2023 International Conferences on Data Science, Agents, and Artificial Intelligence (ICDSAAI)

Deep Learning-Based Classification of Flowers Using FlowerNet Model

Nagaraj P
Department of Computer Science and Engineering
Kalasalingam Academy of Research and Education
Krishnankoil, Virudhunagar, India
nagaraj.p@klu.ac.in

Muneeswaran. V
Department of Electronics and Communication Engineering
Kalasalingam Academy of Research and Education
Krishnankoil, Virudhunagar, India
munees.klu@gmail.com

M. Raja
Department of Computer Science and Engineering
Kalasalingam Academy of Research and Education
Krishnankoil, Virudhunagar, India
mraja@klu.ac.in

Visal J
Department of Computer Science and Engineering
Kalasalingam Academy of Research and Education
Krishnankoil, Virudhunagar, India
visalpandian001@gmail.com

Betham Raj Kumar
Department of Computer Science and Engineering
Kalasalingam Academy of Research and Education
Krishnankoil, Virudhunagar, India
rajkumarbetham7@gmail.com

Pendyala Raja Yaswanth
Department of Computer Science and Engineering
Kalasalingam Academy of Research and Education
Krishnankoil, Virudhunagar, India
rajayaswanthpendyala@gmail.com

2023 International Conference on Data Science, Agents & Artificial Intelligence (ICDSAAI) | 979-8-3503-4891-0/23/$31.00 ©2023 IEEE | DOI: 10.1109/ICDSAAI59313.2023.10452559

***Abstract*—**

Achieving the rapid and accurate detection of flowers in the natural environment is not easy, the study of floral classification systems is a crucial topic. However, there are still some difficulties in the recognition of flower photos due to the complicated backgrounds of flowers, the similarities between the many kinds of flowers, and the differences within the same species of flowers. The conventional classification of flowers is mostly based on the three characteristics of color, shape, and texture; however, the accuracy of this classification is not very high.

***Keywords*—Deep learning, Convolutional Neural Networks, Flower Classification, Flower Net, Optimization.**

I. INTRODUCTION

A crucial piece of botany research is the classification of flowers. Flowers are one of the most abundant species in the world, and it has been discovered that there are hundreds of thousands of different varieties of flower species [1]. As economies and technologies advance, more and more individuals are discovering how much they like traveling to different places during the holidays. At the same time, people photograph flowers using cameras, smartphones, and other tools, but because they are unfamiliar with the different types of flowers, people are difficult to identify. As a result, the creation of a flower classifier will also be quite enjoyable for individuals [2]. To distinguish between different species of flowers, we cannot simply on the same characteristic, such as color, shape, or texture. Even the same species of flowers will differ depending on the shape, scale, viewpoint, and other factors. This presents some difficulties in the field of flower classification [3].

II. MOTIVATION

The most distinctive aspect of a plant is its flower. Recognizing flowers can therefore aid in learning more about the plant. Our main motivation is to build a model to recognize the different types of flowers. In this world, there are different species of flowers are there. A plant's flower is its most noticeable and perceptible feature, a topic of considerable study by botanists, and frequently the key to identifying a species. The main goal is to help the tourist landlords, peasants, formers, etc.…. to identify the flowers.

III. RELATED WORKS

Hiary et al. [4] proposed the classification of flower species which involved the use of numerous models, approaches, and methods. A deep learning model with two steps was represented. Automatically localizing the flower-filled areas of the image was the first stage. The segmentation of the photos was done for this reason. They used Softmax to gather the features in the first stage and then categorized them in the second step. The achievement rate was 97.1%.

Two datasets were used by Cibuk et al [5]. For feature extraction, VGG-16 and AlexNet models were applied. The features collected from these models were blended. They then used the minimum redundancy maximum relevancy (mRMR) technique to choose features. They used SVM to categorize the effective features. 96.39% of attempts were successful.

Dias et al. [6] described a novel CNN-based detection method. They uniformly divided the images and used the superpixel technique to add more pixels. The goal was to create more effective features. They then used a tweaked CNN model to accomplish feature extraction. SVM was employed as a classifier by them. The reported success rate was 90%.

