

Color-Aware Two-Branch DCNN for Efficient Plant Disease Classification

Joao Paulo Schwarz Schuler^{1,✉}, Santiago Romani¹, Mohamed Abdel-Nasser^{1,2}, Hatem Rashwan¹, Domenec Puig¹

¹DEIM, Universitat Rovira i Virgili, Spain

²Electrical Engineering Department; Aswan University, Aswan, Egypt

joapaulo.schwarz@estudiants.urv.cat[✉], santiago.romani@urv.cat, mohamed.abdelnasser@urv.cat, hatem.abdellatif@urv.cat, domenec.puig@urv.cat

Abstract

Deep convolutional neural networks (DCNNs) have been successfully applied to plant disease detection. Unlike most existing studies, we propose feeding a DCNN CIE Lab instead of RGB color coordinates. We modified an Inception V3 architecture to include one branch specific for achromatic data (L channel) and another branch specific for chromatic data (AB channels). This modification takes advantage of the decoupling of chromatic and achromatic information. Besides, splitting branches reduces the number of trainable parameters and computation load by up to 50% of the original figures using modified layers. We achieved a state-of-the-art classification accuracy of 99.48% on the Plant Village dataset and 76.91% on the Cropped-PlantDoc dataset.

Keywords: CNN, DCNN, Deep Learning, Plant Disease, CIE LAB, Neural Networks, Artificial Intelligence, Multipath.

Received: 21 April 2022

Accepted: 24 June 2022

Online: 24 June 2022

Published: 30 June 2022

1 Introduction

The automation of plant disease control is essential for early-stage symptom detection and continuous monitoring of crops. Such automation has a high impact on improving efficiency and productivity, especially in large fields [41]. To automatically recognize plant leaf diseases, emerging AI technologies such as computer vision and deep convolutional neural networks (DCNNs) have been recently employed.

Initial studies used handcrafted features from leaf images [21]. Shallow classifier algorithms were proposed: the K-Nearest-Neighbors (KNN), Support Vector Machines (SVMs), decision trees and shallow non-convolutional neural networks [21]. Later, the trend switched to DCNN architectures capable of automatically extracting features and performing efficient classification [21]. Many CNN architectures, including LeNet [2], CaffeNet [33], AlexNet [18], GoogLeNet [21, 20, 6], Inception V3 [21, 23, 37, 36, 19] and DenseNet [21] have been applied to plant disease image classification.

The majority of the previous architectures applied to plant leaf disease identification use the Red-Green-Blue (RGB) color values of input pixels. However, RGB components are highly correlated [22]. Specifically, intensity variations induced by illumination changes, edges or texture modify the three RGB values by the same proportion.

Transforming RGB channels into some type of achromatic-chromatic space, such as CIE Lab, effectively isolates the gray-level features in the L channel and the color-related features in the AB channels.

Many other color transformations that decorrelate the RGB channels have been proposed, such as HSV, YUV, YIQ, etc., which are vastly known and long-standing in the fields of color perception [25] and colorimetry [40]. However, we think that the CIE LAB color space is the most convenient for our task because the AB chromatic components layout the chromaticities in a cartesian space, while others like HSV use a polar space: the Hue component is circular, so the ends of the component range are connected, which is something that CNNs cannot deal with. Besides, other authors like Gowda et al. [11] found that CIE Lab color space provided the best classification accuracy when training a DCNN with the CIFAR-10 dataset [17], compared to other nine color spaces (RGB, HSV, YUV, YIQ, XYZ, YPbPr, YCbCr, HED and LCH).

In an initial work [30], we trained a two-path CNN with the CIFAR-10 dataset encoded in the CIE Lab color space. Inspired by Deep Roots [15], multipath convolutional neural networks [38] and dual path neural networks [4], we applied the idea of parallel branches for a better learning of color features, so we split the first layers of our CNN in two parallel branches, one dedicated to achromatic data (L) and the other to chromatic data (AB).

Following this idea, in [27] we showed that each of the parallel branches learn the features related to the nature of each cue, and it makes the modified DCNN more resistant to different types of noise at classifying plant diseases. Indeed, L filters focus on achromatic features, like texture, edges, damaged areas, etc. of the leaf, while the AB filters focus on chromatic fea-

tures, like lesions, general color of the leaf, etc. The AB channels should not be further split into separate channels, because the chromaticity of the pixels is encoded in both coordinates simultaneously.

