A Study on CNN Based Transfer Learning for Recognition of Flower Species

Ferhat Bozkurt[1*]

[1*] Atatürk University, Faculty of Engineering, Department of Computer Engineering, Erzurum, Turkey (ORCID: 0000-0003-0088-5825), fbozkurt@atauni.edu.tr

Abstract

The flower that is one of the plant organs, is essential element of the ecological order. Flowers have been used in many areas that are beneficial to humans. There exist about four hundred thousand varieties of flowers known today. It is a difficult task to distinguish flowers from each other due to their similarity in shape and color. Flower classification is a challenging problem due to the high variety of shapes, color distribution, lighting conditions and deformation of exposure. It becomes more difficult to distinguish flowers that are similar in color and shape to each other with the human eye for some images. It takes remarkable training for humans to correctly distinguish between particular species, and often very specific morphological features are the only thing that distinguishes closely related species. CNN models have been recently used by researchers in many classification problems to eliminate the need for manual features. In this study, CNN-based transfer learning methods are studied for recognition of flower species. Popular pretrained learning techniques which are VGG16, VGG19, SqueezeNet, DenseNet-121, DenseNet-201, and InceptionResNetV2 are conducted for classification of flower species. Their classification performances are compared on same flower dataset in experimental results. It was observed that the InceptionResNetV2 model gives superior results than other models in experiments. The highest accuracy (92.25%) is obtained with the InceptionResNetV2 model for flower dataset.

Keywords: Deep learning, Transfer learning, Convolutional neural network, Feature selection, Flower species.

Çiçek Türlerinin Tanınması için ESA Tabanlı Transfer Öğrenme Üzerine Bir Çalışma

Öz

Bitki organlarından biri olan çiçek, ekolojik düzenin önemli bir elementidir. Çiçekler insanlara faydalı olan birçok alanda kullanılmıştır. Günümüzde bilinen yaklaşık dört yüz bin çiçek çeşidi vardır. Çiçekleri şekil ve renk benzerliği nedeniyle birbirinden ayırt etmek zor bir iştir. Çiçek sınıflandırması, çok çeşitli şekiller, renk dağılımı, aydınlatma koşulları ve maruz kalma deformasyonu nedeniyle zorlu bir problemdir. Renk ve şekil olarak birbirine benzeyen çiçekleri insan gözüyle ayırt etmek bazı görüntülerde daha da zorlaşmaktadır. İnsanların belirli türleri doğru bir şekilde ayırt etmesi dikkate değer bir eğitim gerektirir ve genellikle çok spesifik morfolojik özellikler, yakından ilişkili türleri ayırt eden tek şeydir. ESA modelleri son zamanlarda araştırmacılar tarafından manuel özniteliklere olan ihtiyacı ortadan kaldırmak için birçok sınıflandırma probleminde kullanılmaktadır. Bu çalışmada, çiçek türlerinin tanınması için ESA tabanlı transfer öğrenme yöntemleri incelenmiştir. Çiçek türlerinin sınıflandırılması için, önceden eğitilmiş popüler öğrenme tekniklerden VGG16, VGG19, SqueezeNet, DenseNet-121, DenseNet-201 ve InceptionResNetV2 uygulanmaktadır. Deneysel sonuçlarda aynı çiçek veri kümesi üzerinde sınıflandırma performansları karşılaştırılmıştır. Bu çalışmada

---

* Corresponding Author: fbozkurt@atauni.edu.tr

InceptionResNetV2 modelinin diğer modellere göre daha üstün sonuçlar verdiği gözlemlenmiştir. En yüksek doğruluk (%92.25), çiçek veri seti için InceptionResNetV2 modeliyle elde edilmiştir.

Anahtar Kelimeler: Derin öğrenme, Transfer öğrenme, Evrişimli sinir ağları, Öznitelik seçme, Çiçek türleri.

1. Introduction

Flowers, one of the plant organs, are an indispensable element of the ecological order. Flowers have been used in many areas that are beneficial to humans (Roddy et al., 2019). Flowers are in constant demand for the pharmaceutical industry, the food and cosmetics industry, landscaping areas, and special occasions (Madoui et al., 2018). It is a difficult task to distinguish them due to the similarity of shape and color of flowers. Flower classification is a difficult problem due to the high diversity of shapes, color distribution, lighting conditions, and exposure deformation. Flowers are non-rigid bodies and can be deformed in different ways from image to image. More generally, taxonomy is an interesting image classification problem. It takes significant training for humans to correctly distinguish between certain species, and often very specific morphological features are the only thing that distinguishes closely related species (Bayram & Nabiyev, 2021). Moreover, these particular characteristics can be completely different from one particular plant species to another (for example, stem length - petal shape). Today, there are about four hundred thousand flower varieties (Christenhusz et al., 2016). It becomes more difficult to distinguish flowers that are similar in color and shape to each other with the human eye for some images (Seeland et al., 2019). A good understanding of flowers is essential to help identify new or rare species when encountered. This classification will help the development of the pharmaceutical industry. These scientific studies can be used by informatics as well as botanists, campers and doctors (Sangale et al., 2020). As classification of flower species is an important task, it is commonly studied and many different approaches have been developed by researchers. Previously methods such as histogram of oriented gradients, deformable part models, and scale-invariant feature transformation, feature extraction, non-linear classifiers have been used. Then the study focused on segmentation and classification by using handcrafted feature engineering (Wäldchen & Mäder, 2018). Today, however, the best performances have been achieved with CNNs. CNN accomplishes the need for robustness and eliminated the need for manual features. CNNs are similar to neural networks but do not require feature engineering (Narvekar & Rao, 2020).