For the classification of floral species, Ghazi et al. [7] used pre-trained AlexNet, GoogLeNet, and VGG-16 CNNs. To categorize the features gleaned from the models, they employed Softmax. They used the image augmentation technique to create the new augmented image dataset before carrying out the classification process. The achievement rate was 80%.

For the classification of flowers, Seeland et al. [8] advocated the use of conventional techniques. They made use of feature selection detectors, local shape and color identifiers, and these. The SVM classifier had a 94% success rate.

In the work of Mohanty et al. [9] the histogram threshold-based approach was used to segment images that resembled flowers. They next used the Gray Level Co-Occurrence Matrix (GLCM) approach to extract the texture features from each image. They categorized floral species using a genetic algorithm-based method. Here the mean success rate was 96.32%.

In their research, Tian et al. [10] classified 17 different species of flowers. For the floral photos, they applied the data augmentation technique. They applied the proposed CNN model to augmented data. The categorization algorithm in this study was built using the Softmax function. Here 92% of the classifications were successful.

Alaslani [11] proposes a thorough investigation based on feature selection techniques for the classification of flowers. The feature extraction approach in the proposed study used the AlexNet, GoogLeNet, ResNet-50, and VGG-16 CNN models. The transfer learning method is used to train these models. The fact that these models perform well in the ImageNet competition is the key factor in the decision. The four CNN models all include layers that may produce a total of 1000 features, which is what unites them. The CNN model's layers are FC-8, loss-3 classifier, FC-1000, and FC-8, as detailed in the related works' CNN model sequence.

Kadhim and Abed [12] proposed the architectures' last layer's features were extracted and blended. A fresh feature set of 4000 features was produced. The following step was applying f-regression and multiple inclusion criterion (MIC) approaches to choose the effective features. With the aforementioned feature selection techniques, two new feature sets were consequently produced. The third step was extracting the shared (intersecting) characteristics of these two clusters. The SVM approach was then used to categorize these features. As a result, it can be seen from this study that the features acquired by the intersection of the set of features help to boost classification success.

Buda et al. [13] proposed the amount of data in each class was equalized for the study's practical applications. A total of 3670 photographs from all classes were used, with 734 images from each class being used. In flower classes having more than 734 photos, the image selection was carried out at random. The dataset for this study was changed to have 80% and 20% rates for the training and test sets in all experimental steps.

Huk et al. [14] stated the common eligibility across the AlexNet, GoogLeNet, ResNet-50, and VGG-16 architectures is that each of these networks has a layer that can provide 1,000 characteristics for defining the input image. These four models have established themselves in this industry by winning ImageNet competitions. The aforementioned CNN models were chosen for this study since their capacity for generalization was demonstrated in ImageNet contests. Also, they were chosen for this investigation since it is believed that the distinctive element of the features derived from the models with varied architectures is stronger.

Toğaçar et al. [15] proposed the construction of the GoogleNet model is intricate. Block stacks, pooling layers, and convolutional layers make up the model's overall structure. The GoogLeNet model's operating concept is that features taken from images are layered from network to network. Three Softmax classifiers are used by the network-to-network due to their high depth to ensure that training-related mistakes are propagated back. Nonetheless, two additional classifiers (Softmax) are still used throughout the testing phase.

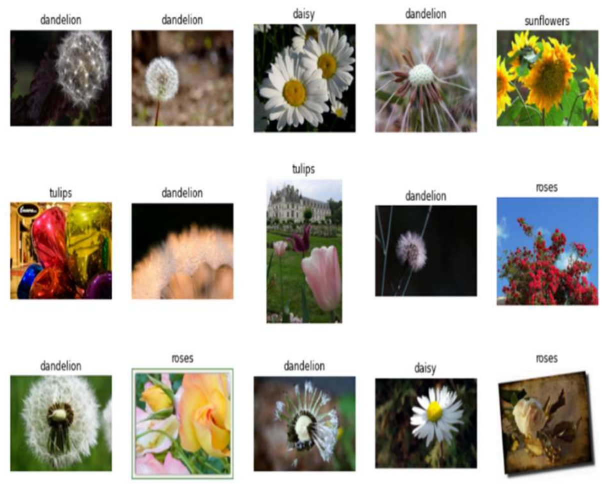
Toğaçar et al. [16] uses ResNet-50 persistent connections in its architecture and has a deep network topology that can successfully deal with the problem of backpropagation. In other words, the ResNet-50 model is made up of residual blocks that either directly feed the layer above or below the current layer in the network structure. 50 leftover blocks are included in this architecture.