In this paper, we investigate the influence of each branch in the classification accuracy. For this aim, we created an Inception V3 [35] based architecture that has two branches (paths) along the first three convolutional layers. One branch is fed from the L component while the other branch is fed from the AB components. Furthermore, we test extreme cases by feeding our DCNN from only one of the two branches, to check what our network can do with a single cue (chromatic or achromatic information) and without the other cue. With this respect, dealing with only the L channel makes our system comparable with other methods that purely work on grayscale images.

For this work, we tested our DCNNs on the Plant Village dataset [14], which contains samples of 12 healthy crops and 26 crop diseases. We also tested our DCNNs on the Cropped-PlantDoc dataset [32], which has 13 plant species and 27 classes of healthy and diseased crops.

The key contributions of this article to image-based plant leaf disease diagnostics can be summarized as follows:

- We present a feasible plant leaf image classification method based on an efficient DCNN architecture with separate branches dedicated to chromatic and achromatic information.
- We provide detailed performance analysis of several variants of our DCNN architecture, tested on the Plant Village dataset [14] and Cropped-PlantDoc dataset [32].
- Our DCNN variants have achieved state-of-the-art plant leaf disease classification with 30% to 50% fewer filter weights and floating point computations along the first three convolutional layers.

The remainder of this article is structured as follows: section II presents and discusses relevant work regarding DCNNs and image-based plant disease diagnostics. Section III presents the proposed method. The results and discussion are given in sections IV and V. Section VI summarizes the main conclusions and suggests future work.

2 Related Work

In 1980, Fukushima [9] devised a layered artificial neural network inspired by the visual cortex structure for image classification. Such network showed that the first layer contains neurons detecting simpler patterns with a small receptive field. Deeper layers detect more complex patterns with wider receptive fields, by composing patterns from previous layers.

In 2012, Krizhevsky et al. achieved a major breakthrough in the ImageNet Large Scale Visual Recognition Challenge [18] with their AlexNet architecture.

Since then, many other CNN architectures have been introduced: ZFNet [42], VGG [31], GoogLeNet [34], ResNet [12], DenseNet [13] and others. Since the number of layers has increased from 5 to more than 200, those models are usually referred to as “deep learning”.

A number of machine learning methods have been proposed specifically for image-based plant disease diagnosis [6, 33], including methods specifically designed for cucumbers [8], bananas [2], cassavas [23], tomatoes [7, 37, 19] and wheat [16].

Ferrentinos [6] tested 5 existing architectures with a 58 class image dataset with healthy and sick plants: AlexNet, AlexNetOWTBn, GoogLeNet, Overfeat and VGG. Ferrentinos found test accuracies ranging from 99.06% with AlexNet to 99.48% with VGG. Despite the enormous difference in the number of trainable parameters in these architectures, the test accuracy was always above 99%.

Maeda et al. [19] studied the application of 5 existing architectures, including: AlexNet, GoogleNet, Inception V3, ResNet-18 and ResNet-50, to tomato diseases. In this study, test accuracies were also found to be above 99%. Despite these excellent results, Ferrentinos [6] noted that there are problematic situations for images captured in the field, such as shading and leaves not centered in the image. When processing field images in the experiments, it drastically reduces the classification accuracy.

Chaudhary et al. [3] studied plant disease spot segmentation in YCbCr, HSI and CIE Lab color spaces. For all color channels, they experimentally found that feeding their segmentation model from the CIE Lab’s A channel provides more accurate results than from other channels. It is important to note that the A channel is a chromatic channel, and it provided better results than the achromatic L channel in their work.

Amara et al. [2] applied a plain LeNet architecture with 60x60 pixels images for banana leaf disease classification. Interestingly, they achieved an 85.94% test accuracy with grayscale images and a 92.88% test accuracy with RGB images. Mohanty et al. [20] studied plant leaf diseases with both grayscale and RGB color images processed with AlexNet and GoogLeNet (Inception V1) architectures. Their architectures were trained with the Plant Village dataset [14]. Their best results were found with 80% of the image samples allocated for training while 20% of the samples were allocated for testing. The only architectural modifications are the number of classes and the input size set at 256x256 pixels for GoogLeNet. All experiments described at [20] done with RGB images achieved higher accuracies than experiments done with grayscale images. Results obtained in [2, 20] indicate that chromatic information is important for plant leaf disease classification.

Previous DCNN works from Ferrentinos [6], Maeda et al. [19], Amara et al. [2] and Mohanty et al. [20] utilized off-the-shelf architectures with minor changes in the number of classes and the input layer.