The CNN model has been recently used by researchers in many classification problems such as disease diagnosis, face recognition and object recognition (Voulodimos et al., 2018). Up to present many techniques and methods have been proposed in the process of classifying flower images in literature. Guo et al. (2019) used Genetic Algorithm (GA) with LeNet model to classify the flowers. They utilized Tabu search algorithm and GA together to obtain effective features in short time. Their optimization methodology on hyper-parameter supplied efficient search for hyper-parameters of learning algorithms. They achieved 78.46% accuracy in flower classification with this model. Luus et al. (2019) proposed to use the SSL method and t-SNE dimensionality reduction in flower classification. They achieved 77.32% accuracy with the VGG-16 model. They merged Inception and ResNet architectures to achieve the best success. They used Semi-Supervised Learning (SSL) method to label their datasets. They achieved 79.33% accuracy by combining CNN, t-SNE dimensionality reduction, and SSL. Chen et al. (2019) used machine learning (Random Forest, SVM) and deep learning models (CNN, DenseNet, VGG16, and ResNet) to classify flower species. They obtained best success rate 90% with the VGGNet model. Toğaçar et al. (2019) used CNN for feature extraction. They used VGG-16, AlexNet, and VGG-19 in their study. They proposed to use CNN architectures and feature selection methods (Ridge regression, Maximum information coefficient, and Recursive feature elimination) together. They achieved 91.10% accuracy by using feature selection methods in CNN models. Mitrović & Milošević (2019) used the predefined models as AlexNet and LeNet. They achieved highest accuracy (73.41%) with LeNet CNN with a sigmoid function. FatihahSahidan et al. (2019) studied flower recognition using CNN. They investigated different stacks of layers to examine the effects of the number of layers on the accuracy performance. They achieved the highest accuracy (74%) with four stacks of layers. Arinda et al. (2018) proposed to use SVM with HSV and HOG. They performed color segmentation by the HSV method. They used the HOG method to obtain regional histograms of each image. Then they classified the grayscale images with the SVM method. The classification accuracy rate in their studies was 87%. Gadkari et al. (2019) studied transfer learning methods (VGG-16, VGG-19, and InceptionV3) and examined the various features of the flowers to recognize them on flower datasets. They obtained the best validation accuracy of 91.73% with the VGGNet model. Gurnani et al. (2017) propsed a deep learning (DL) network for classification of different flowers. They used 8189 flower data in 102 flower categories from the visual geometry group from Oxford University. The authors' methods are basically separeted into two parts; segmentation and classification. They compared two different CNN architectures which are AlexNet and GoogleNet for classification. MostafaMehdipour Ghazi et al. (2017) used pre-trained (transfer learning) CNN models for classification of plant species in their study. They applied image augmentation technique to the dataset for obtaining a new image set for robust classification. They have achieved 80% accuracy rate with their best-combined system. Lv et al. (2021) proposed VGG-16 deep neural network model with stochastic gradient descent algorithm and dropout to optimize the model for flower classification. They achieved 91.9% classification accuracy for experiments on the Oxford flower-102 data set. Yıldıran et al. (2014) proposed to use Dense SIFT features, Bag of Visual Words, and SVM classification. In their study, the plant in the image is described by Dense SIFT and visual word bag approach and classified by SVM. Classification accuracy is obtained as %42.68 for ImageClef'2013 flower data set. Turkoglu and Hanbay (2019), proposed to use a hybrid model with CNN and Color-Local Binary Pattern method for the recognition of plant species. LBP method is processed to each channel of color image. Then, they combined the deep and LBP features to classify features with SVM. Seeland et al. (2017) used classical machine learning methods for classifying flowers. They used a combination of local shape and color descriptors methods and feature selection detectors. They achieved a success rate of 94% with the SVM classifier.

