International Journal of Modern Agriculture, Volume 9, No.3, 2020
ISSN: 2305-7246

Flower Identification & Classification Based on Efficient Deep Learning Technique

Rupali Jena[1],Sujata Chakravarty[2]
[1,2]Centurion University of Technology & Management, Bhubaneswar, Odisha
Email:[1]rupalijena965@gmail.com , [2]sujata.chakravarty@cutm.ac.in

Abstract

Every day we see a huge number of flower species in our house, parks, roadsides, in farms, on our rooftop but we have no knowledge of that flower species or their origin. Even we have no idea about its name. There are several guidebooks for flowers knowledge but it becomes quite difficult to find the name when have the picture. Even the Internet sometimes is not useful. But it is quite difficult for human brain to memorize all the species they see. Even some flower is similar to look at. Image-based automatic flower species classification is an important problem for the biologists who construct digital flower catalogues. Image-based automatic flower species classification is an important problem for the biologists who construct digital flower catalogues. A dozen of work about flower species recognition has been proposed so far based on traditional image processing routines. Nowadays, researchers apply the deep learning on various image-based object recognition tasks. This Paper is based on the automatic flower recognition techniques that can be used by biologists, taxonomist and naïve peoples also. For constructing efficient feature extraction from CNN model is referred. Finally, a feature selection algorithm done automatically by neural network model. A support vector machine (SVM) is used to classify having Radial Bases Function (RBF) kernel is employed on Gray-scale image as well as in colour images, hence the comparison has been done. Flower dataset having 5 species which have a manageable number of images.

Keywords: Identification; classification; CNN, Deep learning; SVM, Random-Forest; RBF

Introduction

While in our daily life, people usually use mobile phones, cameras to shoot flowers, sometimes they are confused because of the flower species. Therefore, the design of an auto recognition system for flowers is necessary in Botany and also will bring so much fun to people's lives. Nowadays many types of research have been conducted aboutto solve the feature selection and classification problems [1,2]. The flower species classification is also one of these problems. An automatic flower categories classification system is still a challenging task due to plenty of similarities among classes [1]. Moreover, flowers can deform in many ways, and consequently, there are also large intra-class variations [2,3]. More variations occur in viewpoint, occlusions, the scale of flower images [3]. Flowers images might have different positions with complex background under varying lighting and climatic conditions [1]. All these problems make the flower recognition more challenging. Some physical features have been used to distinguish among the flowers. The paper [2] combined different features to enhance flower classification performance on a dataset. Different aspects, which are the local shape/texture, the shape of the boundary, the overall

spatial distribution of petals, and the colour. A support vector machine (SVM) classifier was used for classification both Gray-scale images as well coloured images got accuracy for gray-scale is 24% but in color images 74% accuracy whereas in random- forest got only 31% but in CNN got 89% accuracy.

Related Work

Nils back et al. [3] noted that color and type unit of measurement are the key choices in flower classification. The flower is divided using a threshold-based technique, and texture choices, specifically the colour texture moments (CTMs), gray-level co-occurrence matrix (GLCM), and scientist responses, unit of measurement are extracted. This choice unit of measurement used for work and classification is applies using a probabilistic neural network. Y. Yoshioka et al. In this paper, they have done MKL-SVM that is multi kernel labelling with SVM classifier.

Dataset:

Dataset is collected from Kaggle.com which is a categorical dataset. This dataset contains 4242 images of flowers. The data collection is based on the data flick, google images, Yandex images. Using this dataset to recognize plants from the photo. The pictures are divided into five classes: Daisy, tulip, rose, sunflower & dandelion. For each class there are about 800 photos. Photos are not high resolution, about 320x240 pixels. Photos are not reduced to a single size; they have different proportions.