Geetharamani et al. [10] propose a DCNN with 3 convolutional layers, 2 max pooling layers and 2 dense layers trained with the Plant Village dataset using data augmentation. After a number of experiments of the batch size and data augmentation, they achieved a 96.46% classification accuracy. They distributed the dataset into 91%, 6% and 3% of the samples for training, validation and test, respectively.

Toda & Okura’s proposal [36] is based on the Inception V3 architecture [35]. The authors adjusted the last layer of the original network to fit in the 38 classes of the Plant Village dataset. They distributed the dataset into 60%, 20% and 20% of the samples for training, validation and test, respectively. Additionally, they performed an ablation study to determine the optimal number of mixed layers in the inception module, successively trimming the layers from the deepest to the shallowest. They found that with just the 6 former mixed layers, the network provides very similar accuracy to the original network with 11 mixed layers (97.14% vs. 97.15%) while saving approximately 3/4 of the memory for storing the neuron weights (5.17 million vs. 21.88 million weights).

Ngugi et al. [21] also worked with the Plant Village dataset, but they used transfer learning. Since we train our networks from scratch, our results cannot be compared with their results. Transfer learning is a technique that uses the learned data from a related domain to improve the learning in a target domain [39]. We prefer to focus on analyzing the impact of our architectural modifications and not to care about the effect of improving the accuracy with the use of parameters transferred from another domain. Therefore, comparing our results with other works that do apply transfer learning is unfair because they take advantage of pre-trained values. In contrast, we compare our test accuracies and F1 scores directly with those provided in [36], [10] and [20] as those papers also use training from scratch.

Singh et al. [32] created the Cropped-PlantDoc dataset, which has 13 plant species and 27 classes. Similar to the Plant Village dataset, the original PlantDoc dataset includes pictures of individual leaves. However, those images also show complex backgrounds, and the area covered by the target leaves varies, which makes it a much harder problem to classify than for the Plant Village images. To address this drawback, the authors decided to manually crop the image regions containing target leaves. This provides conveniently framed leaves while significantly increasing the number of samples (approximately 9K) because they may extract several samples from each original PlantDoc image (approximately 2.6K).

3 Methodology

Figure 1 shows two designs of CNNs that analyze RGB pictures of plant leaves for plant disease classification. The design shown on the left corresponds to Toda & Okura’s proposal, which we have chosen as our reference baseline model.

The design shown at the right of figure 1 corresponds to our proposal, which splits the first three convolutional layers of the baseline into two branches, one for the L channel and another for the AB channels, computed from the input RGB image. Then, the output from each branch is concatenated to follow the rest of the network as in the baseline.

Another relevant remark is that we use a hyperparameter x to determine the distribution of the original number of filters among the L and AB branches. This allows us to determine the optimal contribution of each branch to the classification task. In the original Inception V3 implementation, the first 3 convolutional layers have 32, 32 and 64 filters, respectively. For our proposal, we have mainly checked three variants, named after the percentage of filters dedicated to achromatic (L) and chromatic (AB) branches: 20%L+80%AB, 50%L+50%AB and 80%L+20%AB. For these variants, the value of x is set to 13, 32 and 51, respectively. Thus, the number of $L|AB$ filters in the first two layers will be 6|26, 16|16 and 26|6, respectively, for each variant. In the third layer, the number of $L|AB$ filters will be 13|51, 32|32 and 51|13, respectively, for each variant.

To compare our proposal with the baseline in a fair fashion, we have imposed that the sum of the filters of the two branches in each layer must be the same as in the Inception V3 design. However, our filters carry a fraction of the original number of weights (from 1/3 to 2/3).

model	1st layer weights	2nd layer weights	3rd layer weights
baseline	0.8k	9k	18k
20%L + 80%AB	0.5k	6k	13k
50%L + 50%AB	0.4k	5k	9k
80%L + 20%AB	0.3k	6k	13k

Table 1: Number of weights for each of the first 3 convolutional layers, for baseline and our variants.

model	1st layer flops	2nd layer flops	3rd layer flops
baseline	21M	227M	453M
20%L + 80%AB	12M	158M	315M
50%L + 50%AB	10M	113M	226M
80%L + 20%AB	8M	158M	315M

Table 2: Number of required forward pass floating point operations for each of the first 3 convolutional layers, for baseline and our variants.