In this study, CNN-based transfer learning methods are studied for recognition of flower species. Popular pretrained learning techniques which are VGG16, VGG19, SqueezeNet, DenseNet-121, DenseNet-201, and InceptionResNetV2 are conducted for classification of flower species. Their classification performances are compared on same flower dataset in experimental results. The rest of the work is organized as follows. In section II, material and method are presented. In this section, the dataset used in the experimental study and method are explained in detail. Experimental results are given in section III, and the discussion and conclusions part of the study is given in section IV.
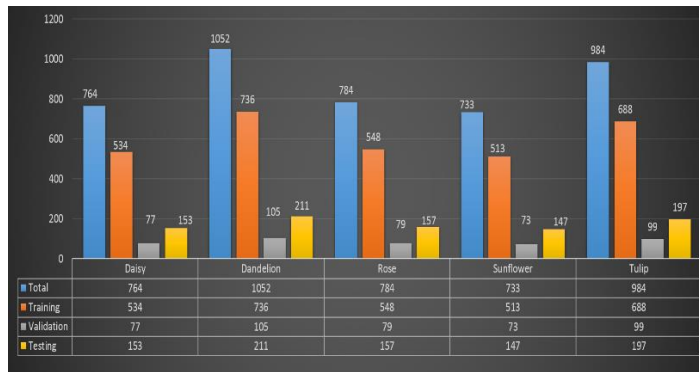

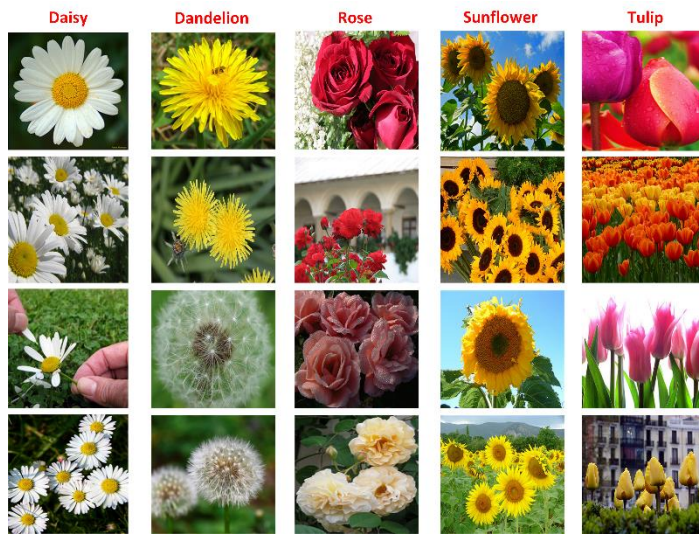Figure 1. Numerical distribution of the images in the dataset by classes

Figure 2. Example of random images from the flowers recognition dataset

2. Material and Method

2.1. Flower Dataset

In this study, an open-access flowers recognition dataset is used. Mamaev (2018) collected this dataset from Google, Flickr, and Yandex images and published on Kaggle website. This dataset contains five different types of flower species with 4317 images. The species of the flowers are dandelion, daisy, sunflower, rose, and tulip. The flower dataset contains images of 764 daises, 1052 dandelions, 784 roses, 733 sunflowers, and 984 tulip. The images in the dataset are not high resolution. Images are 24 bits depth in jpg format and resolutions of photos are not high resolution (about 320x240 pixels). The number of images of flower classes is shown in Figure 1. Image selection is done

randomly. The dataset is adjusted as 70%, 10%, and 20% testing rates for training, validation, and test sets respectively in all experimental steps of this study. 3019 images are used from each class for training, 433 images are used from each class for validation, and 865 images are used from each class for testing. The example of random images from the used dataset are shown in Figure 2.

2.2. Method

Deep learning can be defined as the work of using the necessary layers to train and learn an artificial neural network model, which consists of many hidden layers, in complex problems that contain a large number of data and variables but do not have an existing formula or equation. The main advantage of this process is that the features are extracted from the raw data and the classifier model automatically trains itself according to the extracted features. Classification with machine learning and image processing methods require various preprocessing and feature extraction techniques to process raw images. This is a very challenging and experience-requiring process. For this reason, researchers have worked on learning features automatically instead of learning them manually. As a result of these studies, deep learning has become popular and has proven its capabilities in many areas with the development of parallel processing GPUs that allow complex matrix operations and the increasing amount of data (Zhang et al., 2018). Thanks to deep learning, machines can recognize objects from images, classify data such as audio and text with much lower error rates than other methods.

As shown in Figure 3 in the basic flow of CNN network, features were extracted from flower images with sequential convolution and pooling layers in CNN, and classification and flower recognition could be performed in this study.

2.2.1. Convolutional Neural Networks

The convolutional neural network (CNN) classifies the incoming data through the neural network according to the weight values it obtains from the differences in the data received with the filters in the layers. CNN is a deep learning architecture that is recommended for the solution of image processing problems and can adapt very well to images. The network gets its name from performing complex operations by doing convolution operations. An image is converted into vectors in classical methods, pixels are examined only according to certain neighborhood values or pixels are examined independently of each other. In this case, many information about the objects in the image is lost. The CNN structure eliminates this problem with the filters it uses. CNN does not use predefined features, but instead learns locally connected neurons that represent data-specific features. In CNN, filters are repeatedly applied to the entire image. CNN can automatically learn features from image (Li et al., 2016).

