proposed plant disease recognition system consists of two modules: a contrastive learning module that learns to extract similar features or representations from the same disease category, and a transfer learning module that focuses on classifying the target plant disease. The ultimate goal of the proposed methodology is to enable the CNN (Convolutional Neural Network) to produce similar activation maps upon receiving plant images with a comparable health status. The following measure will essentially minimize the possibility of forming a biased weight against particular plant species, contributing to the development of a crop-agnostic plant disease framework.

Considering the significance of tomatoes within global agricultural sectors, the disease detection system for this plant was readily accessible in the status quo. Yet, realistically, there are various species out in the market, and henceforth the recognition must not be confined to the domain of one particular plant. Incentivized by such findings, I utilized the previous method to detect the diseases of plants other than tomatoes and discovered a remarkable gap between the accuracy rate compared to when detecting the image of tomatoes alone. As the deviation stemmed from the biased weight, I decided to use contrastive learning to address this issue. Initial frameworks consist solely of supervised learning. In the proposed method, the training of AI (Artificial Intelligence) comprises two phases: phase 1 is an unsupervised learning, while phase 2 is supervised.



**Figure 5.** Architecture of Phase 1 in the proposed framework

**Table 1.** Notations used in this paper

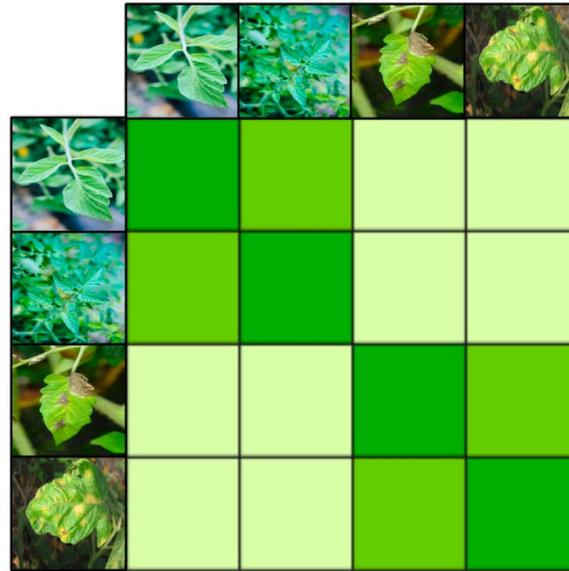
Original	Abbreviated
Input Plant Image	$I$
Feature Extractor	$FE(.)$
Activation Map	$A$
Linear Layers	$LL(.)$
Embedding	$E$
Plant Disease Classifier	$PDC(.)$
Classification Result	$CR$
ReLU Function	$RL(.)$

### 3.1 Unsupervised Learning

In this section, I will explain the detailed pipeline of phase 1 in the proposed framework. First, the proposed Feature Extractor receives the input image  $I \in R^{m \times n}$  to produce the activation map  $A$  that captures the visual characteristics of

the plant diseases. Here,  $H$  and  $W$  denote the height and width of the plant image. Hence, the Feature Extractor is defined as the function  $FE: I \rightarrow A$ .

Subsequently, the activation maps are converted into feature layers, also known as the embedding. For instance, the activation maps of images representative of diverse species are flattened into an respective embedding  $E \in \mathbb{R}$  through Linear Layers  $LL$ . Similarly, the Linear Layer is defined as the function  $LL: A \rightarrow E$ .



**Figure 6.** Similarity Index

The objective is to maximize the similarity index of two embedding vectors from a comparable health status. Here, the term “similarity index” refers to how similar two embedding vectors are. Figure 6 illustrates an exemplar matrix indicating respective similarity indices. The darker the box is, the more similar each vector must be. The indices are calculated using cosine similarity as shown in Equation 1.