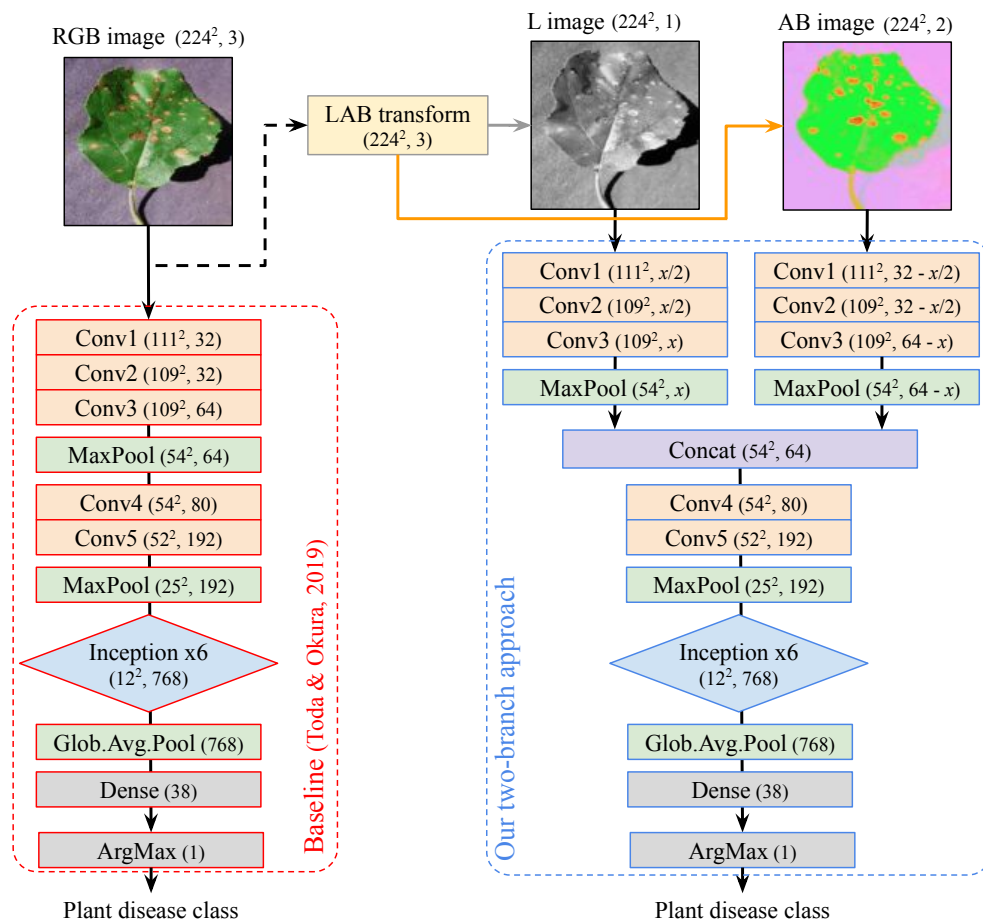


Figure 1: Graphical representation of worked network architectures: at the left, the Toda & Okura’s single-branch (baseline) approach fed from an RGB image; at the right, our two-branch approach fed from L+AB images. Expressions containing x define a varying number of filters in L and AB branches. When $x/2$ is not an integer number, we use the floor function to round it.

The implementation details of our approach are generally based on [36] which is Inception V3 based (see figure 1). Each convolutional layer is composed by a 2D convolution, batch normalization and a ReLU as the activation function. All convolutional filters from Conv1 to Conv5 are 3×3 except for Conv4, which is 1×1 . In Conv1, there is a stride of 2. All convolutional layers from Conv1 to Conv5 do not have padding except for Conv4 that is zeroed padded by 1px. We do not use data augmentation as we study the net effect of the proposed architecture (L/AB separate branches) on the accuracy, beyond extra refinements (data augmentation, transfer learning) that may lead to some degree of improvement but not due to the proposed architecture. The training data is shuffled before each epoch. The optimization method is the stochastic gradient descent. The loss function is the categorical cross entropy function. The batch size is 32. The test accuracy is obtained with the parameters from the epoch with the highest validation accuracy. All models have been trained from scratch without transfer learning.

All experiments were implemented with K-CAI [28] and Keras [5] on top of Tensorflow 2 [1], with various

underlying hardware configurations including NVIDIA 1070, 1080, K80, T4 and V100 video cards.

For the Plant Village dataset, we trained all DCNNs for 30 epochs with a constant learning rate of 0.01. We split the Plant Village Dataset into 60% of the samples for training, 20% for validation and 20% for testing. After a random split of the dataset into the training, validation and testing subsets, this splitting will be used in all experiments, to ensure that results from different experiments are not affected by the sample splitting. For this dataset, we weight the loss function according to the number of samples per class to give the same relevance to each class. This gives similar classification accuracy across all classes. For the Cropped-PlantDoc dataset, we trained all DCNNs for 240 epochs starting with a learning rate of 0.01 and decaying 1% per epoch. We split the Cropped-PlantDoc dataset into 65% of the samples for training, 15% for validation and 20% for testing.