CNN, which is considered to be the most basic architecture of deep learning, consists of one or more convolutional layers and pooling layers, then a fully connected layer and finally a classification layer as shown in Figure 3. With these sequential layers, various features that affect the result are obtained from the first input layer to the last classification layer, and the classification process is performed with the last layer. Thanks to these features, it means that there is an error as much as the difference between the determined class and the class that should be assigned in the data set. With the back propagation algorithm,
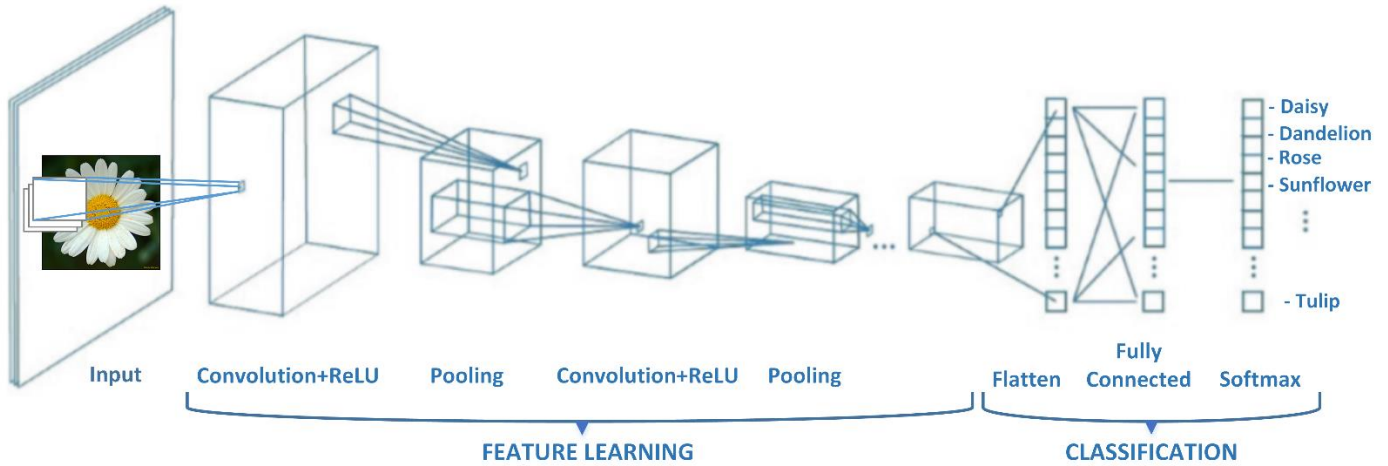
Figure 3. Basic flow of CNN architecture for recognition of flower species

this margin of error is minimized by updating the values of the weights in each stage. In this way, the feature extraction process is improved and the accuracy in the test data is increased.

CNN consists of layers such as input layer, convolutional layer (Conv), pooling layer, fully connected layer (Fcl), and dropout layer. The pooling layer is an important building block that reduces the size of feature maps by summarizing subregions by averaging or maximizing. The fully connected layer is used by the convolution layers of the features obtained from an image to increase the classification performance. The dropout layer excludes some of the connections within the network from training to prevent network overfitting and improve training performance (Krizhevsky et al., 2012).

The convolution layer is the most important layer of the convolutional neural network, in this layer, feature maps are created by applying multidimensional filters to the input data. Generally, models with multiple convolution layers enable different features to emerge, while at the same time shrinking the received data depending on the filter size. In this layer, small filters such as 2x2, 3x3 and 5x5 are applied to the entire image. Hence, a new image is obtained by removing more characteristic features in the image. The weights of the filter matrix used for the convolution operation are determined during the learning phase of the convolutional neural network. The filter matrix is shifted by the specified step and the convolution operation is performed. As in equation (1), the filter coefficients (f) are calculated by multiplying the windows of equal size (w) in the image and summing them. As a result, a new image is created based on distinctive high-level features. The result obtained here is given as input for the next layer. Feature learning is done with the filters in this layer. Different parameters such as the number, size and type of filters used also affect the performance to be achieved in the classification (Kim, 2017).

$$w(x, y) * f(x, y) = \sum_{i=-a}^{a} \sum_{j=-b}^{b} w(i, j) f(x+i, y+j) \quad (1)$$

Pooling layer is usually between one or more convolution layers, it applies a filter like a convolution layer. The pooling layer could be used multiple times in models created like the convolution layer. The pooling layer reduces parameters within the network, it is used to reduce the computational load. The highest of the pixel values in the filter window for maximum

pooling is kept as the maximum, the average of all pixel values in the filter window for average pooling is kept as a single value in the output pixel. Thus, the aspect ratio of the image is reduced at the end of the pooling process. Pixel values are lost in this process, however these losses create less computational load for the next layers.

The above-mentioned layers of convolution and pooling can be applied multiple times in consecutive. The matrix obtained after these processes must be flattened in flattening layer to be used in the fully connected layer. The task of this layer is to flat and transfer the incoming data. It connects a structure consisting of many convolution and pooling layers to the classification layer, while it transforms the incoming multidimensional matrix into a one-dimensional matrix. Thus, this translated matrix is suitable for the number of classes of the network.

