Plant Archives Vol. 22, No. 2, 2022 pp. 431-436 e-ISSN:2581-6063 (online), ISSN:0972-5210

Plant Archives

Journal homepage: http://www.plantarchives.org

DOI Url : https://doi.org/10.51470/PLANTARCHIVES.2022.v22.no2.078

A COLOR SPACE BLENDING WITH DEEP LEARNING NETWORKS IN THE IDENTIFICATION OF PLANT LEAVES

C.P. Blesslin Elizabeth[1]* and S. Baulkani[2]

[1] Department of electronics Arunachala College of Engineering, Kanyakumari, Tamilnadu, India

[2]Government College of Engineering, Tirunelveli, Tamilnadu, India

*Corresponding Author E-mail: bless_christo@yahoo.co.in)

(Date of Receiving : 03-08-2022; Date of Acceptance : 14-10-2022)

ABSTRACT

Color is an important feature in applications like the detection of plants and diseases in plants. Deep learning networks utilize optimizers towards improving the accuracy of classification. Color space is treated as an extra dimension with which the image could be better classified. Hence a particular blending of classifier, optimizer and color space is expected to provide enhanced accuracy of classification. There are very rare cases of studies having examined the effect of color space with deep learning networks. Hence, it is motivated to study the role of color spaces. Leaf datasets available in literature have been utilized. Of the few tried networks, Inception V3 is found to perform better with optimizer Adam. Color space XYZ performed better than RGB in the above combination. It has also been tried to obtain majority voting among various optimizer combinations. This solution is also better with XYZ color space. Among the various datasets utilized, consistent performance has been observed with Flavia data set yielding superior classification accuracy.

Keywords: Deep CNN, InceptionV3, Mendeley, Plant Village, XYZ Color space.

Introduction

Machine learning (ML) and Deep learning (DL) networks have been in practice for image classification in various fields of artificial intelligence. Optimizers have been used in combination with deep learning techniques in order to fine-tune the weight factors that enhance the performance of classification. While a large number of applications have been in practice, identification of plant leaves also draws attention as it is costly or unavailable to have a human expert. Color space is treated as an extra dimension with which the image could be better classified. It is possible that a certain blending may perform in a better way in the classification task. The color spaces tried in this work are XYZ, RGB, YCbCr, HSV, YUV, and LAB. The Vein images are extracted similarly to color space conversions, which is another preprocessing technique

In deep learning convolutional networks (DCNN), a good accuracy rate and lower error rate have been achieved in computer vision-based applications such as leaf detection, plant disease identification, etc. Here, each layer of the network is able to learn discriminant features of the input samples such as color, shape, texture, etc, Johnson *et al.* (2021). During the training process, parameters in the network are optimized and trained. The difficulty is that it is not known which feature would be highly dominant for the better performance of the given application. Color is an important feature in applications like the detection of plants and diseases in plants. Every algorithm responds differently to various color spaces, Oza and Kumar (2020). Therefore, it is essential to analyze various color spaces with DCNN.

The Color of an image may be represented by a mathematical model, which is called a color space. It is also called color transformation, De and Pedersen (2021). Since HSV color space is able to distinguish the color of objects in addition to illumination, it has been applied in maize plant detection, Liu *et al.* (2020). According to Patel *et al.* (2021), HSV color space is widely been used in plant classifications. Badiger *et al.* (2019) have observed that in datasets with distinguished illumination information, YCbCr, as well as HSV, performed better than RGB color space. They also have successfully implemented HSV color space in plant disease detection. Johnson *et al.* (2021) explain that XYZ color space is similar to RGB color space with the reasoning that the distribution of RGB color space is overlapping with nearby regions and there is a larger distribution of intensity values in XYZ color space. They also have explained that HSL color space is similar to HSV color space, with the reasoning that the distribution of HSL color space overlaps with adjacent regions and there is a wide distribution of pixel intensity values in HSV color space. They also observed that YCbCr and LAB color spaces sharply distinguished various classes of plant diseases. According to Rangarajan and Purushothaman (2020), HSV color space was traditionally utilized in the studies of plant disease identification, in which feature extraction has been highly effective. Since YCbCr color space has dominant luminance information, it performed better than HSV color space in the classification of plant diseases.