Our source code written for these experiments and their raw result files are publicly available at <https://github.com/joapauloschuler/two-branch-plant-disease/>.

4 Results

We have assessed the performance of several models: the original Toda et al. single-branch architecture for RGB images, which we refer as the baseline; and three variants of our two-branch architecture for L+AB channels. For these experiments, we computed the classification accuracy for the testing subset and the multiclass F1 score [24].

Table 3 collects the results for the Plant Village dataset. It shows that our two-branch 20%L+80%AB variant provides both the best test accuracy and F1 score. It renders a modest but clear improvement (1.11% in accuracy and 0.87% in F1) over the best pre-existing model, Mohanty’s GoogleLeNet. This 20%L+80%AB variant is slightly better than the other two-branch variants (up to 0.4% test accuracy). The baseline underperforms our worst variant (two-branch 80%L+20%AB) with a gap of almost 2 percentage points in test accuracy.

The 2 worst models (Mohanty’s AlexNet and Mohanty’s GoogLeNet) are both fed Gray images. The other RGB-based architectures achieve test accuracies ranging from 96.46% to 98.37%. In short, we can sort these results into 3 groups: LAB-fed two-branch models (highest accuracy), RGB-fed models (middle accuracy) and gray-level fed models (lowest accuracy). Our worst two-branch variant has a 0.71% higher accuracy and a 0.32 higher F1 score than the best performing RGB implementation.

We have also tested our DCNNs on the Cropped-PlantDoc dataset [32]. On this dataset, we reproduced the experiment from Toda et al. in regards to trimming the number of Inception V3 mixed layers using our 20%L+80%AB variant. Table 4 shows the obtained results. In this experiment, we came to the same conclusion that the ideal number of mixed layers is 6.

Once we decided to use just 6 mixed layers, we trained all our variants on the Cropped-PlantDoc dataset from scratch again. Our next results outperformed all previous models from Singh et al., as shown in table 5. It is interesting to note that we have not used transfer learning while Singh et al. used transfer learning, but we could not find any other proposal using this dataset to train from scratch. It is also vital to note that our variants have less than 10% of the trainable parameters used by Singh et al.’s best model while exceeding its test accuracy by more than 6 percentage points.

In table 5, we have also included two extreme variants of our model: 0%L + 100%AB and 100%L + 0%AB. In those variants, the DCNN is fed pure chromatic or achromatic information; hence, the DCNN operates with a single branch. The test accuracy from these experiments indicates that color is more important than gray-level information for classifying samples from this dataset. However, achromatic information also plays a role in plant disease classification since our best result has been obtained with a combination of both image cues in the same proportion.

5 Discussion

Typical first layer filters of DCNNs specialize in gray level features (Gabor-like filters) or in color-opponent filters, as in [18] and [42]. In our design, our L filters only need one single channel to learn the spatial patterns defined by the gray-level component. In addition, RGB filters replicate the same weights in their three channels to represent those gray-level patterns. Hence, L filters save 2/3 of the weights used by RGB filters. Similarly, for the case of color-opponent filters, our AB filters learn them using only two chromatic channels while regular RGB filters employ three channels for the same task. Therefore, AB filters save 1/3 of the weights. In total, our design saves from 1/3 to 1/2 of the weights in the first three layers. Also, it achieves similar savings in the computational floating point operations for carrying those convolutions, as shown in tables 1 and 2.

The results from table 3 show that the two-branch approach 20%L + 80%AB is slightly better than the classical RGB single-branch approach. This indicates that separating filters for achromatic-chromatic features enhances the classification ability of DCNNs. However, the difference in classification accuracy with respect to the baseline architecture is small because the DCNN optimization procedure is able to decorrelate the features encoded in the RGB channels. With a sufficient number of training epochs, an RGB single-branch DCNN will identify filters sensitive to lightness (Gabor-like grayscale filters) and other filters sensitive to color contrasts (see examples of RGB filters in [18] and [42]). Nevertheless, our two-branch methodology obtains similar accuracy with 30% to 50% fewer filter weights and floating point operations along the first 3 convolutional layers. This proves that the extra weights encoded in the first three layers of the RGB single-branch approach are redundant. All grayscale experiments have lower classification accuracy than their RGB counterparts, corroborating that chromatic information is important for plant leaf disease identification.