**Equation 1:** Cosine Similarity Function

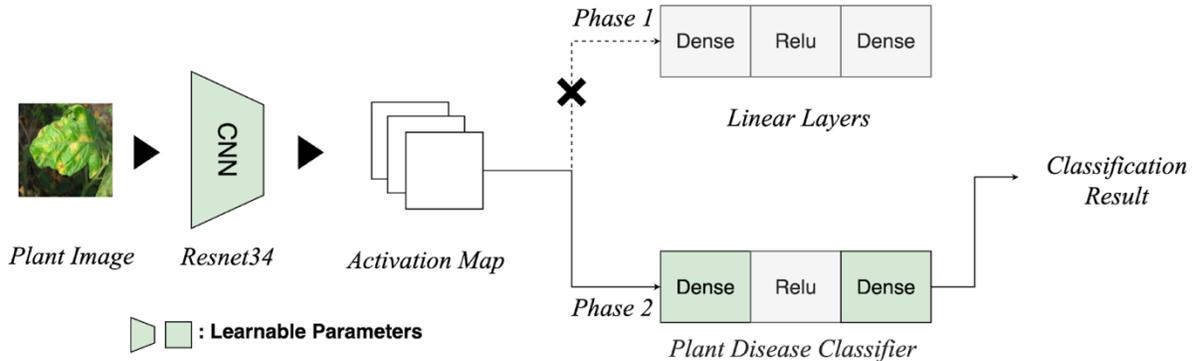
$$s_{i,j} = \frac{z_i^T z_j}{(\tau ||z_i|| ||z_j||)}$$

Here, the numerator is the dot product of two vectors,  $z_i$  and  $z_j$ , and the denominator is their norm. The range of the function is -1 to 1. In the best-case scenario, wherein the two vectors are identical,  $S_{ij}$  would equal 1. On the contrary, if the two vectors are directly opposite,  $S_{ij}$  would equal -1, equivalent to cosine 180 degree.

Then, the NT-Xent loss (Chen et al. 2020) function is employed to train the proposed framework. First, each index is converted into a percentage using the softmax function. Second, the cross-entropy loss function is utilized to present the loss for individual images. Lastly, the average loss of pairs of images with a similar health status is calculated. Hence, it is optimal to keep the number of images as the power of 2. Through iteration of the following process, the CNN is set to have the minimized loss, marking the end of phase.

### 3.2 Supervised Learning (fine-tuning)

In this chapter, the focus will be on providing a comprehensive explanation of the second phase of the proposed framework. Phase 2, or the supervised learning, aims to fine-tune the CNN by feeding an assorted set of plant images.



**Figure 7.** Architecture of Phase 2 in the proposed framework

At this point, the CNN would have been optimized for classifying the images, largely due to the effects of unsupervised learning in the previous phase. Firstly, the feature extractor produces an activation map, as defined by the function  $FE: I \rightarrow A$ . Then, the generated activation map is flattened into a single-column matrix containing pixel information of input image  $I$ . The scores for each category would be calculated by multiplying the matrix with the weight and adding it with the bias. Yet, these scores don't intuitively show the degree to which the image fits into the ten categories; they need to be converted to percentage. Prior to this process, the scores are filtered through the Relu function, defined as  $ReLU(z) = \max(0, z)$ , which alters every negative value to zero. The function is part of an optimization process that enables faster training of the AI. Subsequently, the softmax function will retrieve respective percentages by dividing individual values by the scores all added. Relying on this probability, the loss value is calculated to effectively lead to AI producing a more accurate result. In the cross-entropy loss function, the probability of each category is inputted into a (Equation 3). This way, the closer the probability is to 1, the smaller loss value it will get assigned. On the other hand, an image with probability close to 0 will get assigned an infinite size of loss value, indicating a need for modification in the weight value.

**Equation 2:** Softmax Function

$$\sigma(\vec{z})_i = \frac{e^{z_i}}{\sum_{j=1}^K e^{z_j}}$$

where,  $e^z$  denotes the individual score values after the Relu function is operated.

**Equation 3:** Cross-entropy Loss Function

$$L = -\log(pr)$$

Here,  $pr$  denotes the percentage returned from the softmax function. Eventually, the system produces scores for categories of each disease, allowing the recognition of plant diseases. This holistic process is simplified into the function  $PDC: A \rightarrow Pr$ . The effectiveness of the fine-tuning is explained in detail in chapter 4.