Though many authors have contributed to the above task in various applications including plant diseases, the literature on plant leaves identification with regard to the best DL network and optimizer is still very limited. There are very rare cases of studies having examined the effect of color space with deep learning networks. Hence, it is motivated to study the role of color spaces. Moreover, the view on the best color space to be used is not clear from the literature as experience differs from author to author. Thus, this work deals with the identification of plant leaves utilizing the datasets available in the literature. It is proposed to identify the best blending of DL classifier-optimizer-color space for the considered dataset.

Materials and Methods

Data sets

Readily available datasets in the literature are utilized in the present exercise. The utilized data sets are briefly explained here. Mendeley medicinal leaf dataset consists of 1835 leaves that belong to 30 different species of plants, Roopashree and Anitha (2020). The Flavia dataset contains 1907 leaf images of 32 leaf species, and it is the most popular dataset in leaf recognition, Zhang *et al.* (2020). D Leaf dataset consists of 1290 leaves that belong to 43 plant species, Tan *et al.* (2018). Plantvillage dataset consists of 15,030 healthy leaves belonging to 12 different species, Lee *et al.* (2020).

DL models

The DL models utilized in the present analysis are briefed here. VGGNet is a successor of the AlexNet model. VGGNet has 138 million parameters. There are two variants namely, VGG16 and VGG19. VGG16 architecture has 13 convolution layers and 3 fully connected layers whereas, VGG19 has 16 convolutional layers and three fully connected layers. The convolution layers act as an automated feature extractor that extracts patterns for discriminating each class of plant species. Initial convolution layers learn simple features such as edges which combine these features in the later convolution layers to form complex features. Each convolution layer is generally followed by a non-linear activation layer namely, rectified linear unit (ReLU). ReLU is used to introduce non-linearity for better classification as well as in minimizing computation time, Ghazi *et al.* (2017). The architecture of the VGG16 model is shown in Figure 1. ResNet stands for residual network. It works with 50 layers. Resnet50 architecture consists of 5 stages, each with a convolution and identity block. Each of the blocks has 3 convolution layers. It has over 23 million trainable parameters. ResNet152 has 152 layers. They have an interesting feature of skip connection which initially mitigates the problem of vanishing gradient by alternate shortcut path for the gradient to flow through. It passes the information from the downsampling to upsampling layers such that the higher layer performs well as the lower layer, Albattah *et al.* (2022). Inception V3 model consists of 48 layers and about 24 million parameters. The technique includes factorized convolutions by reducing the number of parameters. Bigger convolutions like (5 x 5 filters) are replaced by two smaller convolutions (3 x 3 filters), thereby the number of parameters gets minimized. There is an auxiliary classifier inserted between the layers in training. The auxiliary classifier acts as a regularizer. By means of pooling operation, grid size reduction is performed, Suh *et al.* (2018).

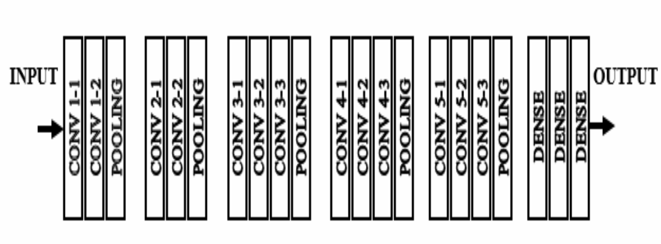


Fig. 1 : Architecture of VGG-16

Optimizers

While mapping inputs to outputs, the optimization algorithm minimizes error by modifying weights and learning rate. Therefore, the loss function gets reduced and the accuracy is improved. The choice of optimizer differs with various types of deep learning models and also with every application. A few optimizers implemented in the present analysis are briefly introduced.

SGD means stochastic gradient descent, in which stochastic means randomness; instead of analyzing the entire dataset, here batches of data are selected. At each iteration, data is shuffled randomly to obtain an approximate minimum. SGD takes more number of iterations to attain local minima. Even though computation time increases, the computational cost is less, Lee *et al.* (2020). Root mean square propagation (Rmsprop) utilizes the magnitude of recent gradient descent to normalize the gradient. The idea of Rmsprop is that for each weight, it keeps the moving average of squared gradients. Then gradient is divided by the square root of the mean square, Turkoglu *et al.* (2019). Adam denotes adaptive moment estimation. On the computation of learning rate, for each parameter, it calculates adaptive learning rates. Both exponentially decaying average of past gradients and the decaying average of past squared gradients were stored. It is called the adaptive learning rate optimization method. Adaptive moment estimation denotes that Adam utilizes the first moment i.e. (mean) and the second moment i.e. (uncentered variance) estimation of gradient, Uguz (2021). Adamax is a variant of Adam optimizer with infinity norm. In each optimization problem, it automatically adapts a separate learning rate for each parameter. The infinite order norm keeps the system stable, Saleem *et al.* (2020). Nadam incorporates Nesterov momentum into Adam. For the calculation of the first moving average, Nesterov momentum is utilized. Momentum term gets modified by updating gradient as parameters are updated, Kamal *et al.* (2019).