The results in table 4 show that a trimmed version of Inception V3, with less than 25% of the original parameters (5 million vs. 22 million), is capable of performing significantly better (5% in test accuracy) than the full-fledged version. We think that the extra number of parameters may become a drawback for training due to overfitting.

Finally, table 5 corroborates the advantage of our proposal. The results of these experiments show more significant differences than the results in table 3 because most of the methods obtained truly high test accuracy on the Plant Village dataset, leaving little room for improvement. Since Cropped-PlantDoc dataset is much more difficult to classify, our simple but effective methodology shows its advantage over single-branch RGB based approaches.

author	architecture	color space	parameters	accuracy	F1
Schuler	20%L + 80%AB	L—AB	5M	99.48%	0.9923
Schuler	50%L + 50%AB	L—AB	5M	99.11%	0.9866
Schuler	80%L + 20%AB	L—AB	5M	99.08%	0.9867
Mohanty [20]	GoogLeNet	RGB	5M	98.37%	0.9836
Mohanty [20]	AlexNet	RGB	60M	97.82%	0.9782
Toda [36]	Inception V3	RGB	5M	97.15%	0.9720
Geetharamani [10]	9 layers CNN	RGB	0.2M	96.46%	0.9815
Mohanty [20]	GoogLeNet	Gray	5M	96.21%	0.9621
Mohanty [20]	AlexNet	Gray	60M	94.52%	0.9449

Table 3: Test accuracy and F1 score of several DCNN models on Plant Village dataset classification. The results extracted from other papers are the ones obtained without transfer learning (not their best ones), for a fair model comparison with our results, since we do not use transfer learning.

architecture	color space	mixed layers	max. val. accuracy	test accuracy
20%L + 80%AB	L—AB	1	74.12%	72.50%
20%L + 80%AB	L—AB	2	77.27%	76.97%
20%L + 80%AB	L—AB	4	77.19%	74.68%
20%L + 80%AB	L—AB	6	78.77%	77.08%
20%L + 80%AB	L—AB	8	77.12%	73.90%
20%L + 80%AB	L—AB	10	75.62%	73.84%
20%L + 80%AB	L—AB	11	73.14%	74.23%

Table 4: Max validation and test accuracies trimming the number of mixed layers, trained on Cropped-PlantDoc dataset.

author	architecture	color space	parameters	accuracy	F1
Schuler	0%L + 100%AB	L—AB	5M	71.55%	0.71
Schuler	20%L + 80%AB	L—AB	5M	76.58%	0.76
Schuler	50%L + 50%AB	L—AB	5M	76.91%	0.76
Schuler	80%L + 20%AB	L—AB	5M	75.85%	0.75
Schuler	100%L + 0%AB	L—AB	5M	64.67%	0.66
Singh	InceptionResNet V2	RGB	55M	70.53%	0.70
Singh	InceptionV3	RGB	22M	62.06%	0.61
Singh	VGG16	RGB	138M	60.41%	0.60

Table 5: Test accuracy and F1 score with the Cropped-PlantDoc dataset.

6 Conclusion

In this paper, we proposed a two-branch DCNN for plant disease classification, where the first three convolutional layers specialize in learning chromatic and achromatic features from the CIE Lab color space.

The experiments conducted empirically prove that our approach can perform better than the classic one-branch RGB images fed DCNN while saving a portion of learnable parameters and floating point operations, reducing the numbers from 1/3 to 1/2 in those initial three layers. This is feasible because the RGB channels are highly correlated; hence, working in a decorrelated color space avoids redundant filter weights.

With regard to the optimal distribution of filters among achromatic and chromatic branches for plant disease classification, our experiments show that approximately 50% to 80% of the filters should enter the chromatic branch. This clearly indicates that color is important for this task.

In this work, we propose an optimization along the

first 3 convolutional layers only. As a next step, we aim to combine the optimization presented in this paper with optimizations done in deeper network layers [26, 29] using the worked datasets and other datasets commonly found in the deep learning literature.

References

- [1] ABADI, M., ET AL. TensorFlow: Large-scale machine learning on heterogeneous systems, 2015. Software available from tensorflow.org.
- [2] AMARA, J., BOUAZIZ, B., AND ALGERGAWY, A. A deep learning-based approach for banana leaf diseases classification. In *Datenbanksysteme für Business, Technologie und Web – Workshopband* (01 2017), pp. 79–88.
- [3] CHAUDHARY, P., CHAUDHARI, A., AND GODARA, S. Color transform based approach for disease spot detection on plant leaf. *International Journal of Computer Science and Telecommunications* 3 (06 2012), 65–70.