Bae et al. [17] analyzed that the Residual blocks are not present in the VGG-16 model. The layers of the AlexNet model serve as the general framework for this model (convolutional layer, pooling layer, etc.). There are a total of 21 layers in it. The fact that the VGG-16 design has a growing network structure sets it apart from AlexNet architecture more than any other aspect. Three by three-pixel or five by five-pixel filters are chosen in convolutional layers and moved over the input image. The FC layers are where the probability values for the characteristics of this design are calculated.

IV. Materials and Methods

A. Dataset Description

we have used a flower dataset, which consists of around 4242 images of five different classes. The dataset was taken from Kaggle. It has a daisy, dandelion, sunflower, rose, and tulip. For each class, there are about 800 photos. Photos are not high resolution, about 320x240 pixels. They are not reduced to a single size, as they have different proportions [18]. Fig.1 illustrates the information about the data set used in this research.




```
Downloading flower images from http://download.tensorflow.org/example_images/flower_photos.tgz...
Flower photos are located in ./flower_photos

The dataset has 5 label classes: odict_values(['daisy', 'dandelion', 'roses', 'sunflowers', 'tulips'])
There are 2934 training images
there are 736 test images
```

Fig.1 Dataset Information

B. CNN MODELS

Convolutional neural networks (CNN/ConvNet) are a class of deep neural networks used most frequently to interpret visual data in deep learning. Normally, matrix multiplications come to mind when we think of a neural network, but that is not the case with ConvNet. It makes use of a unique method called convolution. Convolution is a mathematical procedure that takes two functions and creates a third function that expresses how the shape of one is changed by the other in mathematics. Artificial neurons are arranged in numerous layers to form convolutional neural networks. Artificial neurons are mathematical functions that compute the weighted sum of several inputs and output an activation value, roughly imitating their biological counterparts. Each layer of a ConvNet creates several activation functions that are passed on to the following layer when an image is entered. Typically, the first layer extracts fundamental features like edges that run horizontally or diagonally. The following layer receives this output and detects more intricate features like corners or multiple edges. The network may recognize increasingly more complex elements, including objects, faces, etc., as we go further into it [19].

C. SIMPLE CNN MODEL

We are arranging the layers one after another in sequential order. Then we apply the convoluted matrix for two-dimensional with a filter size of 16*16, with stride value 3, and with the same padding size with an activation function of Leaky Relu. After the Convolution layer work is completed, we go on with the Pooling layer (i.e., Max Pooling) [20]. A dense layer with a filter size of 128 and an activation function with a Leaky Relu optimizer is used. Finally, the output layer with 26 neurons and a Softmax optimizer is used for multi-class classification. At compilation, we go on with Adam optimizer and,sparse_categorical_crossentropy for loss and with matrices accuracy. Fig.2 describes the simple CNN model.

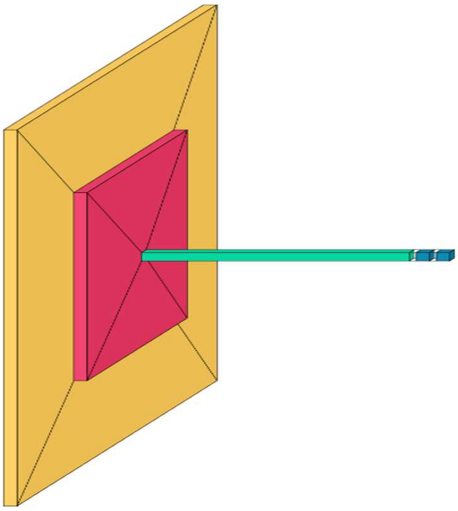


Fig.2 A Simple CNN Model

D. CNN MODEL WITH DATA AUGMENTATION AND DROPOUT LAYERS

Sequential order is performed. Initially, at pre-processing we randomly zoom the images with size 0.3 and increase the dataset. Then we apply the convoluted matrix for two-dimensional with a filter size of 16*16, with stride value 3 and with the same padding size with an activation function of Leaky Relu followed by Max Pooling [21]. Once again, we are doing the same to reduce the dimensions. Then comes the flattening. A dense layer with activation functions and with some dropout layer (i.e.for increasing accuracy) is used. Finally, the output layer with 26 neurons and a softmax optimizer is used for multi-class classification. At compilation, we go on with Adam optimizer and,sparse_categorical_crossentropy for loss and with matrices accuracy. Fig.3 describes the CNN model with data augmentation and dropout layers.

