Image Data Augmentation for Plant Leaf Disease Classification Using Neural Style Transfer

Sofian Zalouk †
szalouk@stanford.edu

Sharan Ramjee †
sramjee@stanford.edu

I. Introduction

Plant diseases devastatingly reduce the potential crop yield by an average of 40%, and up to 100% in the developing world. The detection of plant diseases remains difficult due to the lack of infrastructure and expertise. As such, several applications have been developed for diagnosis of plant diseases based on the success of deep learning techniques. However, these applications suffer from a drastic performance degradation when tested in practical settings and fail to generalize to real world data. This degradation can be attributed to the severe class imbalance (as observed in figure 1) and lack of statistical diversity of the background images in the available datasets, in particular, the PlantVillage dataset, which is the state-of-the-art for plant leaf disease classification.

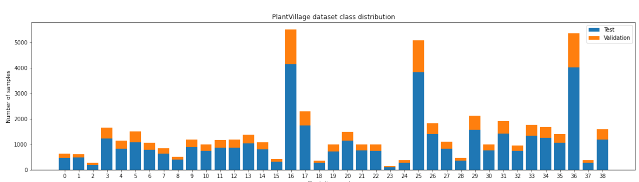


Fig. 1: Class imbalance in PlantVillage dataset

We investigate the effectiveness of various data augmentation methods in improving the generalization performance of plant disease classifiers. In particular, we propose the use of Neural Style Transfer (NST) to generate images of diseased plants from healthy plants as a data augmentation method for improving the generalization performance of plant disease diagnosis. In addition, we investigate the effectiveness of NST in comparison to various conventional and GAN-based data augmentation techniques.

The inputs to our NST data augmentation algorithm are a set of healthy leaf content images and a single diseased leaf style image, both belonging to the same plant species. Our NST data augmentation algorithm will transfer the style of the diseased leaf image to the set of healthy leaf content images. As such, the output of the algorithm is a set of newly generated diseased leaf images belonging to the same disease class and species as the diseased leaf style image.

† Equal contribution

II. Related work

A. Plant Disease Classification

Substantial prior research has been conducted on using deep learning to perform image-based diagnosis of plant diseases. Notably, [1] trained two deep learning models (AlexNet and GoogleNet) on the PlantVillage dataset and achieved a mean accuracy of 99.3%. However, their models fail to generalize well, evidenced by an accuracy of 31% when tested using a set of on-site plant images. As previously discussed, this degradation in performance can be attributed to the biases in the PlantVillage dataset. Furthermore, [2] evaluated the classification performance of different state-of-the-art deep learning models and transfer learning strategies on the PlantVillage dataset. Most interestingly, they found that deep training (fine-tuning all the layers) of pre-trained networks yielded the best performance results compared to shallow training (fine-tuning only the fully-connected layers). Following their findings, we used deep training for all of our classification models.

B. Plant Disease Data Augmentation

In the specific application of image-based plant disease diagnosis, several papers have proposed successful dataset augmenting methods for improving performance and generalization. In particular, [3] made use of GANs and NST to augment the PlantVillage dataset. However, the effectiveness of their augmentation methods was evaluated on the same problematic PlantVillage dataset, which is not a good indicator of generalization performance. The work done by [4] is considered the state-of-the-art for the problem of plant disease classification as [4] used the latest StyleGAN to augment the PlantVillage dataset and reported significant improvements in generalization performance on a separate, private, plant disease dataset, as evaluated by the state-of-the-art plant disease classification models. However, to the best of our knowledge, no prior work exists that has investigated the impact of NST-based dataset augmentation on the generalization performance of plant disease classifiers.

C. Attention Guided GANs

[5] proposed a novel LeafGAN model for augmenting plant disease datasets using an attention mechanic, where their model generated segmentation masks of the leaf that are used to guide GAN to only transfer the disease to the leaf regions. Their model significantly improved generalization

performance by solving the known ”bleeding” problem with GANs where the plant disease is applied to the background of the image. Most notably, [5] showed that using a traditional CycleGAN without an attention mechanism for data augmentation would not improve generalization performance due to this ”bleeding” problem.