Color spaces

Standard RGB color space is the most widely used in computer vision applications. In RGB color space, pixel intensity is stored in terms of red, green, and blue values. Sometimes RGB color space does not differentiate brightness and color values. YUV color space is preferred where light differences are addressed in an image, Cetiner (2021). YCbCr color space stores color and illumination information separately. Y means luma, Cb means (luma-blue color), and Cr means (luma-red color). It is better for human eyes that are highly sensitive to brightness value than color. LAB color space is derived from XYZ, which maps all colors according to human perception. It is a device-independent and perpetually uniform color space, Oza and Kumar (2020). In CIE LAB color space, L denotes lightness, and A and B represent green-red and blue-yellow components respectively. XYZ color space has a wide range of intensity values for all components. Since the intensity values are widely distributed for all the components in XYZ color space, it distinguishes the leaf from the background region, Johnson *et al.* (2021). HSV color space, H means Hue, S for saturation, and V for brightness. It stores luminance and chrominance individually. HSV color space has linear scalability, good in the separation of background, removal of noises, and extraction of contours. S and H components help in the identification of leaf veins, Yang *et al.* (2022).

ANALYSIS PROCEDURE

Images are separated as training and testing samples. Initially, the training samples are input to the DL model. The color space is specified. The hyper-parameters are fine-tuned by the optimizer which is attached to the DL model. When the test samples are supplied, predictions are compared to the known answers of the nearest test sample. The optimizer helps to minimize the error between prediction and the known answer of the nearest trained sample. Figure 2 shows the block diagram of the functioning of DL model with the optimizer. The majority voting classifier (MVC) is able to retain the most appropriate classification and remove redundant data hence only the elite votes are obtained and presented, Suh *et al.* (2018). Figure 3 shows the working of the majority voting classifier.

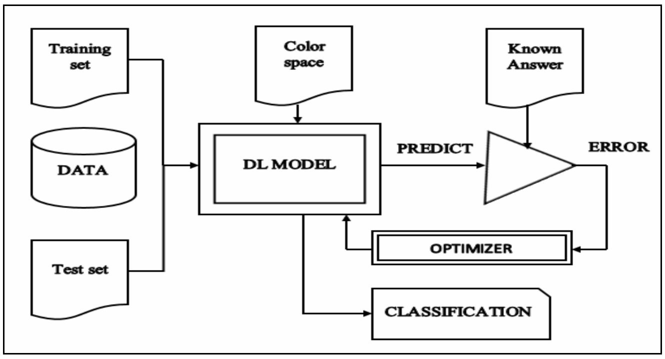
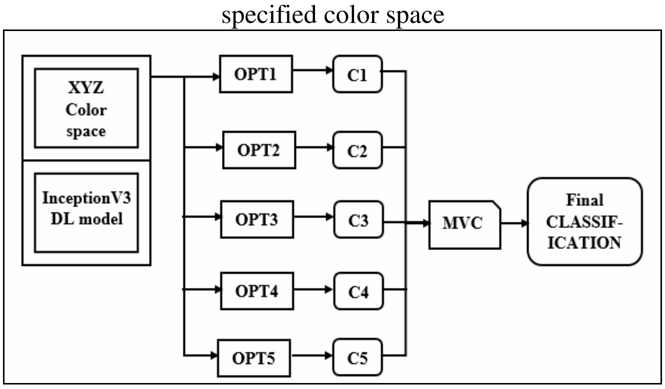