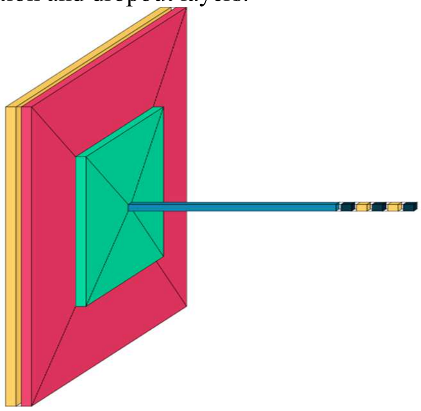


Fig.3 A CNN Model with Data Augmentation and Dropout Layers

E. CNN WITH ONLY DROPOUT LAYERS

In the sequential order, we are straightly going on with the Pooling layer (Max Pooling), followed by Flattening. Dense and Dropout layers are performed one after another with leaky relu with activation function in the Dense layer [22]. Finally, the compilation is done. Fig.4 describes the CNN Model with only Dropout Layers.

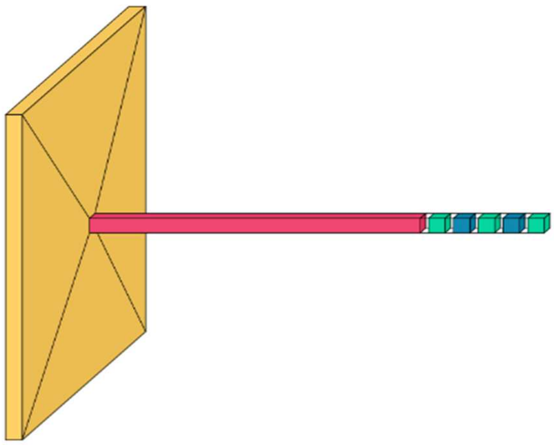


Fig.4 A CNN Model with only Dropout Layers

F. CNN WITH ONLY AUGMENTATION LAYERS

Going on in the sequential layers and proceeding with random zoom pre-processing by increasing the dataset. Then we apply the convoluted matrix for two-dimensional with a filter size of 16*16, with stride value 3 and with the same padding size with an activation function of Leaky Relu, followed by 2D Max Pooling. Flattening is done. Then we are going on with some dense layers with appropriate activation functions. End up with 26 neurons in the output layer with a softmax activation function for classifying the output [23]. At compilation, we go on with Adam optimizer and,sparse_categorical_crossentropy for loss and with matrices accuracy. Fig.5 describes the CNN model with only data augmentation.

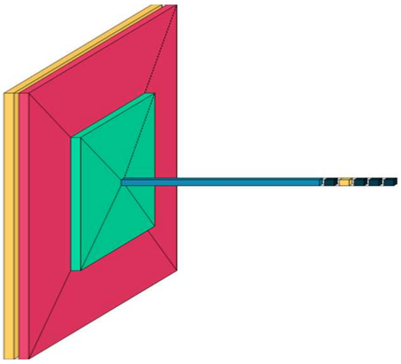


Fig.5 A CNN Model with Only Data Augmentation

G. CNN WITH TWO DIFFERENT AUGMENTATION LAYERS

In the correct format, at augmentation, we are going with random zoom and random flip (vertical and horizontal) with a 2D convolution layer (16 neurons with stride value 3 and with padding size same), after comes the 2DPooling [24]. Flattening is done after that dense layer and the output layer is performed. At compilation, we go on with Adam optimizer and,sparse_categorical_crossentropy for loss and with matrices accuracy. Fig.6 describes the CNN model with two data augmentation.