Fully connected layers use feature information to obtain a final class label. The nodes of the fully connected layers are connected to each output of the previous layer, and their outputs are connected to each node of the next layer. In the last layer, there exist n nodes (n is equal to the number of classes). Nodes in the last layer of fully connected layers usually have a special activation function (Erdem & Aydin, 2021). This function will generate a probabilistic result for each label. This activation function is usually a softmax function or, in the case of binary classification, a sigmoid function.

$$soft \max(z)_j = \frac{e^{z_j}}{\sum_{k=1}^{K} e^{z_k}}; j = 1,......., K \quad (2)$$

Classification process is done in the classification layer of the CNN model. The output values of this layer depend on the number of objects to be recognized and are equal to the number of classes. Any complex and continuous relationship between network variables are learnt by activation functions. For instance ReLU activation function is used to avoid linearity in the network. Because multiclass classification is required in the fully connected layer (Fcl) in CNN architecture for flower classification, a softmax function is used as given in equation (2). This classifier generates probabilistic values between 0-1 for each class. Consequently, the highest probability value gives the class that the model predicts (Kim, 2017).

When CNN architectural structure is examined, it is seen that it contains many parameters. However, this situation may

cause overfitting the network with the training data and cause low accuracy in validation and test data or excessive learning. Features can be transferred between layers. The process of transferring features means that a matrix processes the whole image by shifting the feature matrices in the convolution layer on the image. Thus, each matrix affects the image. With this feature, the network parameters become much less than they should be. The low training parameter both speeds up the training process and eliminates the overfitting problem. As the network gets deeper, in other words, the number of hidden layers increases, the size of the images and the learned features in the network are reduced. In this reduction process, the pixels with the highest representation ability in the image are used. The location of the object in the image is also unimportant, this problem has been overcome by shifting the filters (Krizhevsky et al., 2012).

3. Experimental Results

Experimental studies are preformed on a computer with processor of Intel (R) Core 8700U i7 CPU (3.20 GHz), graphics card of 4GB NVIDIA GeForce 1050 GTX Ti and 16GB primary memory. The software is done by using Python 3.10.0 version and its related libraries. In experimental studies, the data is reserved for 70% training, 10% validation and 20% testing. At this stage, the performance of different deep learning techniques are studied on flower dataset with different evaluation criteria. These pretrained learning techniques are VGG16, VGG19, SqueezeNet, DenseNet-121, DenseNet-201, and InceptionResNetV2.

The VGG16 Architecture was developed and introduced in 2014 by Karen Simonyan and Andrew Zisserman of the University of Oxford. 'VGG' is short for Visual Geometry Group, a group of researchers at Oxford University who developed this architecture, and '16' implies that this architecture has 16 layers. The VGG-16 architecture consists of convolutional, pooling, and fully connected layers. It consists of 21 main layers in total. The image input resolution is 224×224 pixels. Moreover, the filter size in the convolutional layer is 3×3 pixels. In this model, the last layers consist of fully connected layers used for feature extraction. The main layers of the VGG-19 architecture consist of 16 convolutional, five pooling, and three fully connected layers. This architecture has a total of 24 main layers. Since VGG-19 has a deep network, filters used in the convolutional layer are used to reduce the number of parameters. The size of the filter selected in this architecture is 3×3 pixels. The VGG-19 architecture contains approximately 138 million parameters. The SqueezeNet architecture was developed by Iandola et al. in 2016. The aim of this architecture is to create a neural network with fewer parameters and the architecture provides AlexNet level accuracy with 50 times less parameters. The advantage of the SqueezeNet architecture is that thanks to more efficient distributed layers, the computational load in the neural network is reduced and thus it works faster. DenseNet-121 (Densely connected CNN) forward connects each layer to the other layers. In DenseNet architecture, each layer uses the features of all previous layers as input and its own features in the layer are given as input to the next layers. The advantage of DenseNet architectures is that it provides feature propagation and reduces the number of parameters by allowing feature reuse. This architecture contains of 4 dense blocks, 3 transition layers, and 121 layers these are 117-conv, tree-transition, and one-classification. The other advanced DenseNet-

201 model which has a total of 121 layers is also studied on this dataset. The Inception-ResNet model is a hybrid model inspired by the Inception modules in the Inception architecture and the performance of the ResNet architecture. The network is 164 layers deep. In this state, InceptionResNetV2 model has 55,855,205 trainable parameters (Raj & Vajravelu, 2019).

Table 1. Number of Trainable Parameters of Different Models

Model	# of Trainable Parameters
VGG16	66,309
VGG19	132,613
SqueezeNet	119,301
DenseNet-121	591,109
DenseNet-201	1,049,861
InceptionResNetV2	55,855,205

Table 2. Performance Scores of the InceptionResNetV2 model

Accuracy: 92.25%	Precision	Recall	F1-Score
Daisy	0.9664	0.9412	0.9536
Dandelion	0.9182	0.9573	0.9374
Rose	0.9178	0.8535	0.8845
Sunflower	0.9710	0.9116	0.9404
Tulip	0.8679	0.9340	0.8998
Average:	0.9283	0.9195	0.9231