Fig. 2 : Functioning of DL model with optimizer and the specified color space



Legend: OPT1-OPT5 Optimizers; MVC Majority voting classifier

Fig. 3 : Working of Majority voting classifier

The analysis is carried out using the Google Colab facility by Python programming. Python Programming is a high-level programming language that is applied in data science as well as in deep learning algorithms. Some of the libraries used in python are NumPy, Keras, Pandas, Matplotlib, Theano, TensorFlow, and SciPy. TensorFlow is a python library, from which both machine learning and deep learning models experimented. Here TensorFlow tf version 2.8.2 is utilized. Keras is a high-level python library, which is user-friendly and easy to implement deep learning networks. After unzipping the input samples, images are split into training and testing samples using the function ‘ImageDataGenerator’. The models like InceptionV3, VGG16, VGG19, ResNet50, and ResNet152 are imported from the Keras library. The Deep learning model is made to compile with a particular optimizer, which may be Adam, Adamax, Rmsprop, Nadam, and SGD. The program further classifies and calculates accuracy as well as loss. For color space transformation, an image batch processor is utilized, which is one of the applications in MATLAB. Initially, the input samples are converted into various color spaces by means of MATLAB and finally, it is given as input to python programming. For a combination of a DL classifier with specific color space, the performance of various optimizers is studied and the majority voting classifier (MVC) is also invoked for enhanced performance.

Results and Discussions

The performance has been evaluated for the Mendeley dataset with a batch size of 32 and a learning rate of 0.001. On the evaluation of deep learning models VGG16, VGG19, Resnet50, Resnet152, and InceptionV3, with various optimizers, accuracy values are tabulated. From Table 1, it can be seen that the accuracy is the highest for the InceptionV3 model with all the optimizers namely, Adam, Adamax, Nadam, SGD, and Rmsprop. The accuracy values are 0.9953, 0.9936, 0.9964, 0.9931, and 0.9959 respectively. These values are, comparable to the accuracy value of 0.99 obtained in deep learning by Cetiner (2021). Hence, it is motivated to experiment with InceptionV3 for further enhancement.

Table 1: Comparison of Accuracy of DL classifiers with different optimizers (Mendeley data set; Epoch 50; Batch size 32; LR 0.001)

Optimizers	Inception V3	VGG16	VGG19	Res Net50	Res Net152
Adam	0.9953	0.9906	0.9884	0.9749	0.9770
Adamax	0.9936	0.9846	0.9822	0.9773	0.9751
Nadam	0.9964	0.9908	0.9904	0.9766	0.9777
SGD	0.9931	0.9631	0.9575	0.9378	0.9532
RMSprop	0.9959	0.9912	0.9906	0.9742	0.9747

Table 2: Performance of various color spaces with DL classifier-InceptionV3 with Adam optimizer (Mendeley data set; Batch size 32; LR 0.001)

Color spaces	Accuracy	Recall	Precision	F Score
Epoch 20				
RGB	0.9931	0.8970	0.9050	0.8961
XYZ	0.9946	0.9174	0.9255	0.9169
HSV	0.9927	0.8867	0.9032	0.8877
YCbCr	0.9925	0.8859	0.9001	0.8867
LAB	0.9826	0.7365	0.7723	0.7276
YUV	0.9845	0.7639	0.8201	0.7720
VEIN	0.9873	0.8101	0.8335	0.8055
Epoch 50				
RGB	0.9953	0.932	0.9378	0.9302
XYZ	0.9957	0.928	0.9398	0.9304
HSV	0.9929	0.8942	0.8973	0.8912
Ycbcr	0.9934	0.9056	0.9154	0.9039
LAB	0.9852	0.7781	0.8012	0.7783
YUV	0.9856	0.7856	0.8163	0.7871
VEIN	0.9904	0.8621	0.8666	0.8519

Legend: LR-Learning rate

Table 3: Performance of various optimizers with DL classifier-InceptionV3 (Mendeley data set 1835 samples; Epoch 20; Batch size 32; LR 0.001)

Optimisers	Accuracy	Recall	Precision	F Score
RGB				
Adam	0.9931	0.8970	0.9050	0.8961
Adamax	0.9912	0.8644	0.8837	0.8680
Nadam	0.9944	0.9226	0.9206	0.9167
SGD	0.9923	0.8914	0.9018	0.8890
Rmsprop	0.9934	0.9105	0.9218	0.9043
MVC	0.9968	0.9605	0.9587	0.9551
XYZ				
Adam	0.9946	0.9174	0.9255	0.9169
Adamax	0.9919	0.8782	0.8884	0.8787
Nadam	0.9927	0.8941	0.8941	0.8928
SGD	0.9914	0.8686	0.8882	0.8719
Rmsprop	0.9933	0.8959	0.9075	0.8971
MVC	0.9973	0.9631	0.9649	0.9626