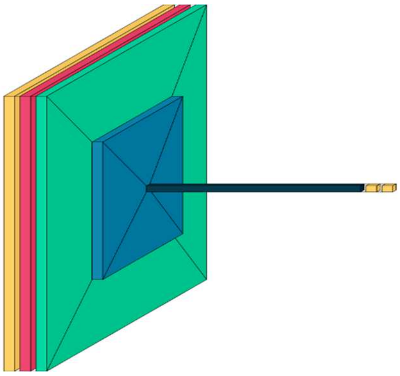


Fig.6 A CNN Model with Two Data Augmentation

H. CNN MODEL WITH DATA ARGUMENTATION LAYERS AND MORE DENSE LAYERS

In the correct format, at augmentation, we are going with random zoom and random flip (vertical and horizontal) with a 2D convolution layer(16 neurons with stride value 3 and with padding size same), after comes the 2DPooling.Flattening is done after that dense layer and the output layer is performed. Additional Dense layers are performed [24] At compilation, we go on with Adam optimizer and,sparse_categorical_crossentropy for loss and with matrices accuracy. Fig.7 describes the CNN model with two data augmentation with more dense layers.

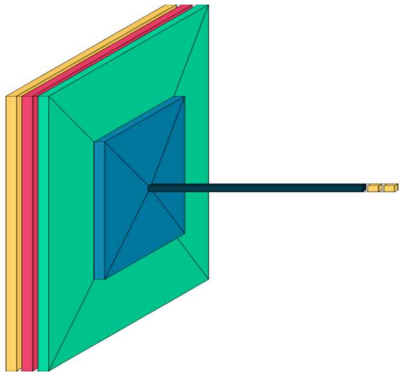


Fig.7 A CNN Model with Two Data Augmentation with Dense Layer

I. CNN WITH FILTER IN ALL LAYERS

Sequential order is used. Convolution is done to compress the image and pooling is done to reduce the dimension. Using convolution and pooling again and again followed by flattening with a dense layer and finally comes the output layer [24]. At compilation, we go on with Adam optimizer and,sparse_categorical_crossentropy for loss and with matrices accuracy. Fig.8 describes the CNN model with filters with all input and output layers.

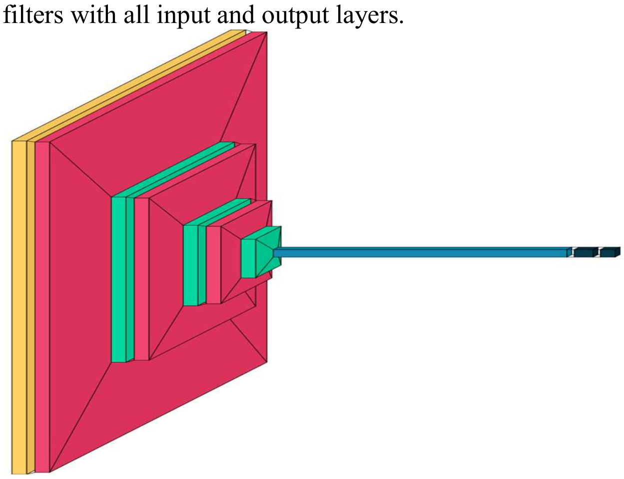
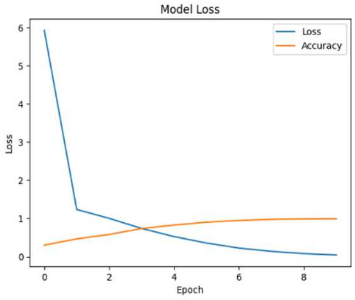
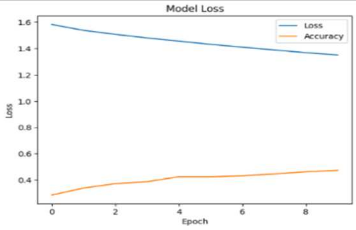
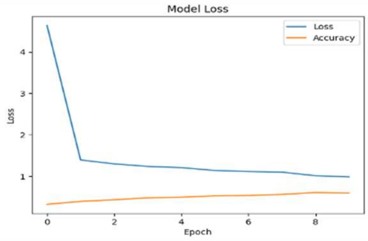
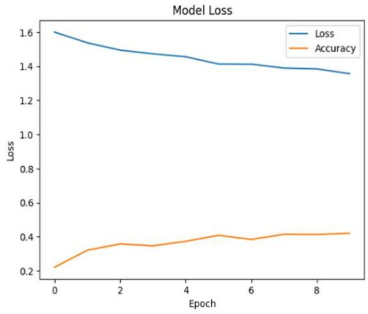
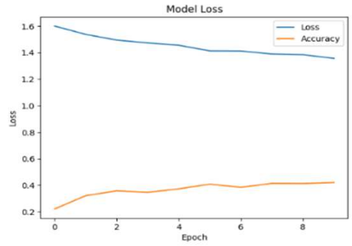
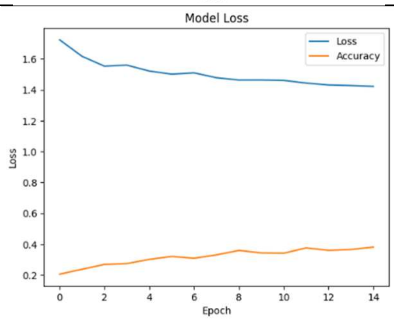