III. Dataset and Features

A. PlantVillage

The primary dataset that was used for training and testing is the PlantVillage [6], which contains 55448 images with a resolution of 256×256, consisting of 38 plant-disease classes. Since the deep learning classifier models were pre-trained on ImageNet [7], each channel of the RGB images in the PlantVillage dataset were normalized using the ImageNet normalization parameters $\mu = [0.485, 0.456, 0.406]$ and $\sigma = [0.229, 0.224, 0.225]$. We found that this normalization scheme yielded the best classification performance on the baseline model. Furthermore, all images were cropped to a resolution of 224×224 to match the input size of the deep learning models. Each image in the dataset is accompanied by a label of the form $\{plant\, species, \, disease\}$ and an associated leaf segmentation map. The PlantVillage dataset suffers from a class imbalance, which causes significant degradation in the performance of trained models in practical settings.

B. Plant Pathology 2020 challenge dataset

A secondary dataset that was used for testing the generalization performance of the plant disease classifier is the PlantPathology [8] dataset, which contains 1730 images with a resolution of 256×256, consisting of 3 distinct apple-disease classes (healthy, rust, scab). Using the same reasoning given for PlantVillage, each channel of the RGB images in the PlantPathology dataset were normalized using the ImageNet normalization parameters and cropped to a resolution of 224×224 to match the input size of the deep learning models. Since the PlantPathology dataset was used exclusively to evaluate the generalization performance of the plant disease classifier, no data augmentation was performed on the dataset. The images in this dataset were captured under a diverse range of angles, lighting, and distances, making it a far more accurate measure of generalization performance than the PlantVillage dataset.

IV. Methods

The implementations of all the algorithms proposed and considered in this paper are available on GitHub[1].

A. Neural Style Transfer

Neural Style Transfer (NST) [9] uses a pre-trained model to avoid the use of paired content-style images and takes advantage of transfer learning to extract the content and style from images. Our pre-trained model of choice for NST was the VGG19 model [10] since VGG models are unable to capture non-robust features as well as other architectures, which allows VGG models to perform style transfer to produce outputs that look more correct to humans [11]. We used the output of the last block of the VGG19 network to extract the content of the healthy leaf image as the VGG19 model builds on top of previous layers to form complex feature representations of the input image. In order to obtain more robust style representations and capture multi-scale information, we obtain feature correlations from the outputs of all the blocks of the VGG19 model to obtain the gram matrix that represents the style of the diseased leaf image.

The style representation is given by the gram matrix:

$$\mathcal{G}(F^{[l]}(\overrightarrow{\mathrm{x}})) = [F^{[l]}(\overrightarrow{\mathrm{x}})][F^{[l]}(\overrightarrow{\mathrm{x}})]^\top$$

where $\overrightarrow{\mathrm{x}}$ denotes the image and $F^{[l]}(\overrightarrow{\mathrm{x}})$ denotes the output of the VGG19 network at layer l.

Our objective is to perform NST using the style of a diseased leaf image $\overrightarrow{\mathrm{d}}$ and the content of a healthy leaf image $\overrightarrow{\mathrm{h}}$ to produce a synthesized image $\overrightarrow{\mathrm{x}}$. We first initialize $\overrightarrow{\mathrm{x}}$ with noise and minimize the linear combination of the content loss and the style loss:

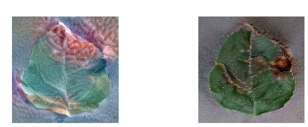
$$\mathcal{L}_{NST}(\overrightarrow{\mathrm{h}}, \overrightarrow{\mathrm{d}}, \overrightarrow{\mathrm{x}}) = \alpha \mathcal{L}_{content}(\overrightarrow{\mathrm{h}}, \overrightarrow{\mathrm{x}}) + \beta \mathcal{L}_{style}(\overrightarrow{\mathrm{d}}, \overrightarrow{\mathrm{x}})$$

where α and β are the weights of the content and style losses, respectively. The content loss $\mathcal{L}_{content}$ is the summation of content losses over the $content$ layers (last layer of the VGG19 network) and style loss $\mathcal{L}_{style}$ is the summation of the style loss over the $style$ layers (all layers of the VGG19 network):

$$\mathcal{L}_{content} = \sum_{l \in content} w^{[l]}_{content} ||F^{[l]}(\overrightarrow{\mathrm{h}}) - F^{[l]}(\overrightarrow{\mathrm{x}})||^2_2$$

$$\mathcal{L}_{style} = \sum_{l \in style} w^{[l]}_{style} ||\mathcal{G}(F^{[l]}(\overrightarrow{\mathrm{h}})) - \mathcal{G}(F^{[l]}(\overrightarrow{\mathrm{x}}))||^2_2$$

NST, however, extracts information from the entire image, including irrelevant information such as the background of the style leaf image and applies this to the entire content information, including the background of the content leaf image, which we would like to leave untouched. Specifically, we would like to exercise spatial control on NST in order to extract just the relevant plant leaf pixels from both the content image and the style image as observed in figure 2.