Table 4: Performance of various datasets with DL classifier-InceptionV3, Adam Optimizer (Batch size 32; LR 0.001)

DATASETS	Accuracy	Recall	Precision	F Score
RGB-Epoch 20				
Mendeley	0.9931	0.8970	0.9050	0.8961
Flavia	0.9959	0.9306	0.9475	0.9333
DLeaf	0.9942	0.8760	0.8777	0.8659
PlantVillage	0.9953	0.9386	0.9765	0.9527
XYZ-Epoch 20				
Mendeley	0.9946	0.9174	0.9255	0.9169
Flavia	0.9961	0.9351	0.9449	0.9355
DLeaf	0.9922	0.8333	0.8633	0.8334
PlantVillage	0.9950	0.9546	0.9972	0.9631
RGB-Epoch 50				
Mendeley	0.9953	0.9320	0.9378	0.9302
Flavia	0.9976	0.9587	0.9712	0.9613
DLeaf	0.9959	0.9109	0.9248	0.9126
PlantVillage	0.9956	0.9343	0.9739	0.9482
XYZ-Epoch 50				
Mendeley	0.9957	0.9280	0.9398	0.9304
Flavia	0.9980	0.9646	0.9747	0.9668
DLeaf	0.9930	0.8488	0.8661	0.8420
PlantVillage	0.9960	0.9614	0.9756	0.9681