Table 3. Performance Comparisons of CNN-based Transfer Learning Models

	Precision	Recall	F1-Score	Acc (%)
VGG16	0.8681	0.8694	0.8678	86.94
VGG19	0.8569	0.8484	0.8516	85.09
SqueezeNet	0.7893	0.7938	0.7871	79.19
DenseNet-121	0.8318	0.8249	0.8238	82.89
DenseNet-201	0.8379	0.8029	0.7891	80.69
InceptionResNetV2	0.9283	0.9195	0.9231	92.25

At this stage, the number of trainable parameters was increased by increasing the number of deep layers as shown in Table 1. The input image resolutions are adjusted to 240x224x3 in experimental results for low computation load. All experiments are done with batch size (mini-batch) 128 and with 100 epochs at training. Performance evaluations of the pertained methods are conducted by four performance criteria as Precision, Recall, F1-score, and Accuracy (Acc). Performance criteria are computed by using actual and predicted labels from confusion matrices (Coban, 2021; Nahzat & Yağanoğlu, 2021). These performance metrics are computed with confusion matrices for each class (0=Daisy, 1=Dandelion, 2=Rose, 3=Sunflower, and 4=Tulip) as shown in Figure 4. For instance, performance scores of the InceptionResNetV2 model are given in Table 2. The calculated scores about precision, recall, F1-score for each class and average results are given in this table. According to this

experimental study, the accuracy value obtained with the InceptionResNetV2 model is 92.25%.

As shown in Table 3, CNN-based transfer learning models have been compared. Through the InceptionResNetV2 model, higher accuracy results were obtained in the experiments compared to other models. The highest accuracy (92.25%) is obtained with the InceptionResNetV2 model for flower dataset. Since number of trainable parameters is increased by increasing the number of deep layers, accuracy rate of flower classification increases. According to these experimental studies, the accuracy value obtained with the InceptionResNetV2 model is 92.25% (with 55,855,205 trainable parameters). In Figure 5, graphs of training/test loss and training/test accuracy for 100 iterations of the InceptionResNetV2 model are shown. When the number of iterations increases, a rising curve is shown in training and test accuracy. Thus learning occurs on training data. Error rate is low and stable as shown in loss curve of training/test loss graph. We can say that learning of the network occurs at a reasonable learning rate according to the training process. Thus the end of 100 iterations, the accuracy rate increases and loss value decreases and learning occurs for the training set as shown in Figure 5. Performance comparison of the current study with other studies using same flower dataset is given as shown in Table 4.
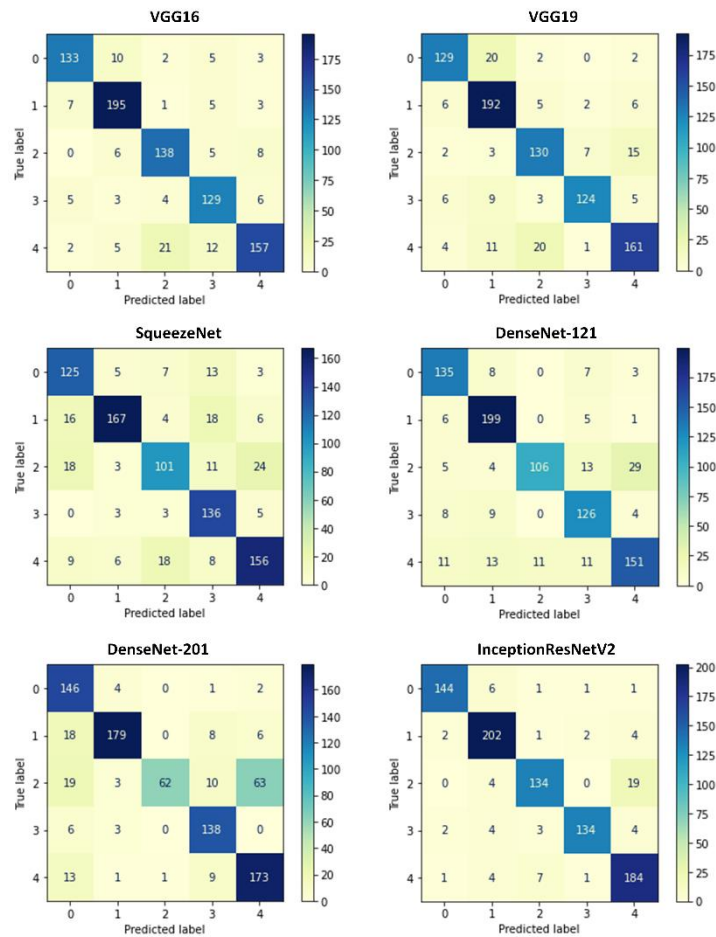
from each other due to their similarity in shape and color. Flower classification is a challenging problem due to the high variety of shapes, color distribution, lighting conditions and deformation of exposure. It becomes more difficult to distinguish flowers that are similar in color and shape to each other with the human eye for some images. It takes remarkable training for humans to correctly distinguish between particular species, and often very specific morphological features are the only thing that distinguishes closely related species.