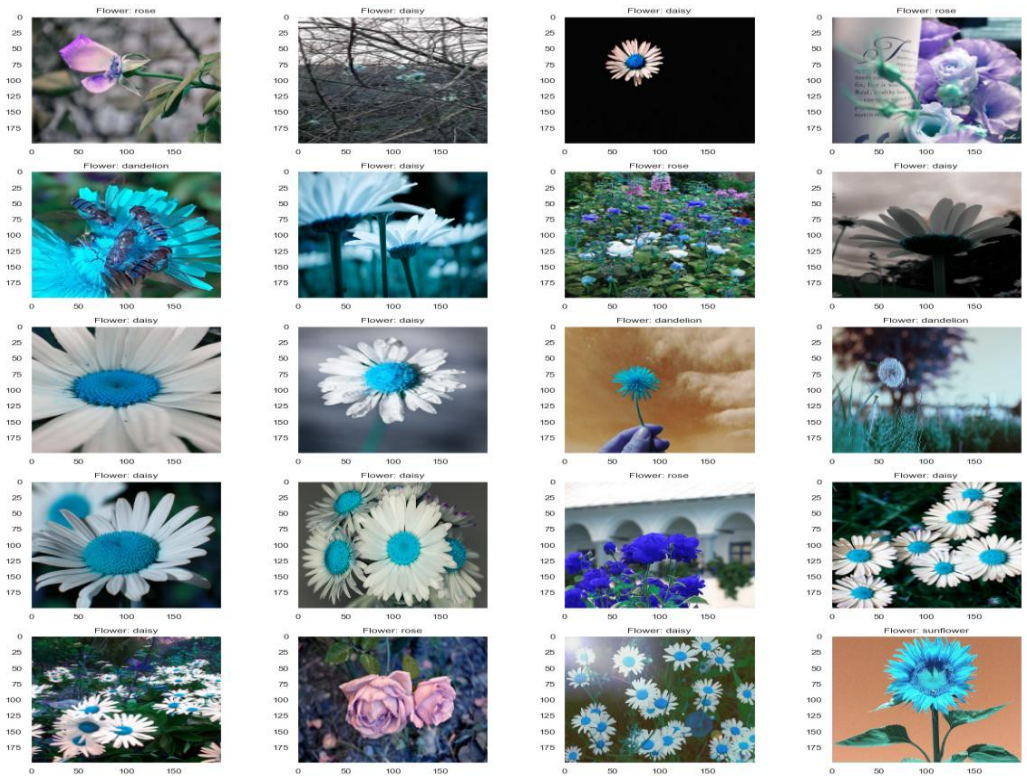


Fig 1 Sample of flower images

Proposed Flower Identification Methodology

The proposed method for flower classification and identification is represented in fig 1. The process is started with image pre-processing step with image resizing, noise filtering and RGB and Gray scale conversion. Afterwards, image segmentation is completed with Sobel technique. After performing segmentation, the features are taken out from segmented image including basic features of the flowers. The outcome is of gray-scale images as done by manual feature extraction to get pre-trained the SVM model as well as Random Forest Model.

The SVM model is mostly popular model for classification. So here multi-class labelling has been introduced with the help of SVM, Random-forest and CNN. Here the model gets trained and hence, classify the flowers. It is divided into 3 group's 80%training, 10%testing and 10%validation.

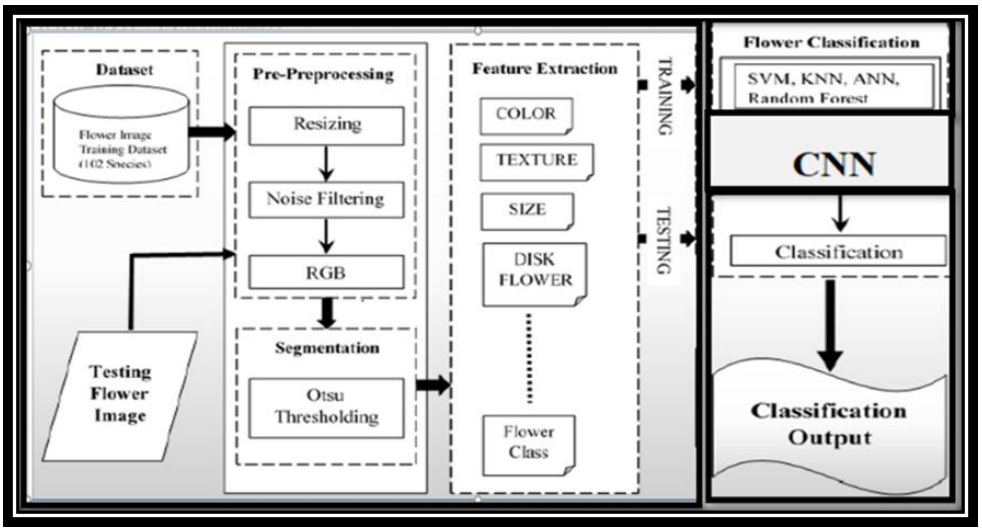


Fig2 Architecture of methodology

Flower Pre-processing

The first step towards flower image processing is the pre-processing of an image. As image pre-processing is the fundamental step, it is applied to increase the quality of images and removing the inappropriate noises presented in images. The aim of pre-processing techniques is image enhancement and image restoration [4,5]. Image Enhancement is a significant process that aimed to recover the visual look of an image. It is provided for the better transform representation for the next phases of image detection [4,6]. Image enhancement is characterized in three types: Image scaling, Colour space transformation, Contrast Enhancement [5,2]. For image restoration, first noise level resides in the image is measured. Normally, an image may contain various types of noises including Gaussian, Salt and Pepper, Poison, Speckle etc. [6,2]. These noises are removed by applying appropriate filter that gives an optimum result for specific noise removal. So, we are using traditional feature extraction process for SVM & Random Forest models. Because the data amount of the images, including width, height and channels is huge. And in the recognition process, there are a large

number of iterations and epochs to train a neural network, so we choose to use Deep Learning in this situation. Training a traditional CNN needs a huge training dataset and testing dataset, and before classifying, it needs feature extracting which usually done automatically.