(a) NST (b) Masked-NST

Fig. 2: Examples of NST and Masked-NST with black rot disease applied to a healthy leaf

As documented by [12], this can be achieved by simply applying a mask over the pixel locations belonging to the leaf segments for both the content image and the style image.

[1]https://github.com/sharanramjee/plant-disease-nst

Fig. 3: Leaf mask generation pipeline

[1] generate the segmentation mask by converting the image to grayscale, applying gaussian blur to smoothen the edges within the leaf, applying image thresholding with an inverted binary threshold to obtain a mask of the leaf foreground, applying the closing (dilation followed by erosion) image morphological operation to close small holes in the mask, and then finally obtaining the contours of the mask as illustrated in Fig. 3.

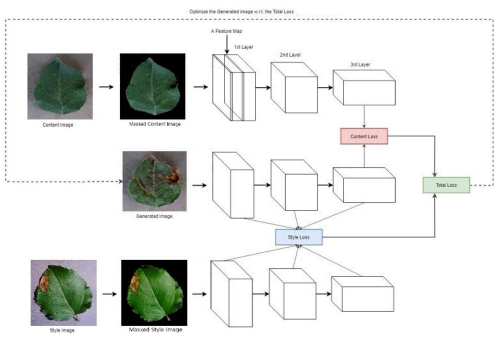


Fig. 4: Masked-NST disease application

The masks are applied to the content and style images in order to extract the foreground information from both the content and style images after passing through the VGG19 model. Finally, NST is performed as usual to apply the disease from the diseased style image to the healthy content image as illustrated in Fig. 4.

B. *CycleGAN*

CycleGAN [13] is a Generative Adversarial Network for unpaired Image-to-Image translation. For the task of converting healthy leaves (H) to diseased leaves (D), CycleGAN learns a forward mapping $G : H \rightarrow D$ such that the generated diseased leaves $G(H)$ are indistinguishable from the real diseased leaves D. Likewise, CycleGAN learns the inverse mapping from diseased leaves to healthy leaves $F : D \rightarrow H$, and introduces a cycle consistency loss to enforce that the reconstructed healthy leaf is as close as possible to the original ($F(G(H)) \approx H$ and vice versa $G(F(D)) \approx D$). By doing so, CycleGAN can learn mappings that transform healthy leaves to diseased ones while preserving the structure and content of the original healthy leaf, which is desirable for the task of style transfer.

V. Baseline Model Experiments and Results

In order to establish a baseline model, we evaluated a variety of both traditional machine learning and modern deep learning methods on the PlantVillage dataset in sections V-A and V-B, respectively. Our evaluation metrics of choice were: Precision, Recall, F1-score, and PlantVillage test set accuracy. The PlantVillage dataset was divided using a 60:20:20 train-val-test split for these baseline model experiments.

A. *Traditional Machine Learning Methods*

The use of traditional machine learning methods for plant leaf disease diagnosis has been extensively documented in [14] and [15]. For each classifier, we employed a K-fold cross-validation with a K value of 10, where the hyperparameters for each of the models were chosen using a grid-search. We compare and contrast the performance of the following classifiers on the PlantVillage dataset in table I: Logistic Regression (LR) [16], Linear Discriminant Analysis (LDA) [17], K Nearest Neighbors (KNN) [18], Decision Trees (CART) [19], Random Forests (RF) [20], Naive Bayes (NB) [21], and Support Vector Machines (SVM) [22].

TABLE I: Evaluation metrics of traditional ML models on PlantVillage dataset

	Evaluation Metrics			
Model	**Precision**	**Recall**	**F1-score**	**Test accuracy**
LR	0.94	0.94	0.94	0.941
LDA	0.93	0.93	0.92	0.925
KNN	0.94	0.94	0.93	0.934
CART	0.92	0.92	0.92	0.916
RF	0.95	0.95	0.95	0.951
NB	0.86	0.83	0.82	0.828
SVM	0.94	0.93	0.93	0.934

While traditional machine learning methods benefit from faster training times and are less computationally expensive to train, their performances, as evaluated through the metrics, still fall short compared to modern deep learning methods as examined in section V-B.