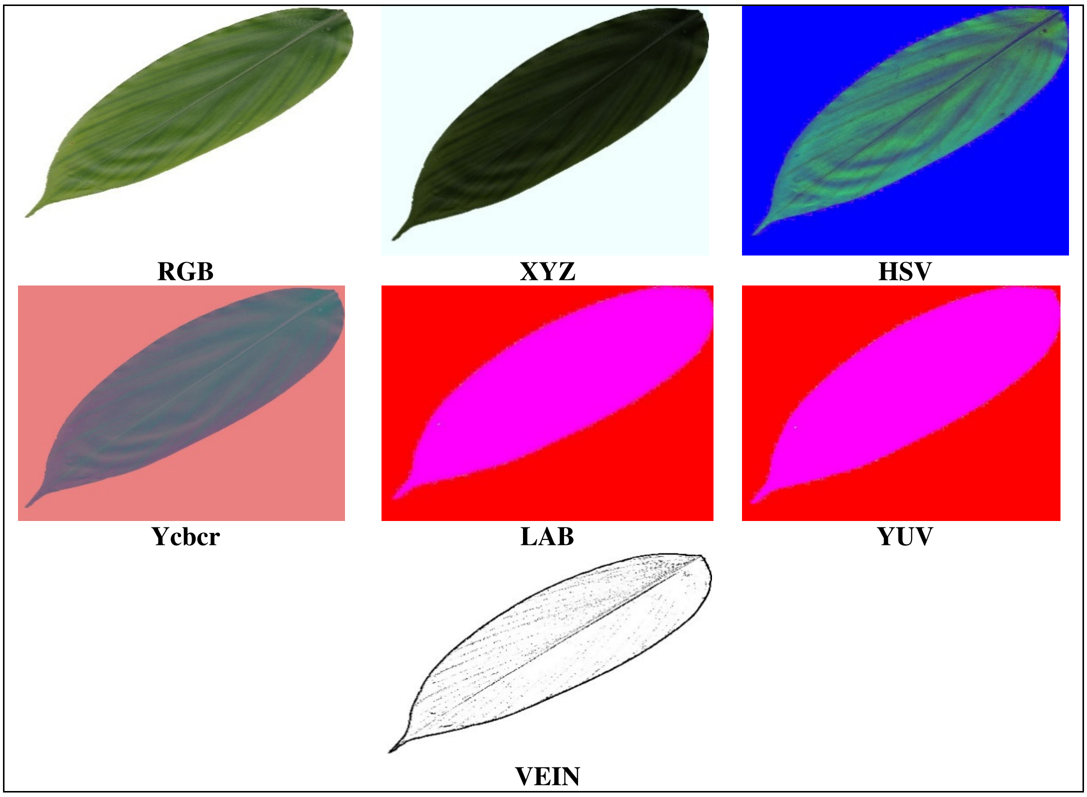


Fig. 4 : Alpinia Galanga Leaf images in various color spaces

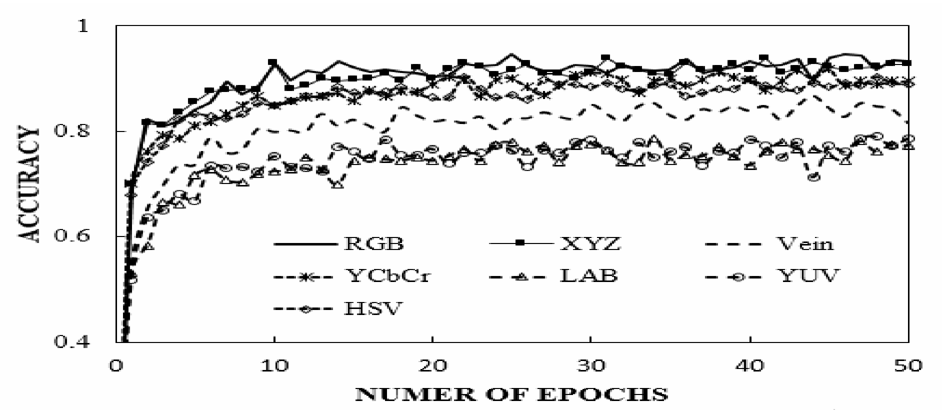


Fig. 5 : Performance of various color spaces with DL classifier-InceptionV3, Adam Optimizer (Batch size 32; LR 0.001)