Table 4. Performance Comparison of This Study with Other Studies Using the Same Flower Dataset

Study	Year	Method	Acc (%)
Guo et al.	2019	CNN & Genetic Algorithm	78.46
Luus et al.	2019	CNN& t-SNE Dim Red.&SSL	79.33
Chen et al.	2019	VGG16	90.00
Toğaçar et al.	2019	CNN+Feature Selection	91.10
Mitrović & Milošević	2019	LeNet	73.41
FatihahSahidan et al.	2019	CNN	74.00
Arinda et al.	2018	SVM with HSV and HOG	87.00
Gadkari et al.	2019	InceptionV3	91.73
This study	2021	InceptionResNetV2	**92.25**

Figure 4. Confusion matrices of studied CNN models

4. Discussion and Conclusions

Flowers have been used in many areas that are beneficial to humans. There exist about four hundred thousand varieties of flowers known today. It is a difficult task to distinguish flowers
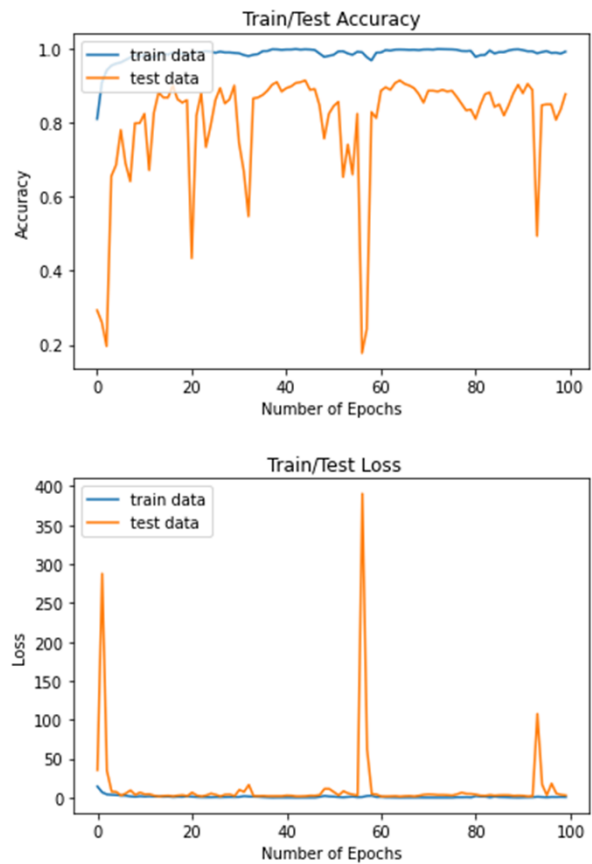

Figure 5. Training/test accuracy and training/test loss graphs of the InceptionResNetV2 model

As the network gets deeper with CNN models, the number of hidden layers increases. Thus the size of the images and the learned features in the network are reduced. In this reduction

process, the pixels with the highest representation ability in the image are used. The location of the object in the image is also unimportant, this problem has been overcome by shifting the filters. In this study, CNN-based transfer learning methods are studied for recognition of flower species. Popular pretrained learning techniques which are VGG16, VGG19, SqueezeNet, DenseNet-121, DenseNet-201, and InceptionResNetV2 are conducted for classification of flower species. Their classification performances are compared on same flower dataset in experimental results. Through the InceptionResNetV2 model, higher accuracy results are obtained in the experiments compared to other models. The Inception-ResNet model is a hybrid model inspired by the Inception modules in the Inception architecture and the performance of the ResNet architecture. The network is 164 layers deep and InceptionResNetV2 model has 55,855,205 trainable parameters. In this study, the number of trainable parameters was increased by increasing the number of deep layers. According to these experimental studies, the accuracy value obtained with the InceptionResNetV2 model is 92.25% (with 55,855,205 trainable parameters). Since number of trainable parameters is increased by increasing the number of deep layers, accuracy rate of flower classification increases. In further studies, other deep learning-based techniques, hybrid methods and feature engineering with deep, handcrafted, and fusion-based techniques will be studied and comparisons will be made on this dataset.

References

Arinda, Y. K., Rahman, M. A., & Alamsyah, D. (2018). Klasifikasi Jenis Bunga menggunakan SVM dengan Fitur HSV dan HOG. Ijccs, no. x, 1-12.

Bayram, E., & Nabiyev, V. (2021). Classification of Camouflage Images Using Local Binary Patterns (LBP). In 2021 29th Signal Processing and Communications Applications Conference (SIU) (pp. 1-4). IEEE.

Christenhusz, M. J., & Byng, J. W. (2016). The number of known plants species in the world and its annual increase. Phytotaxa, 261(3), 201-217.

Chen, B., Liu, J., Sun, J., Liu, J. (2019). Flowers Classification via Deep Learning Models. http://noiselab.ucsd.edu/ECE228_2019/Reports/Report40.pdf (accessed November 10, 2021).

Coban, O. (2021). IRText: An Item Response Theory-Based Approach for Text Categorization. Arabian Journal for Science and Engineering, 1-17.

Erdem, E., & Aydin, T. (2021). A CNN-based hybrid model to detect Coronavirus disease. Avrupa Bilim ve Teknoloji Dergisi, (27), 66-73.