B. *Deep Learning Methods*

Modern deep learning and neural network-based methods have taken the agriculture industry by storm and has seen major advances in the field of plant leaf disease diagnosis as documented in [23]. Given the computational power limitations, we were unable to employ K-fold cross-validation for the deep learning models. However, similar to the traditional machine learning methods, a grid-search was used to tune the hyperparameters of each of the deep learning methods. We compare and contrast the performance of the following deep learning models on the PlantVillage dataset in table II: AlexNet [24], MobileNet [25], and ResNet152 [26].

TABLE II: Evaluation metrics of the baseline models on original PlantVillage dataset

	Evaluation Metrics			
Model	**Precision**	**Recall**	**F1-score**	**Test accuracy**
AlexNet	0.96	0.97	0.96	0.965
MobileNet	0.95	0.95	0.95	0.953
ResNet152	0.98	0.98	0.98	0.961

Given the computational constraints and the performances of the models in table II as evaluated by the evaluation metrics, we decided to select ResNet152 as our baseline model since it yielded the best performance for the computational power and time taken to train the model.

VI. Data Augmentation Experiments and Results

In order to evaluate the classifier's generalization performance, the model was first trained on the PlantVillage dataset until convergence, and subsequently tested on the PlantPathology dataset. Since the PlantPathology dataset only consists of apple diseases, we focused our task to investigating the impact of augmenting apple images in the PlantVillage dataset on the classifier's performance for the PlantPathology dataset. Our evaluation metrics are Precision, recall, F-1 score on the PlantPathology dataset. Furthermore, we present the Precision, recall, F-1 score and test accuracy on the PlantVillage dataset (with an 80:20 train-val split) to demonstrate the impact of each data augmentation method.

For CycleGAN and Masked-NST, we use PCA [27] and T-SNE [28] to reduce the dimensionality of the images from 196608 ($256 \times 256 \times 3$) to 3 dimensions so that the distributions of the data can be visualized for each of the 4 classes (blue: healthy, red: apple scab, pink: black rot, cyan: cedar apple rust) as observed in figures 7, and 9. The formation of 4 distinct clusters in the T-SNE plots indicates that the distribution of the data for each of the 4 classes are separable despite the fact that the content of the data across the 4 classes are the same (set of healthy content images) and the similarity of the PCA visualizations for the CycleGAN and NST (i.e. cross-label data augmentation methods) indicate that these clusters correspond to the same set of classes. In other words, CycleGAN and NST were successful in generating distributions of images that look like diseased plant images using healthy plant images.

TABLE III: Evaluation of augmentation methods on the PlantVillage dataset

	Evaluation Metrics			
Augmentation	**Precision**	**Recall**	**F1-score**	**Test accuracy**
None	**0.98**	**0.98**	**0.98**	**0.98**
Traditional	0.976	0.976	0.976	0.976
Balancing	0.955	0.953	0.953	0.953
CycleGAN	0.962	0.961	0.961	0.961
NST	0.963	0.963	0.963	0.962

TABLE IV: Evaluation of augmentation methods on the PlantPathology dataset

	Evaluation Metrics			
Method	**Precision**	**Recall**	**F1-score**	**Accuracy**
None	0.749	0.664	0.665	0.664
Traditional	0.794	0.721	0.722	0.721
Balancing	0.713	0.639	0.642	0.639
CycleGAN	0.702	0.691	0.683	0.691
NST	**0.799**	**0.782**	**0.777**	**0.782**

A. *Traditional Data Augmentation*

To establish a baseline, we first implemented traditional augmentation methods that that include rotation, resizing, horizontal flipping and sheering on the PlantVillage dataset. These simple transformations are intended to make the model more robust, and to prevent over-fitting to non-semantic representations in the dataset. Initially, we augmented all the classes PlantVillage dataset equally. In addition, to combat the imbalance in PlantVillage, we also implemented augmentation to balance all classes in the dataset. The traditional data augmentation method without class balancing significantly outperformed augmentation with balancing on both datasets as shown in Tables III and IV.