### 3.3 Implementation Details

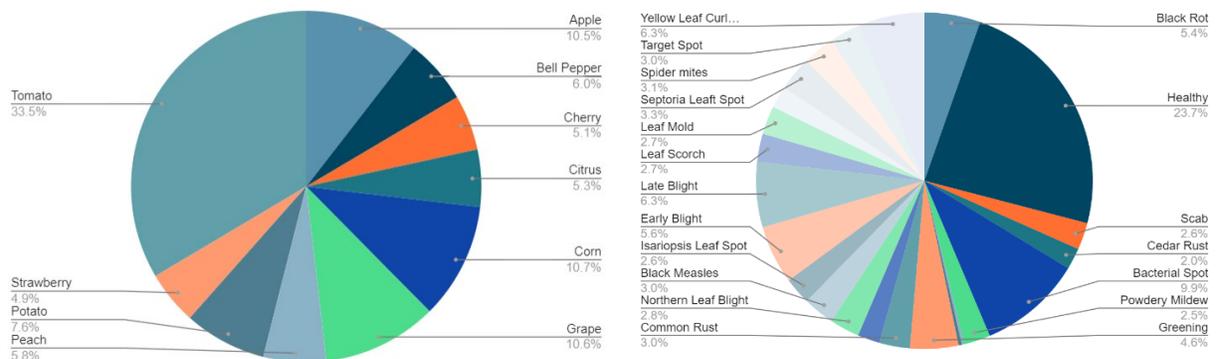
The following information details out the variables for the proposed method during respective experiments. For unsupervised learning, the resnet34 (He et al. 2016) architecture is used. An individual batch contains 1024 images, and the training lasted for 100 epochs. The initial learning rate is 0.0001, and the decay is at 40 and 80 by 1/10, respectively. Lastly, the Adam optimizer (Kingma et al. 2014) is employed. For supervised learning, the same architecture is used. The batch size remains the same as 1024, but the number of epochs is reduced by half to 50. The initial learning rate is also the same as 0.0001, but the decay is at 40 by 1/10. Lastly, the Adam optimizer is used, and the augmentation includes horizontal flip and color jitter.

## 4. Results

Several experiments are conducted to demonstrate that the proposed method has more benefits than the system used commonly today. In this chapter, the systematic protocols for such experiments are revealed.

### 4.1 Dataset

All necessary images are sourced from the Plant Village Dataset (Singh et al. 2020), created by David Hughes of Penn State University. The dataset comprises numerous photographs of healthy and unhealthy leaves, each labeled with its species and corresponding disease name. This organization, hence, makes the dataset a popular choice among users seeking tasks related to the recognition and diagnosis of plant diseases through machine learning. The dataset maintains a diverse selection by incorporating 54,000 images of 26 plants, and the images could be downloaded from the Plant Village website for free. Out of total images, 80% were used for training, and 20% were used for testing.



**Figure 8.** Dataset distribution of Plant Village Dataset (Singh et al. 2020)

Nevertheless, it is critical to address the two imbalances within the dataset. Firstly, images of tomatoes outnumber those of the remaining 25 crops. Indeed, tomato images alone make up one-third of the entire dataset. This imbalance will cause the feature extractor to develop a bias on tomato images, which will in turn decrease the accuracy when the input is a crop other than a tomato. Second, the images of healthy leaves outnumber the rest of the categories. An ideal dataset would contain an equally distributed number of images for each category because, again, the presence of a dominant input merely increases the probability of generating a bias. Despite these issues, the proposed framework is expected to produce a robust classification of plant diseases regardless of the data set's quality.

## 4.2 Comparison with State-of-the-art Methods

In order to demonstrate the superiority of the proposed framework, four distinct experiments are conducted.

**Table 2.** Comparison of Accuracy with Existing State-of-the-art Methods

Method	Experiment-1	Experiment-2	Experiment-3	Experiment-4
VGG based (Ferentinos et al. 2018)	0.7274	0.8001	0.7087	0.7217
Resnet based (Karthik et al. 2019)	0.7680	0.8448	0.6879	0.7306
DenseNet based (Kim et al. 2021)	0.7818	0.8912	0.6604	0.6991
Ours	<b>0.8588</b>	<b>0.8914</b>	<b>0.8185</b>	<b>0.8390</b>