FatihahSahidan, N., Juha, A. K., Mohammad, N., & Ibrahim, Z. (2019). Flower and leaf recognition for plant identification using convolutional neural network. Indonesian Journal of Electrical Engineering and Computer Science, 16(2), 737-743.

Gadkari, S., Mathias, J., & Pansare, A. (2019). Analysis of Pre-Trained Convolutional Neural Networks to Build a Flower Classification System. International Journal for Research in Applied Science & Engineering Technology (IJRASET), 2321-9653, Vol 7, Issue 11.

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

Guo, B., Hu, J., Wu, W., Peng, Q., & Wu, F. (2019). The Tabu_genetic algorithm: a novel method for hyper-parameter optimization of learning algorithms. Electronics, 8(5), 579.

Gurnani, A., Mavani, V., Gajjar, V. and Khandhediya, Y., (2017). Flower Categorization using Deep Convolutional Neural Networks, ArXiv, 4321-4324.

Kim, P. (2017). Convolutional neural network. In MATLAB deep learning (pp. 121-147). Apress, Berkeley, CA.

Krizhevsky, A., Sutskever, I., & Hinton, G. E. (2012). Imagenet classification with deep convolutional neural networks. Advances in neural information processing systems, 25, 1097-1105.

Li, Y., Hao, Z. B., & Lei, H. (2016). Survey of convolutional neural network. Journal of Computer Applications, 36(9), 2508-2515.

Luus, F., Khan, N., & Akhalwaya, I. (2019). Active learning with tensorboard projector. arXiv preprint arXiv:1901.00675.

Lv, R., Li, Z., Zuo, J., & Liu, J. (2021). Flower Classification and Recognition Based on Significance Test and Transfer Learning. In 2021 IEEE International Conference on Consumer Electronics and Computer Engineering (ICCECE) (pp. 649-652). IEEE.

Madoui, S., Charef, N., Arrar, L., Baghianni, A., & Khennouf, S. (2018). In vitro Antioxidant Activities of Various Extracts from Flowers-Leaves Mixture of Algerian Cytisus triflorus. Annual Research & Review in Biology, 1-13.

Mamaev, A., Flowers Recognition | Kaggle, (2018). https://www.kaggle.com/alxmamaev/flowers-recognition (accessed November 10, 2021).

Mitrović, K., & Milošević, D. (2019). Flower classification with convolutional neural networks. In 2019 23rd International Conference on System Theory, Control and Computing (ICSTCC) (pp. 845-850). IEEE.

Nahzat, S., & Yağanoğlu, M. (2021). Diabetes Prediction Using Machine Learning Classification Algorithms. Avrupa Bilim ve Teknoloji Dergisi, (24), 53-59.

Narvekar, C., & Rao, M. (2020). Flower classification using CNN and transfer learning in CNN-Agriculture Perspective. In 2020 3rd International Conference on Intelligent Sustainable Systems (ICISS) (pp. 660-664). IEEE.

Raj, A. P. S. S., & Vajravelu, S. K. (2019). DDLA: dual deep learning architecture for classification of plant species. IET Image Processing, 13(12), 2176-2182.

Roddy, A. B., Jiang, G. F., Cao, K., Simonin, K. A., & Brodersen, C. R. (2019). Hydraulic traits are more diverse in flowers than in leaves. New Phytologist, 223(1), 193-203.

Sangale, R., Jangada, R., De, A., Sanga, N., & Deokar, S. (2020). Flower Recognition Using Deep Learning. International Journal of Research Publication and Reviews Vol (1) Issue (8), 20-23.

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.

Seeland, M., Rzanny, M., Boho, D., Wäldchen, J., & Mäder, P. (2019). Image-based classification of plant genus and family for trained and untrained plant species. BMC bioinformatics, 20(1), 1-13.

Toğaçar, M., Ergen, B., & Özyurt, F. (2020). Evrişimsel Sinir Ağı Modellerinde Özellik Seçim Yöntemlerini Kullanarak Çiçek Görüntülerinin Sınıflandırılması. Fırat Üniversitesi Mühendislik Bilimleri Dergisi, 32(1), 47-56.

Turkoglu, M., & Hanbay, D. (2019). Plant Recognition System based on Deep Features and Color-LBP method. In 2019 27th Signal Processing and Communications Applications Conference (SIU) (pp. 1-4). IEEE.

Voulodimos, A., Doulamis, N., Doulamis, A., & Protopapadakis, E. (2018). Deep learning for computer vision: A brief review. Computational intelligence and neuroscience, 2018.

Wäldchen, J., & Mäder, P. (2018). Plant species identification using computer vision techniques: A systematic literature review. Archives of Computational Methods in Engineering, 25(2), 507-543.

Yıldıran, S. T., Yanıkoğlu, B., & Abdullah, E. (2014). Plant identification using local invariants. In 2014 22nd Signal Processing and Communications Applications Conference (SIU) (pp. 2094-2097). IEEE.

Zhang, Q., Yang, L. T., Chen, Z., & Li, P. (2018). A survey on deep learning for big data. Information Fusion, 42, 146-157.