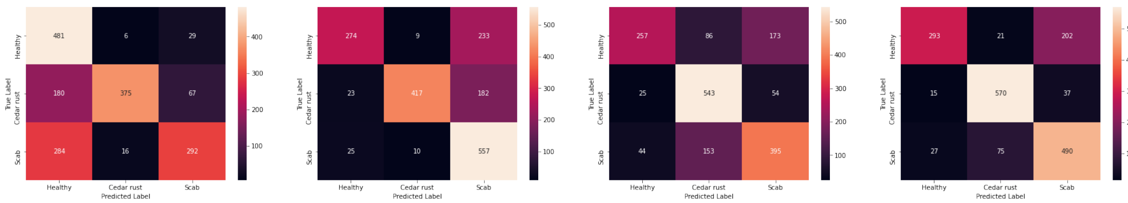
Moreover, traditional augmentation with class balancing performed worse than no augmentation on both datasets, which can be attributed to overfitting to samples in the underrepresented plant-disease classes; since images underrepresented classes are duplicated several times to implement class balancing.

Although traditional data augmentation had better generalization performance than no data augmentation, it performed worse on the PlantVillage dataset. Since the statistical distribution of images is very similar across the PlantVillage dataset, it is expected that no data augmentation would strongly overfit to the PlantVillage dataset and would not have good generalization performance. Our results confirm this, since traditional data augmentation notably improved the generalization performance of the classifier. Furthermore, it can observed from the confusion matrices in figure 5-b the False positive rate is very high between Scab and healthy apple leaves, which can attributed to scab being a very subtle discoloration that the classifier is unable to robustly capture with traditional data augmentation.

B. *CycleGAN*

In order to compare the effectiveness of NST to the more popular GAN-based augmentation methods, a vanilla CycleGAN [13] was used based on the authors' implementation to transform all the 1317 healthy apple leaf images in PlantVillage's training set into each of the apple disease classes (Scab, Black rot, Cedar rust) for data augmentation. The classifier was then trained on the PlantVillage dataset with the augmented diseased apple leaf leaves, using the traditional augmentation policy with no balancing. As shown in tables III and IV, the performance of the classifier trained on the CycleGAN augmented dataset was worse than the model trained with no augmentation for both datasets.

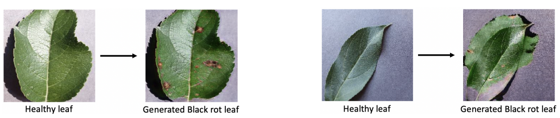
Although CycleGAN qualitatively produces very realistic results as shown in Figure 6-a, the degradation in performance can be attributed to the failure case where the disease is applied outside of the leaf region and into the background, as seen in Figure 6-b. These results are inline with the findings of [5]; using a vanilla CycleGAN with no attention mechanism will lead to a degradation in classifier



(a) No augmentation (b) Traditional (c) CycleGAN (d) NST

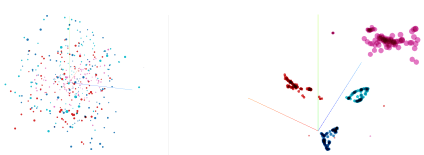
Fig. 5: Confusion Matrix of different augmentation methods on the PlantPathology dataset

performance due to the CycleGAN failure cases. This failure case has also lead to a notable increase in the false positive between Cedar rust and scab leaves, as well as between Cedar rust and healthy leaves, as seen in figure 5-c, which indicates that CycleGAN is unable to help the classifier's generalization performance.



(a) Successful disease transfer (b) Failure case

Fig. 6: CycleGAN applied to the apple leaf diseases



(a) PCA (b) T-SNE

Fig. 7: 3-D visualizations of the CycleGAN generated data

C. Masked - Neural Style Transfer

Prior to performing Masked-NST, the VGG19 model was deep trained in accordance to [2] on the PlantVillage dataset. The following Masked-NST hyperparameters were set to be the same values as those used in the NST-based data augmentation implemented in [3]: noise to content ratio = 0, content weight (α) = 1, style weight (β) = 0.2. As for the optimizer, we decided to use the Limited-memory Broyden–Fletcher–Goldfarb–Shanno (L-BFGS) algorithm [29] because L-BFGS does not have a learning rate hyperparameter and we found that the efficacy of optimizers such as Adam, when it came to performing Masked-NST, was highly dependent on the learning rate used and this learning rate would have to be tuned for each pair of content and style images in order to obtain ideal results. Similar to the CycleGAN, Masked-NST was used to transform all the 1317 healthy apple leaf images in PlantVillage's training set into each of the apple disease classes (Scab, Black rot, Cedar rust) for data augmentation, which are then used to train the classifier. As seen in tables III and IV, while the performance of the classifier trained on the Masked-NST augmented dataset deteriorated for PlantVillage dataset (which can be attributed to a reduction in overfitting), the classifier generalizes better as it has the best performance across all evaluation metrics in comparison to the other data augmentation methods considered on the PlantPathology dataset.