- [4] CHEN, Y., LI, J., XIAO, H., JIN, X., YAN, S., AND FENG, J. Dual path networks.
- [5] CHOLLET, F., ET AL. Keras. <https://keras.io>, 2015.
- [6] FERENTINOS, K. P. Deep learning models for plant disease detection and diagnosis. *Computers and Electronics in Agriculture* 145 (2018), 311 – 318.
- [7] FUENTES, A., YOON, S., KIM, S. C., AND PARK, D. S. A robust deep-learning-based detector for real-time tomato plant diseases and pests recognition. *Sensors* 17, 9 (2017).
- [8] FUJITA, E., KAWASAKI, Y., UGA, H., KAGIWADA, S., AND IYATOMI, H. Basic investigation on a robust and practical plant diagnostic system. In *2016 15th IEEE International Conference on Machine Learning and Applications (ICMLA)* (2016), pp. 989–992.
- [9] FUKUSHIMA, K. Neocognitron: A self-organizing neural network model for a mechanism of pattern recognition unaffected by shift in position. *Biological Cybernetics* 36, 4 (Apr 1980), 193–202.
- [10] G., G., AND J., A. P. Identification of plant leaf diseases using a nine-layer deep convolutional neural network. *Computers & Electrical Engineering* 76 (2019), 323 – 338.
- [11] GOWDA, S. N., AND YUAN, C. Colornet: Investigating the importance of color spaces for image classification. In *Computer Vision – ACCV 2018* (Cham, 2019), C. Jawahar, H. Li, G. Mori, and K. Schindler, Eds., Springer International Publishing, pp. 581–596.
- [12] HE, K., ZHANG, X., REN, S., AND SUN, J. Deep residual learning for image recognition. In *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)* (Los Alamitos, CA, USA, jun 2016), IEEE Computer Society, pp. 770–778.
- [13] HUANG, G., LIU, Z., AND WEINBERGER, K. Q. Densely connected convolutional networks. In *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)* (2017), IEEE, pp. 2261–2269.
- [14] HUGHES, D. P., AND SALATHÉ, M. An open access repository of images on plant health to enable the development of mobile disease diagnostics through machine learning and crowdsourcing. *CoRR abs/1511.08060* (2015).
- [15] IOANNOU, Y., ROBERTSON, D. P., CIPOLLA, R., AND CRIMINISI, A. Deep roots: Improving cnn efficiency with hierarchical filter groups. 5977–5986.
- [16] JOHANNES, A., PICON, A., ALVAREZ-GILA, A., ECHAZARRA, J., RODRIGUEZ-VAAMONDE, S., NAVAJAS, A. D., AND ORTIZ-BARREDO, A. Automatic plant disease diagnosis using mobile capture devices, applied on a wheat use case. *Computers and Electronics in Agriculture* 138 (2017), 200–209.
- [17] KRIZHEVSKY, A. Learning multiple layers of features from tiny images. Tech. rep., University of Toronto, Toronto, Ontario, 2009.
- [18] KRIZHEVSKY, A., SUTSKEVER, I., AND HINTON, G. E. Imagenet classification with deep convolutional neural networks. In *Advances in Neural Information Processing Systems 25*, F. Pereira, C. J. C. Burges, L. Bottou, and K. Q. Weinberger, Eds. Curran Associates, Inc., 2012, pp. 1097–1105.
- [19] MAEDA-GUTIÉRREZ, V., ET AL. Comparison of convolutional neural network architectures for classification of tomato plant diseases. *Applied Sciences* 10, 4 (2020).
- [20] MOHANTY, S. P., HUGHES, D. P., AND SALATHÉ, M. Using deep learning for image-based plant disease detection. *Frontiers in Plant Science* 7 (2016), 1419.
- [21] NGUGI, L. C., ABELWAHAB, M., AND ABO-ZAHHAD, M. Recent advances in image processing techniques for automated leaf pest and disease recognition – a review. *Information Processing in Agriculture* 8, 1 (2021), 27–51.
- [22] POULI, T., REINHARD, E., AND CUNNINGHAM, D. W. *Image Statistics in Visual Computing*, 1st ed. A. K. Peters, Ltd., USA, 2013.
- [23] RAMCHARAN, A., BARANOWSKI, K., MCCLOSKEY, P., AHMED, B., LEGG, J., AND HUGHES, D. P. Deep learning for image-based cassava disease detection. *Frontiers in Plant Science* 8 (2017), 1852.
- [24] RIJSBERGEN, C. J. V. *Information Retrieval*, 2nd ed. Butterworth-Heinemann, USA, 1979.
- [25] ROBERTSON, A. R. The cie 1976 color-difference formulae. *Color Research & Application* 2, 1 (1977), 7–11.
- [26] SCHULER, J., ROMANÍ, S., ABDEL-NASSER, M., RASHWAN, H., AND PUIG, D. *Grouped Pointwise Convolutions Significantly Reduces Parameters in EfficientNet*. IOS Press, 10 2021, pp. 383–391.
- [27] SCHULER, J., ROMANÍ, S., ABDEL-NASSER, M., RASHWAN, H., AND PUIG, D. *Reliable Deep Learning Plant Leaf Disease Classification Based on Light-Chroma Separated Branches*. IOS Press, 10 2021, pp. 375–381.
- [28] SCHULER, J. P. S. K-cai neural api, Dec. 2021.
- [29] SCHWARZ SCHULER, J. P., ROMANI, S., ABDEL-NASSER, M., RASHWAN, H., AND PUIG, D. Grouped pointwise convolutions reduce parameters in convolutional neural networks. *MENDEL* 28, 1 (Jun. 2022), 23–31.
- [30] SCHÜLER, J. P. S. Optimizing cnns first layer with respect to color encoding. In *6th URV Doctoral Workshop in Computer Science and Mathematics* (Tarragona, Catalunya, Spain, 4 2020), C. J. . A. Valls, Ed., vol. 1, Universitat Rovira i Virgil, Universitat Rovira i Virgil, p. 4.
- [31] SIMONYAN, K., AND ZISSERMAN, A. Very deep convolutional networks for large-scale image recognition. In *3rd International Conference on*

- Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings* (2015), Y. Bengio and Y. LeCun, Eds.
- [32] SINGH, D., JAIN, N., JAIN, P., KAYAL, P., KUMAWAT, S., AND BATRA, N. Plantdoc: A dataset for visual plant disease detection. In *Proceedings of the 7th ACM IKDD CoDS and 25th COMAD* (New York, NY, USA, 2020), CoDS COMAD 2020, Association for Computing Machinery, p. 249–253.
- [33] SLADOJEVIC, S., ARSENOVIC, M., ANDERLA, A., CULIBRK, D., AND STEFANOVIC, D. Deep neural networks based recognition of plant diseases by leaf image classification. *Computational Intelligence and Neuroscience 2016* (Jun 2016), 3289801.
- [34] SZEGEDY, C., LIU, W., JIA, Y., SERMANET, P., REED, S., ANGUELOV, D., ERHAN, D., VANHOUCKE, V., AND RABINOVICH, A. Going deeper with convolutions. In *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)* (Los Alamitos, CA, USA, jun 2015), IEEE Computer Society, pp. 1–9.
- [35] SZEGEDY, C., VANHOUCKE, V., IOFFE, S., SHLENS, J., AND WOJNA, Z. Rethinking the inception architecture for computer vision. In *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)* (2016), IEEE, pp. 2818–2826.
- [36] TODA, Y., AND OKURA, F. How convolutional neural networks diagnose plant disease. *Plant Phenomics 2019* (03 2019).
- [37] WANG, G., SUN, Y., AND WANG, J. Automatic image-based plant disease severity estimation using deep learning. *Computational Intelligence and Neuroscience 2017* (Jul 2017), 2917536.
- [38] WANG, M. Multi-path convolutional neural networks for complex image classification. *CoRR abs/1506.04701* (2015).
- [39] WEISS, K., KHOSHGOFTAAR, T. M., AND WANG, D. A survey of transfer learning. *Journal of Big Data* 3, 1 (May 2016), 9.
- [40] WYSZECKI, G., AND STILES, W. *Color Science: Concepts and Methods, Quantitative Data and Formulae*, 2nd ed. John Wiley & Sons, New York, 07 2000.
- [41] ZAMBON, I., CECCHINI, M., EGIDI, G., SAPORITO, M. G., AND COLANTONI, A. Revolution 4.0: Industry vs. agriculture in a future development for smes. *Processes* 7, 1 (2019), 36.
- [42] ZEILER, MATTHEW D. AND FERGUS, R. Visualizing and understanding convolutional networks. In *Computer Vision – ECCV 2014* (Cham, 2014), D. Fleet, T. Pajdla, B. Schiele, and T. Tuytelaars, Eds., Springer International Publishing, pp. 818–833.