### 4.2.1 Comparison on Entire Samples

Experiment 1 aims to compare the performance of the proposed method with that of the state-of-the-art methods when the test set comprises images selected from a wide range of species. The experiment set includes photographs of healthy and unhealthy plants, ranging from tomato, apple, potato, to corn, to verify how the introduction of contrastive learning contributes to the enhancement of the framework. As established in table 2, the proposed framework achieved an accuracy of 0.8588, whereas the other reached an average accuracy of 0.7591. Contrastive learning approach enforces the trained feature extractor to extract more robust features by preventing the CNN (Convolutional Neural Network) from developing a bias on tomato images. As aforementioned, the functionalities of traditional disease detectors are limited to the scope of tomato diseases, thus yielding dissatisfactory results when the detector is used with other types of plants. The contrastive learning method keeps the CNN away from highlighting the features of tomato leaves but strives to generate more robust features for all species, taking a step closer to the crop-agnostic disease classification.

### 4.2.2 Comparison on Tomato Samples

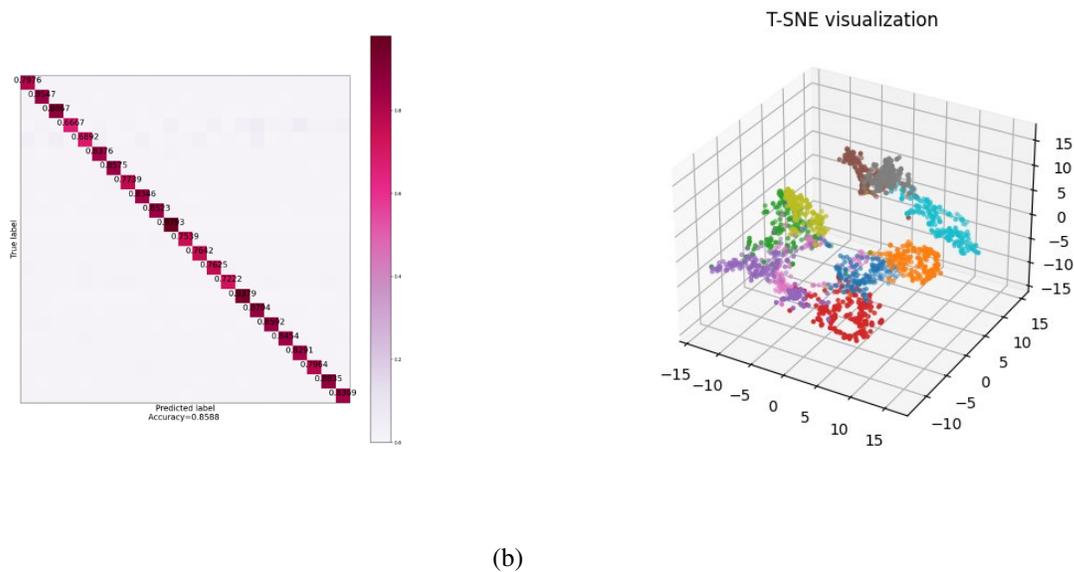
Experiment 2 incorporates a test set composed only of tomato images. The ultimate goal of this test is to demonstrate that the proposed method is capable of producing accurate results even after the introduction of contrastive learning. Similar to previous experiments, images of tomato leaves - both healthy and unhealthy - were used, and the accuracy of classification was measured. The results in table 2 presents that the proposed framework functions without problems, reaching 0.8914 in accuracy. Note that this accuracy is the highest among the four experiments.

### 4.2.3 Comparison of Crops with Small Proportions in Dataset

Experiment 3 aims to compare the performance of the proposed method with that of the state-of-the-art methods under circumstances in which a small-sized data set was given as an input. While the feature extractor from three previous tests were fed with approximately 99,700 images, the input of experiment 3 contained 27,000 images. As shown in Table 2, the proposed framework reached an accuracy of 0.8185, while the SOTA achieved up to 0.7087, demonstrating its ability to classify plant disease with a relatively small number of images. This success is attributable to the nature of contrastive learning. In the agricultural industry, obtaining large amounts of labeled data is a burdensome task, particularly because the types of diseases vary. The use of unlabeled data, however, makes the extraction of meaningful representations of images possible, which is hindered in the domain of supervised learning when the size of the data set is small. Another notable insight comes from the consistency of the framework. Compared to the result of experiment 1, which was acquired using the entire data set, the state-of-the-art methods showed an average decrease of 0.0734 in their accuracies. Conversely, the proposed method showed a smaller decrease of 0.0403. These results demonstrate that the proposed method can persistently produce accurate classifications despite a deficit in data.