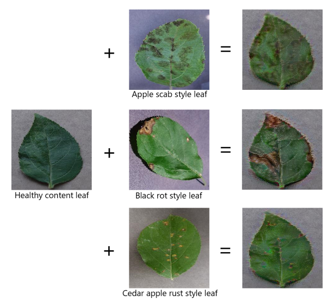
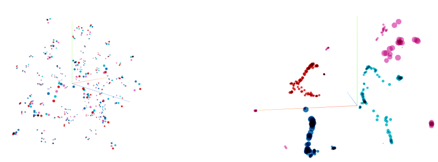


Fig. 8: Masked-NST applied to the apple leaf diseases

Furthermore, the confusion matrices in figure 5 indicate that data augmentation using Masked-NST, unlike the other data augmentation methods considered, is successfully able to rectify the scab/cedar rust misclassification on the PlantPathology dataset as the classifier is now able to generalize better



(a) PCA (b) T-SNE

Fig. 9: 3-D visualizations of the NST generated data

VII. Conclusion / Future Work

To combat the significant performance degradation of classifiers trained on PlantVillage when evaluated on leaf images captured under different conditions, several data augmentation methods were evaluated as part of this study. In summary, masked NST was found to be the most effective data augmentation method for improving generalization performance. For future work, the effectiveness of masked NST in improving generalization performance should be compared to attention guided GAN models, such as LeafGAN.

VIII. Contributions

A. Sofian Zalouk

Examined and documented the class imbalance of the PlantVillage dataset. Implemented and performed data augmentation using traditional data augmentation methods and CycleGAN. Implemented and collected evaluation metrics for ResNet152. Evaluated the effectiveness of the different augmentation methods on the PlantPathology dataset.

B. Sharan Ramjee

Implemented and performed data augmentation using NST and Masked-NST. Implemented and collected evaluation metrics for traditional machine learning methods, AlexNet, and MobileNet. Implemented PCA and T-SNE to examine the effectiveness of data augmentation using CycleGAN and Masked-NST. Created and maintained the `plant-disease-nst` GitHub repository with the algorithms proposed and considered in this paper along with all the images used in this paper.

References

[1] S. P. Mohanty, D. P. Hughes, and M. Salathé, "Using deep learning for image-based plant disease detection," *Frontiers in plant science*, vol. 7, p. 1419, 2016.

[2] M. Brahimi, M. Arsenovic, S. Laraba, S. Sladojevic, K. Boukhalfa, and A. Moussaoui, "Deep learning for plant diseases: detection and saliency map visualisation," in *Human and machine learning*. Springer, 2018, pp. 93–117.

[3] J. A. Pandian, G. Geetharamani, and B. Annette, "Data augmentation on plant leaf disease image dataset using image manipulation and deep learning techniques," in *2019 IEEE 9th International Conference on Advanced Computing (IACC)*. IEEE, 2019, pp. 199–204.

[4] M. Arsenovic, M. Karanovic, S. Sladojevic, A. Anderla, and D. Stefanovic, "Solving current limitations of deep learning based approaches for plant disease detection," *Symmetry*, vol. 11, no. 7, p. 939, 2019.

[5] Q. H. Cap, H. Uga, S. Kagiwada, and H. Iyatomi, "Leafgan: An effective data augmentation method for practical plant disease diagnosis," *arXiv preprint arXiv:2002.10100*, 2020.

[6] D. Hughes, M. Salathé *et al.*, "An open access repository of images on plant health to enable the development of mobile disease diagnostics," *arXiv preprint arXiv:1511.08060*, 2015.

[7] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei, "Imagenet: A large-scale hierarchical image database," in *2009 IEEE conference on computer vision and pattern recognition*. Ieee, 2009, pp. 248–255.

[8] R. Thapa, N. Snavely, S. Belongie, and A. Khan, "The plant pathology 2020 challenge dataset to classify foliar disease of apples," *arXiv preprint arXiv:2004.11958*, 2020.