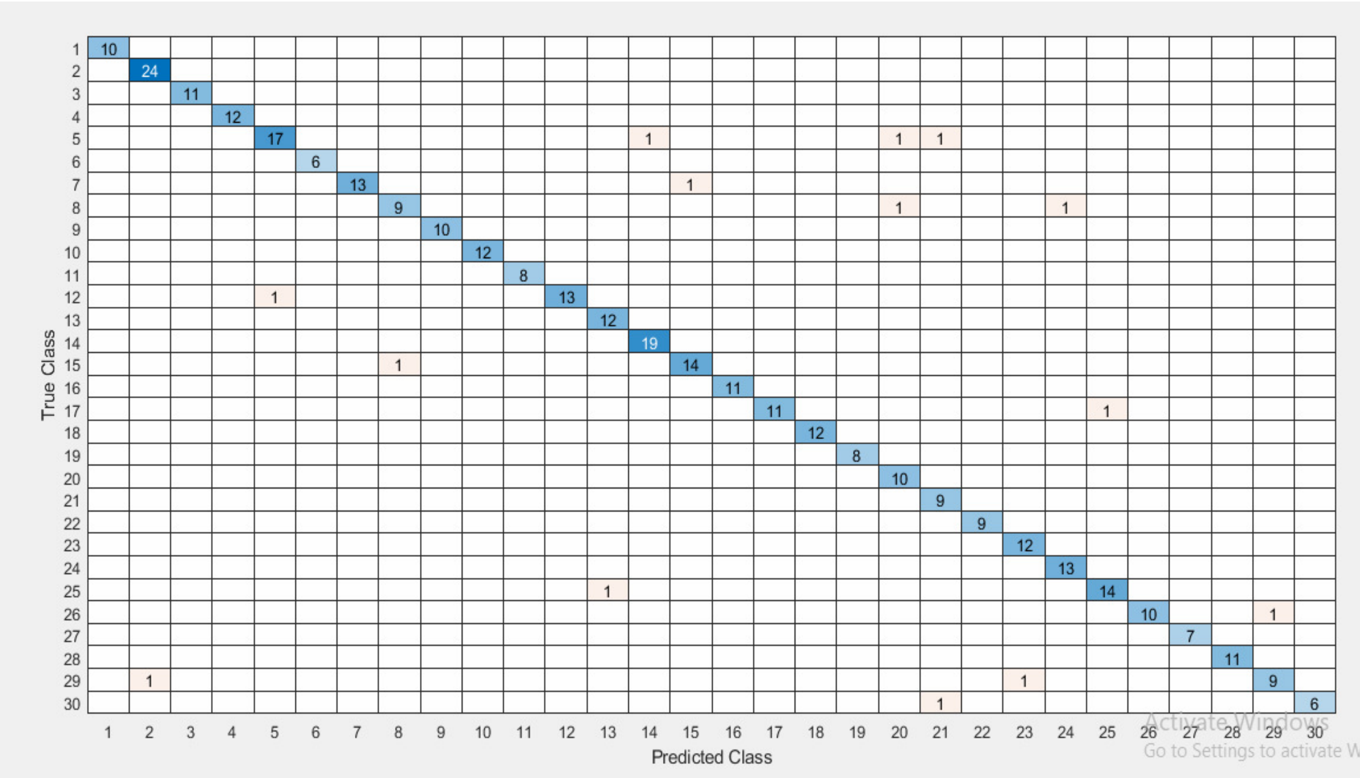


Fig. 6 : Majority voting of all Optimizers in Inception V3 model (XYZ color space)

It is a possible fact that few color spaces would improve the accuracy of classification. All the color spaces mentioned above have been analyzed with the InceptionV3 model of deep learning network with the help of Adam optimizer. Figure 4 shows various color spaces of the leaf Alpinia Galanga (Rasna). The color space models including RGB, YUV, YCbCr, LAB, XYZ, HSV, and vein images have been tried to improve the accuracy. The above analysis was performed for 20 epochs as well as for 50 epochs. Figure 5 shows a graphical view of relative performance with all the color spaces. From Table 2 it is found that epoch 50 produces better results compared to epoch 20. Table 2 presents the performance measures namely, accuracy, recall, precision, and F-score relevant to the InceptionV3 model with Adam optimizer using seven different color spaces. The formulae to calculate these performance metrics from the confusion matrix may be referred to from the literature, Suh *et al.* (2018). Out of all color spaces, XYZ performed well with the highest accuracy of 0.9957. Correspondingly, larger values of precision and recall denote better classification. Precision is the ratio of correctly classified positive predictions to the sum of positive predictions. The best precision of 0.9398 has been obtained by XYZ color space. The recall of XYZ color space is 0.928 which is slightly lower than that with RGB color space. The recall is the ratio of correctly classified positive predictions to the total number of all predictions. F score is also an important measure in plant recognition since it is the harmonic mean of precision and recall, Kazerouni *et al.* (2019). The highest F score of 0.9304 has been obtained with XYZ color space among all color spaces.

Since XYZ is found to be performing better than other color spaces, the performance of various optimizers has been analyzed with DL classifier InceptionV3. For comparison, results with RGB as well as XYZ color space are presented in Table 3. Results correspond to 20 epochs. The majority voting of all optimizers produced better results compared to the individual optimizers. On Majority voting of all optimizers with XYZ color space, the accuracy is increased to 0.9973, whereas majority voting of optimizers with RGB color space is 0.9968. Figure 6 shows the confusion matrix on majority voting of all optimizers with the Inception V3 model utilizing XYZ color space. Only a few samples are misclassified with a majority voting classifier compared with individual optimizers. For XYZ color space, maximum accuracy is obtained as 0.9946 with Adam optimizer, and with majority voting classifier, it is increased to 0.9973.

Further, the performance of the Inception V3 model with optimizer Adam has been explored through various datasets such as Mendeley, Flavia, DLeaf, and PlantVillage, with batch size 32 and a learning rate of 0.001. Epochs 20 and 50 have been tried and found that analysis with epochs 50 has given better performance as presented in Table 4. The XYZ color space produced better results than the RGB color space. Though the performance is comparable with all the data sets, the Flavia data set has a consistently higher performance. The accuracy of datasets may differ due to the quality of images, the number of training and testing samples, etc.

Conclusion

Since plant recognition highly depends on color information, the role of color space has been analyzed with the deep learning models VGG16, VGG19, Resnet50, Resnet152, and InceptionV3. InceptionV3 is better among the considered DL network in combination with the optimizer Adam. Images are better classified with XYZ color space compared to RGB color space. InceptionV3-Adam optimizer-InceptionV3 blend has been found to be the best for plant recognition. It is observed that the performance with 50 epochs is higher than that with 20 epochs. Majority voting of all the optimizers studied has given still better performance with InceptioV3 classifier; it yields higher accuracy of 0.9973 with XYZ color space. Among the various datasets examined, consistent performance has been observed with Flavia data set yielding superior classification accuracy.