### 4.2.4 Comparison of Other Crops (Excluding Tomato Samples)

The objective of the last experiment is to verify the boosted performance of the proposed method compared to that of the state-of-the-art methods when the test set did not include tomato images. Instead, the data set comprises images of healthy and unhealthy leaves from apple, potato, and corn, plants that carry the same diseases as tomatoes. In this case, tomato images are displaced intentionally to eliminate the possibility of the feature extractor developing a biased weight, which poses an obstacle to achieving high accuracy. The result for experiment 4 is 0.8390, whereas the rest featured an average accuracy of 0.7171. Thanks to contrastive learning, the desired goal of being crop-agnostic is observable from the consistent accuracies of classification. The state-of-the-art methods, on the other hand, show a fluctuation of results depending on the condition of datasets.



**Figure 9.** Confusion Matrix and t-SNE (Van der Maaten et al. 2008) evaluation of the proposed method (a): Confusion matrix, and (b) T-SNE evaluation

Figure. 9 (a) is the confusion matrix for each experiment. A confusion matrix is a visualized table that is frequently used to evaluate a machine learning model’s accuracy. Each row and column represent one of the ten categories of plant diseases, and the color shows the frequency of the predictions made for each category. The darker the diagonal lines get, the more accurate the framework is.

Figure. 9 (b) is the t-SNE (Van der Maaten et al. 2008) visualization of the proposed method. Recall that the primary goal of contrastive learning is to make the activation maps for images with a similar health status look alike. For instance, the activation maps for plants infected with early brights must look similar to one another. T-SNE visualizes this relationship between inputs by simplifying the multi-dimensional activation maps and plotting them into a three-dimensional plane. The extent to which the contrastive learning is successful then could be evaluated by observing how clustered each point with the same color is. As shown in the same diagram, the points are well-clustered, representing the success of our framework.

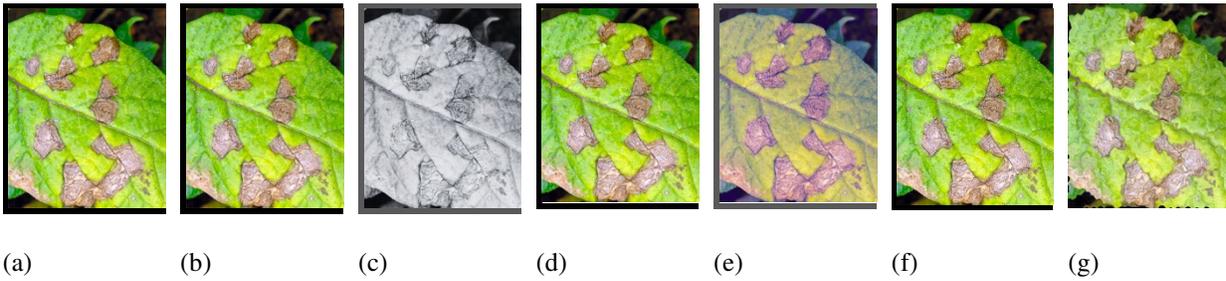
### 4.3 Architecture Replacement Experiment

**Table 3.** Accuracy before and after training with the proposed method

Architecture	Accuracy		Accuracy Increment
	Before	After	
VGG 19 (Szegedy et al. 2015)	0.7274	0.7785	+5.1%
MobileNetV2 (Sandler et al. 2018)	0.7582	0.8198	+6.1%
Xception (Chollet et al. 2017)	0.7801	0.8674	+8.7%
Resnet34 (He et al. 2016)	0.7680	0.8588	+9.0%