[9] L. A. Gatys, A. S. Ecker, and M. Bethge, "Image style transfer using convolutional neural networks," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 2414–2423.

[10] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," *arXiv preprint arXiv:1409.1556*, 2014.

[11] R. Nakano, "A discussion of 'adversarial examples are not bugs, they are features': Adversarially robust neural style transfer," *Distill*, 2019, https://distill.pub/2019/advex-bugs-discussion/response-4.

[12] A. Handa, P. Garg, and V. Khare, "Masked neural style transfer using convolutional neural networks," in *2018 International Conference on Recent Innovations in Electrical, Electronics & Communication Engineering (ICRIEECE)*. IEEE, 2018, pp. 2099–2104.

[13] J.-Y. Zhu, T. Park, P. Isola, and A. A. Efros, "Unpaired image-to-image translation using cycle-consistent adversarial networks," https://github.com/junyanz/pytorch-CycleGAN-and-pix2pix, 2017, commit f13aab8.

[14] S. Ramesh, R. Hebbar, M. Niveditha, R. Pooja, N. Shashank, P. Vinod *et al.*, "Plant disease detection using machine learning," in *2018 International Conference on Design Innovations for 3Cs Compute Communicate Control (ICDI3C)*. IEEE, 2018, pp. 41–45.

[15] X. Yang and T. Guo, "Machine learning in plant disease research," *European Journal of BioMedical Research*, vol. 3, no. 1, pp. 6–9, 2017.

[16] R. Harikrishnan and L. d. Rio, "A logistic regression model for predicting risk of white mold incidence on dry bean in north dakota," *Plant disease*, vol. 92, no. 1, pp. 42–46, 2008.

[17] N. Wu, M. Li, L. Chen, Y. Yuan, and S. Song, "A lda-based segmentation model for classifying pixels in crop diseased images," in *2017 36th Chinese Control Conference (CCC)*. IEEE, 2017, pp. 11 499–11 505.

[18] E. Hossain, M. F. Hossain, and M. A. Rahaman, "A color and texture based approach for the detection and classification of plant leaf disease using knn classifier," in *2019 International Conference on Electrical, Computer and Communication Engineering (ECCE)*. IEEE, 2019, pp. 1–6.

[19] H. Sabrol and S. Kumar, "Intensity based feature extraction for tomato plant disease recognition by classification using decision tree," *International Journal of Computer Science and Information Security*, vol. 14, no. 9, p. 622, 2016.

[20] B. J. Samajpati and S. D. Degadwala, "Hybrid approach for apple fruit diseases detection and classification using random forest classifier," in *2016 International Conference on Communication and Signal Processing (ICCSP)*. IEEE, 2016, pp. 1015–1019.

[21] J. Yuen and G. Hughes, "Bayesian analysis of plant disease prediction," *Plant Pathology*, vol. 51, no. 4, pp. 407–412, 2002.

[22] K. Elangovan and S. Nalini, "Plant disease classification using image segmentation and svm techniques," *International Journal of Computational Intelligence Research*, vol. 13, no. 7, pp. 1821–1828, 2017.

[23] K. P. Ferentinos, "Deep learning models for plant disease detection and diagnosis," *Computers and Electronics in Agriculture*, vol. 145, pp. 311–318, 2018.

[24] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," in *Advances in neural information processing systems*, 2012, pp. 1097–1105.

[25] A. G. Howard, M. Zhu, B. Chen, D. Kalenichenko, W. Wang, T. Weyand, M. Andreetto, and H. Adam, "Mobilenets: Efficient convolutional neural networks for mobile vision applications," *arXiv preprint arXiv:1704.04861*, 2017.

[26] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 770–778.

[27] S. Wold, K. Esbensen, and P. Geladi, "Principal component analysis," *Chemometrics and intelligent laboratory systems*, vol. 2, no. 1-3, pp. 37–52, 1987.

[28] L. v. d. Maaten and G. Hinton, "Visualizing data using t-sne," *Journal of machine learning research*, vol. 9, no. Nov, pp. 2579–2605, 2008.

[29] D. R. S. Saputro and P. Widyaningsih, "Limited memory broyden-fletcher-goldfarb-shanno (l-bfgs) method for the parameter estimation on geographically weighted ordinal logistic regression model (gwolr)," in *AIP Conference Proceedings*, vol. 1868, no. 1. AIP Publishing LLC, 2017, p. 040009.