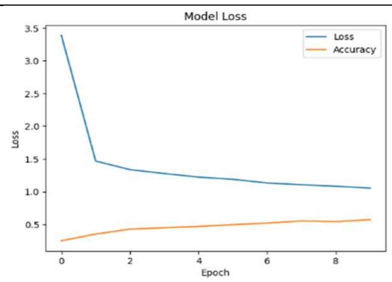
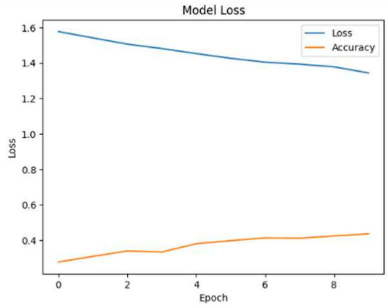
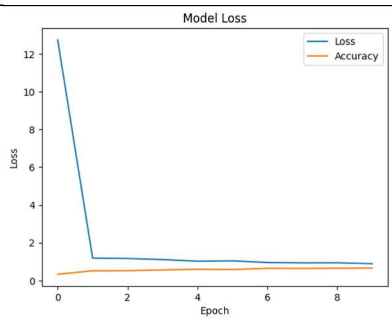
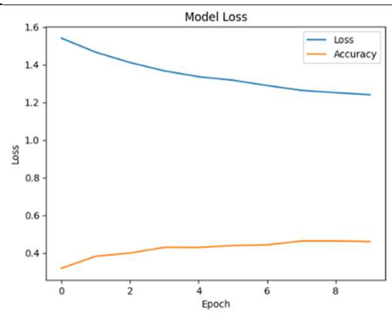
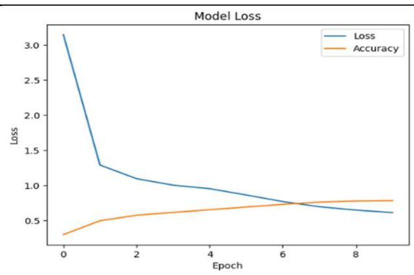
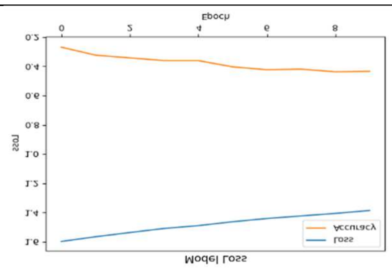
Fig.8 A CNN Model with Filters in All Layers

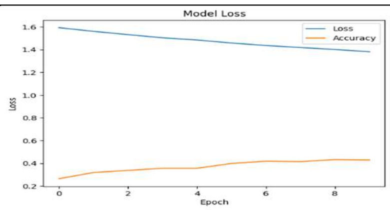
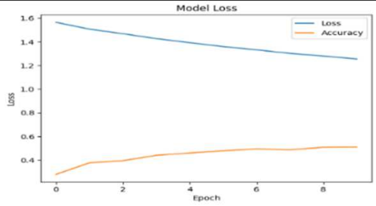
V. EXPERIMENTAL RESULTS

With the help of numpy, pandas, matplotlib. pyplot, torch, and keras packages which we had installed in Google Colab (python platform) and other machine learning and deep learning libraries with various optimizers and various activation functions, we had to get better accuracy (accuracy for 100 epoch and validation accuracy) in terms of both time and memory. For our seven different models, we went with Adam and Adadelta Optimizer. From our experiment, we can conclude that accuracy keeps on increasing, as we increase the epochs. As of our understanding in our graph, our model doesn't lead to overfitting or underfitting. Our model will play a vital role in the field of agriculture, helping farmers, tourists, peasants, landlords, and pharmacists to identify unknown flowers, and beware of the poisonous flower. The below Table2 comprises all the optimization techniques with all Deep CNN Models.