References

Albattah, W.; Nawaz, M.; Javed, A.; Masood, M. and Albahli, S. (2022). A novel deep learning method for detection and classification of plant diseases, Complex & Intelligent Systems, 8(1): 507-524.

Badiger, M.; Kumara, V.; Shetty, S.C. and Poojary, S. (2022). Leaf and Skin Disease Detection using Image Processing, Global Transitions Proceedings, 1-11.

Çetiner, H. (2021). Classification of apple leaf diseases using the proposed convolution neural network approach, 9(4): 1130-40.

De, K. and Pedersen, M. (2021). Impact of colour on robustness of deep neural network, In Proceedings of the IEEE/CVF International Conference on Computer Vision, 21-30.

Fathi Kazerouni, M.; Mohammed Saeed, N.T. and Kuhnert, K.D. (2019). Fully-automatic natural plant recognition system using deep neural network for dynamic outdoor environments, *SN Applied Sciences*, 1(7): 1-18.

Ghazi, M.M.; Yanikoglu, B. and Aptoula, E. (2017). Plant identification using deep neural networks via optimization of transfer learning parameters, *Neurocomputing*, 235: 228-235.

Johnson, J.; Sharma, G.; Srinivasan, S.; Masakapalli, S.K.; Sharma, S.; Sharma, J. and Dua, V.K. (2021). Enhanced field-based detection of potato blight in complex backgrounds using deep learning, *Plant Phenomics*, 1-13.

Kamal, K.C.; Yin, Z.; Wu, M. and Wu, Z. (2019). Depthwise separable convolution architectures for plant disease classification, *Computers and Electronics in Agriculture*, 165: 1-5.

Krishnaswamy Rangarajan, A. and Purushothaman, R. (2020). Disease classification in eggplant using pre-trained VGG16 and MSVM, Scientific reports, 10(1): 1-11.

Lee, S.H.; Goeau, H.; Bonnet, P. and Joly, A. (2020). "New perspectives on plant disease characterization based on deep learning, *Computers and Electronics in Agriculture*, 170: 1-12.

Liu, H.; Sun, H.; Li, M. and Iida, M. (2020). Application of color featuring and deep learning in maize plant detection, *Remote Sensing*, 12(14): 1-21.

Oza, U. and Kumar, P. (2020). Empirical examination of color spaces in deep convolutional networks", *Int. J Recent Technology and Engineering*, 9(2): 1011-1018.

Patel, S.; Oza, U. and Kumar, P. (2021). Is deep learning image classification invariant to color space transforms.

Roopashree, S. and Anitha, J. (2020). Medicinal leaf dataset, Mendeley Data, 1.

Saleem, M.H.; Potgieter, J. and Arif, K.M. (2020). Plant disease classification: A comparative evaluation of convolutional neural networks and deep learning optimizers, *Plants*, 9(10): 1-16.

Suh, H.K.; Ijsselmuiden, J.; Hofstee, J.W. and Van Henten, E.J. (2018). Transfer learning for the classification of sugar beet and volunteer potato under field conditions, *Biosystems engineering*, 174: 50-65.

Turkoglu, M.; Hanbay, D. and Sengur, A. (2019). Multi-model LSTM-based convolutional neural networks for detection of apple diseases and pests, *Journal of Ambient Intelligence and Humanized Computing*, 1-11.

Uğuz, S. and Uysal, N. (2021). Classification of olive leaf diseases using deep convolutional neural networks, *Neural Computing and Applications*, *33*(9): 4133-4149.

Wei Tan, J.; Chang, S.W.; Abdul-Kareem, S.; Yap, H.J. and Yong, K.T. (2018). Deep learning for plant species classification using leaf vein morphometric, IEEE/ACM transactions on computational biology and bioinformatics, 17(1): 82-90.

Yang, X.; Ni, H.; Li, J.; Lv, J.; Mu, H. and Qi, D. (2022). Leaf recognition using BP-RBF hybrid neural network, *Journal of Forestry Research*, 33(2): 579-89.

Zhang, Y.; Cui, J.; Wang, Z.; Kang, J. and Min, Y. (2020). Leaf image recognition based on bag of features, *Applied Sciences*, 10(15): 1-17.