As previously introduced, the architecture used throughout the experiment is Resnet34 (He et al. 2016). To ensure the proposed method is not limited to certain architectures, a follow-up experiment is conducted to demonstrate the framework’s generalizable qualities. As featured in Table 3, three additional architectures are tested. In a controlled situation, wherein all models are trained using supervised learning, VGG19 (Szegedy et al. 2015), Xception (Chollet et al. 2017), and MobileNetV2 (Sandler et al. 2018), Resnet34 classifies plant diseases with an accuracy of 0.7274, 0.7801, 0.7582, and 0.7680, respectively. However, after training each architecture using the proposed method, VGG19 reaches 0.7785, a +5.1% increase; Xception reaches 0.8674, an +8.7% increase; MobileNetV2 reaches 0.7582, a +6.1% increase; and Resnet reaches 0.8588, a +9% increase. The result shows that the proposed method is capable of producing a robust classification independent of the type of architecture used. This finding verifies the claim that the proposed method can pave the path for a more powerful generalizable framework in the future, thanks to its apparent consistency even when it is paired with different architectures.

#### 4.4 Data Augmentation



**Figure 10.** Data Augmentation

(a): Original Image, (b) Rotation, (c): Grayscale, (d): Flip, (e): Color Jitter, (f): Crop, and (g): Elastic Transform

The aim of this ablation study is to collect data about how different types of data augmentation methods affect the accuracy of classification. As shown in Figure 10, six types of augmentations are implemented: rotation, grayscale, flip, color jitter, crop, and elastic transform. Table 4 shows the accuracies relative to the baseline model, in which the architecture is only trained using the proposed method without any augmentation in input data. To begin with, the control model presents an accuracy of 0.8291. Out of six methods, color jitter and flip facilitate enhancement: for color jitter, the accuracy improves by +1.6%, reaching 0.8455, and for flip, the accuracy improves by +2.3%, reaching 0.8314. However, the rest shows a negative correlation. Applying rotation reduced the accuracy by 1.3%; crop by 0.2%; elastic transform by 14%; and grayscale by 4.9%. Hence, only the two methods, color jitter and flip, are selected and tested again to calculate the accuracy. Ultimately, the accuracy improved by 3.0%, verifying the significance of aforementioned augmentation methods.

**Table 4.** Accuracy Comparison on Various Data Augmentation

Method	Accuracy	Accuracy Increment
Baseline	0.8291	-
Color Jitter	0.8455	+1.6%
Rotation	0.8163	-1.3%
Crop	0.8274	-0.2%
Elastic Transform	0.6891	-14.0%
Flip	0.8314	+2.3%
Grayscale	0.7796	-4.9%
Color Jitter + Flip	0.8588	+3.0%

## 4.5 Transfer Learning

**Table 5.** Accuracy Result on Transfer Learning

Architecture	Accuracy		Accuracy Increment
	From Scratch	Transfer Learning	
VGG 19 (Szegedy et al. 2015)	0.8082	0.8304	+2.2%
MobileNetV2 (Sandler et al. 2018)	0.8219	0.8492	+2.8%
Xception (Chollet et al. 2017)	0.8941	0.9323	+3.8%
Resnet34 (He et al. 2016)	0.9085	0.9476	+3.9%

The last experiment illustrates the application of the proposed method. In this experiment, the accuracy of each model classifying rice leaf diseases when it is trained from the scratch and trained using transfer learning is measured. For the latter option, the pre-trained CNN using the proposed method is used. As shown in Table 5, using transfer learning produces an approximately 3.18% higher accurate classifications than learning from scratch. This is because the CNN pre-trained using the proposed method has embedding vectors of the same category (plant disease type) clustered together, capturing meaningful representations, as previously explained in the T-SNE visualization. Therefore, transfer learning produces more accurate results than learning from scratch. The following finding proves that the proposed method could be applied in various tasks to enhance the accuracy of models.

## 5. Conclusion

In this research paper, I investigated the effects of contrastive learning on improving the accuracy of plant disease recognition. I proposed a novel framework that enables the generalization of visual representations of different plant species to produce robust extractions of plant diseases. The proposed approach outperformed conventional supervised learning methods and achieved state-of-the-art performance in accurately identifying and differentiating between healthy and diseased plants. These findings are valuable insights to future implications of contrastive learning, and further research in exploring alternative self-supervised learning techniques, more diverse datasets, and the transferability of acquired visual characteristics across various crops, will greatly benefit the field of plant disease recognition. In conclusion, the proposed research contributes to the development of more accurate, robust, and widely applicable disease recognition models, thereby revolutionizing agricultural practices and positively impacting global food production.