Table1. Optimization techniques Vs. Deep CNN Techniques

MODEL	OPTIMIZERS	EPOCH vs LOSS	DESCRIPTION	ACCURACY FOR 100 EPOCHS	VALIDATION ACCURACY
SIMPLE CNN	ADAM		Accuracy is kept on increasing as we increase the number of epochs	99%	53%
	ADADELTA		Accuracy increases and loss decreases with epoch	52%	49%
CNN MODEL WITH DATA AUGMENTATION AND DROPOUT LAYERS	ADAM		Loss reaches a local minimum, and it decreases in opposite to accuracy	47%	27%
	ADADELTA		Accuracy and loss are inversely proportional to its epochs	54%	52%
. CNN WITH ONLY DROPOUT LAYERS	ADAM		Accuracy and loss are inversely proportional to its epoch	23%	20%
	ADADELTA		Accuracy keeps on increasing as we increase the number of epochs	37%	34%

CNN WITH ONLY AUGMENTATION LAYERS	ADAM		Loss reaches a local minimum, and it decreases, Accuracy keeps on increasing as we increase the number of epochs	61%	57%
	ADADELTA		Accuracy keeps on increasing as we increase the number of epochs	58%	54%
CNN WITH TWO DIFFERENT AUGMENTATION LAYERS	ADAM		Loss reaches a local minimum, and it decreases, Accuracy slightly increases	67%	57%
	ADADELTA		Accuracy and loss are inversely proportional	43%	46%
CNN MODEL WITH DATA ARGUMENTATION LAYERS AND MORE DENSE LAYERS	ADAM		Loss is going to local minima and decreases and accuracy increases	79%	60%
	ADADELTA		Accuracy is getting down and loss increases as we increase the number of epochs	46%	42%

CNN WITH FILTER IN ALL LAYERS	ADAM		Accuracy and loss are inversely proportional	97%	57%
	ADADELTA		Loss decreases, Accuracy increases	52%	48%

VI. CONCLUSION AND FUTURE WORK

Image processing technique plays an important role in flower classification. The use of a neural network classifier for flower classification using DWT and GLCM has been demonstrated. Only gray-level features have been used. Classifiers play an important role in testing the data and checking the accuracy of the classification algorithm. Identifying different flower images based on their surface parameter is a challenging and expensive task. Flower image surface parameters are color and texture. Beyond these aversion techniques, our deep learning with various activation functions and optimizers will give better accuracy.

REFERENCES

[1] Peryanto, A., Yudhana, A., & Umar, R. (2022). Convolutional neural network and support vector machine in classification of flower images. Khazanah Informatika: Jurnal Ilmu Komputer dan Informatika, 8(1), 1-7.

[2] Xia, X., Xu, C., & Nan, B. (2017, June). Inception-v3 for flower classification. In 2017 2nd international conference on image, vision and computing (ICIVC) (pp. 783-787). IEEE.

[3] Solanki, A., & Singh, T. (2022). Flower species detection system using deep convolutional neural networks. In Emerging Technologies for Computing, Communication and Smart Cities: Proceedings of ETCCS 2021 (pp. 217-231). Singapore: Springer Nature Singapore.

[4] Hiary, H., Saadeh, H., Saadeh, M., & Yaqub, M. (2018). Flower classification using deep convolutional neural networks. IET Computer Vision, 12(6), 855-862.

[5] Cıbuk, M., Budak, U., Guo, Y., Ince, M. C., & Sengur, A. (2019). Efficient deep features selections and classification for flower species recognition. Measurement, 137, 7-13.

[6] Dias, P. A., Tabb, A., & Medeiros, H. (2018). Apple flower detection using deep convolutional networks. Computers in Industry, 99,17-28.

[7] Ghazi, M. M., Yanikoglu, B., & Aptoula, E. (2017). Plant identification using deep neural networks via optimization of transfer learning parameters. Neurocomputing, 235, 228-235.