## References

- Abbas, A., Jain, S., Gour, M., & Vankudothu, S. (2021). Tomato plant disease detection using transfer learning with C-GAN synthetic images. *Computers and Electronics in Agriculture*, 187, 106279.  
<https://doi.org/10.1016/j.compag.2021.106279>
- Arora, R. K., Sharma, S., & Singh, B. P. (2014). Late blight disease of potato and its management. *Potato Journal*, 41(1).
- Campbell, C. L., & Madden, L. V. (1990). *Introduction to plant disease epidemiology*. John Wiley & Sons..

- Chaerani, R., & Voorrips, R. E. (2006). Tomato early blight (*Alternaria solani*): the pathogen, genetics, and breeding for resistance. *Journal of general plant pathology*, 72, 335-347. <https://doi.org/10.1007/s10327-006-0299-3>
- Chen, T., Kornblith, S., Norouzi, M., & Hinton, G. (2020, November). A simple framework for contrastive learning of visual representations. In *International conference on machine learning* (pp. 1597-1607). PMLR. <https://doi.org/10.48550/arXiv.2002.05709>
- Chollet, F. (2017). Xception: Deep learning with depthwise separable convolutions. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 1251-1258). <https://doi.org/10.48550/arXiv.1610.02357>
- Ferentinos, K. P. (2018). Deep learning models for plant disease detection and diagnosis. *Computers and electronics in agriculture*, 145, 311-318. <https://doi.org/10.1016/j.compag.2018.01.009>
- He, K., Zhang, X., Ren, S., & Sun, J. (2016). Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 770-778). <https://doi.org/10.48550/arXiv.1512.03385>
- Jones, C. D., Jones, J. B., & Lee, W. S. (2010). Diagnosis of bacterial spot of tomato using spectral signatures. *Computers and electronics in agriculture*, 74(2), 329-335. <https://doi.org/10.1016/j.compag.2010.09.008>
- Karthik, R., Hariharan, M., Anand, S., Mathikshara, P., Johnson, A., & Menaka, R. (2020). Attention embedded residual CNN for disease detection in tomato leaves. *Applied Soft Computing*, 86, 105933. <https://doi.org/10.1016/j.asoc.2019.105933>
- Kim, S. K., & Ahn, J. G. (2021). Tomato crop diseases classification models using deep CNN-based architectures. *Journal of the Korea Academia-Industrial cooperation Society*, 22(5), 7-14.
- Kingma, D. P., & Ba, J. (2014). Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*. <https://doi.org/10.48550/arXiv.1412.6980>
- Sandler, M., Howard, A., Zhu, M., Zhmoginov, A., & Chen, L. C. (2018). Mobilenetv2: Inverted residuals and linear bottlenecks. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 4510-4520). <https://doi.org/10.48550/arXiv.1801.04381>
- Singh, D., Jain, N., Jain, P., Kayal, P., Kumawat, S., & Batra, N. (2020). PlantDoc: A dataset for visual plant disease detection. In *Proceedings of the 7th ACM IKDD CoDS and 25th COMAD* (pp. 249-253). <https://doi.org/10.1145/3371158.3371196>
- Szegedy, C., Liu, W., Jia, Y., Sermanet, P., Reed, S., Anguelov, D., ... & Rabinovich, A. (2015). Going deeper with convolutions. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 1-9). <https://doi.org/10.48550/arXiv.1409.4842>
- Tm, P., Pranathi, A., SaiAshritha, K., Chittaragi, N. B., & Koolagudi, S. G. (2018, August). Tomato leaf disease detection using convolutional neural networks. In *2018 eleventh international conference on contemporary computing (IC3)* (pp. 1-5). IEEE. (DOI) <https://doi.org/10.1109/IC3.2018.8530532>
- Van der Maaten, L., & Hinton, G. (2008). Visualizing data using t-SNE. *Journal of machine learning research*, 9(11).
- Wang, Q., Zhou, R., & Yan, Y. (2017). A two-stage approach to note-level transcription of a specific piano. *Applied Sciences*, 7(9), 901. <https://doi.org/10.3390/app7090901>