[8] Seeland, M., Rzanny, M., Alaqraa, N., Wäldchen, J., & Mäder, P. (2017). Plant species classification using flower images—A comparative study of local feature representations. PloS one, 12(2), e0170629.

[9] Mohanty, A. K., & Bag, A. (2017). Image mining for flower classification by genetic association rule mining using GLCM features. International Journal of Advanced engineering, Management and Science, 3(5), 239846.

[10] Tian, M., Chen, H., & Wang, Q. (2019, June). Flower identification based on Deep Learning. In Journal of Physics: Conference Series (Vol. 1237, No. 2, p. 022060). IOP Publishing.

[11] Alaslani, M. G. (2018). Convolutional neural network based feature extraction for iris recognition. International Journal of Computer Science & Information Technology (IJCSIT) Vol, 10.

[12] Kadhim, M. A., & Abed, M. H. (2020). Convolutional neural network for satellite image classification. Intelligent Information and Database Systems: Recent Developments 11, 165-178.

[13] Buda, M., Maki, A., & Mazurowski, M. A. (2018). A systematic study of the class imbalance problem in convolutional neural networks. Neural networks, 106, 249-259.

[14] Huk, M., Maleszka, M., & Szczerbicki, E. (Eds.). (2020). Intelligent information and database systems: Recent developments. Springer International Publishing.

[15] Toğaçar, M., Ergen, B., & Cömert, Z. (2020). Classification of flower species by using features extracted from the intersection of feature selection methods in convolutional neural network models. Measurement, 158, 107703. 14. Tanaka, Y., & Kageyama, Y. ImageNet/ResNet-50 Training in 224 Seconds.

[16] Toğaçar, M., Özkurt, K. B., Ergen, B., & Cömert, Z. (2020). BreastNet: A novel convolutional neural network model through histopathological images for the diagnosis of breast cancer. Physica A: Statistical Mechanics and its Applications, 545, 123592

[17] Bae, K. I., Park, J., Lee, J., Lee, Y., & Lim, C. (2020). Flower classification with modified multimodal convolutional neural networks. Expert Systems with Applications, 159, 113455.

[18] https://www.kaggle.com/datasets/alxmamaev/flowers-recognition

[19] Nagaraj, P., Muneeswaran, V., Sunethra, B., Sreeya, C., Dhannushree, U., & Nithisaa, S. (2023, January). A Comparative Analysis of Retinal Disease Image Classification for OCT Using Deep Learning Techniques. In *2023 International Conference on Computer Communication and Informatics (ICCCI)* (pp. 1-10). IEEE.

[20] Sudar, K. M., Nagaraj, P., Yeshwanth, K. V., Kumar, Y. D., Kumar, V. S. J., & Reddy, V. N. S. (2022, May). Recognitionof Diseases in Paddy using Deep Learning. In *2022 6th International Conference on Intelligent Computing and Control Systems (ICICCS)* (pp. 1458-1463). IEEE.

[21] Vb, S. K. (2020). Perceptual image super resolution using deep learning and super resolution convolution neural networks (SRCNN). *Intelligent Systems and Computer Technology*, *37*(3).

[22] Nagaraj, P., Rao, J. S., Muneeswaran, V., & Kumar, A. S. (2020, May). Competent ultra data compression by enhanced features excerption using deep learning techniques. In *2020 4th International Conference on Intelligent Computing and Control Systems (ICICCS)* (pp. 1061-1066). IEEE.

[23] Nagaraj, P., Muneeswaran, V., Jeyanathan, J. S., Panda, B., & Bhoi, A. K. (2023). Optimized TSA ResNet Architecture with TSH—Discriminatory Features for Kidney Stone Classification from QUS Images. In *Enabling Person-Centric Healthcare Using Ambient Assistive Technology: Personalized and Patient-Centric Healthcare Services in AAT* (pp. 227-245). Cham: Springer Nature Switzerland.

[24] Nagaraj, P., Ganesh, M., Yadav, G. B. P., Suneeth, T. S., & Kumar, K. M. (2023, March). Classification of Weeds Detection Control Management Using Artificial and Deep Convolutional Neural Networks. In *2023 9th International Conference on Advanced Computing and Communication Systems (ICACCS)* (Vol. 1, pp. 1294-1299). IEEE..