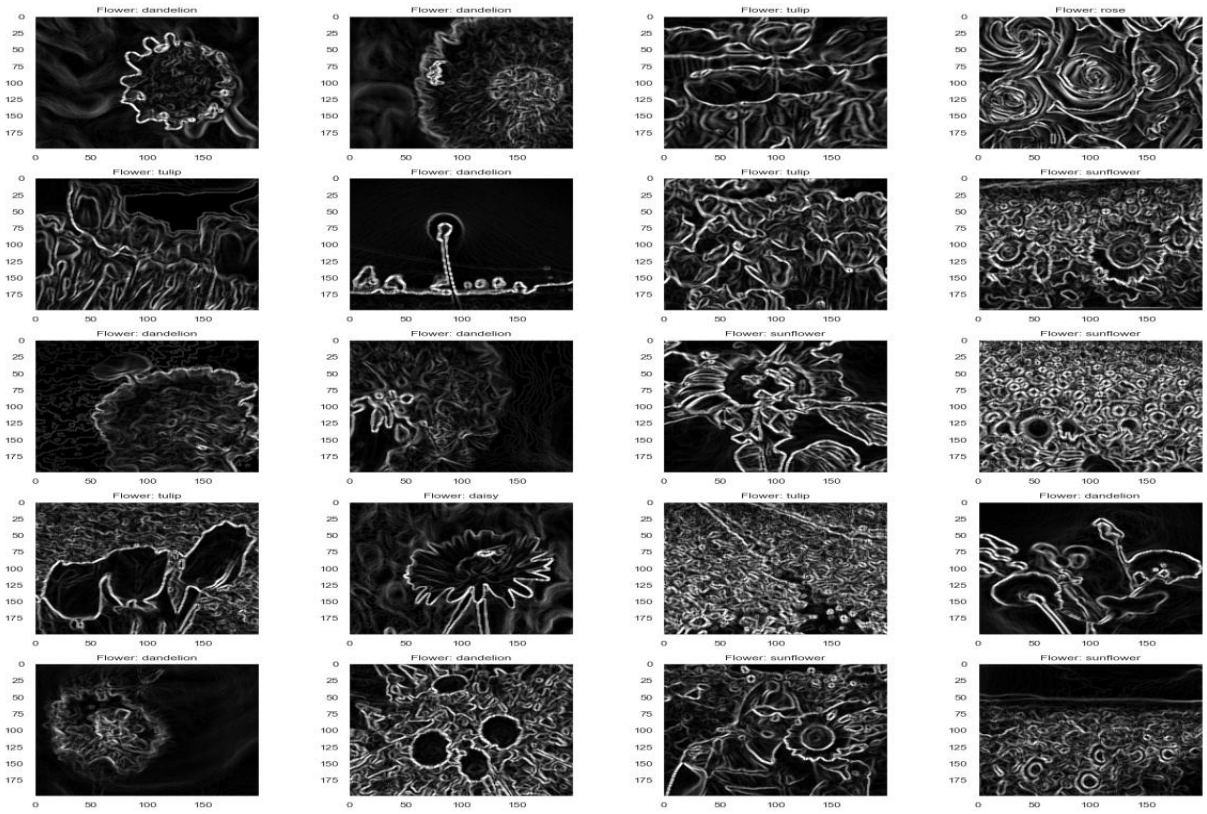


Fig 3 Samples of Gray-scale images after filtering

Flower Segmentation

After applying necessary image pre-processing techniques, segmentation is applied on image for further processing. Removing the undesirable background in the image is the second step in flower identification. Images that contain flowers are too contain parts of plant, leaves or grass in the background. In order to extract the correct features, it is required to separate the flower image from its background. To remove the background of images and improve the quality of flower image foreground, segmentation techniques are used.

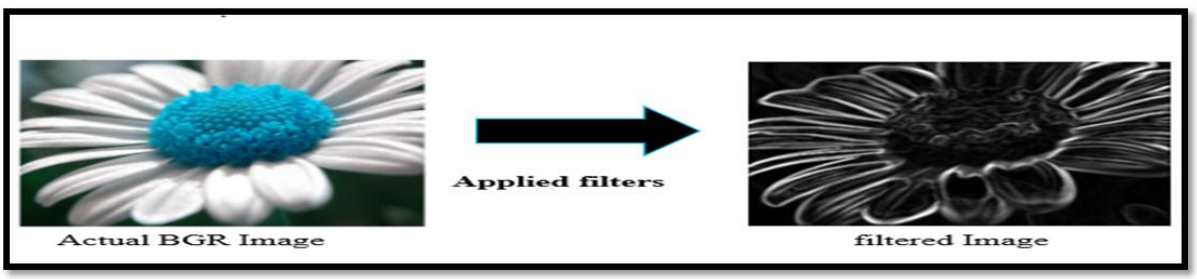


Fig 4 Final Images of Segmentation

Normalizing the image values.

Applying image segmentation, features are extracted from the segmented image. In the field of computer vision and image processing, feature describes an information that is used to identify or detect some object related to certain application. Feature extraction is the process applied to an image to convert its visual information into vector space. The normalization is done for efficient computation the values ranges from 0 to 1.

Build model:

For this paper we have introduced three model to testing our performance, the algorithms are; Support Vector Machine (SVM), Random-Forest & convolutional neural networks (CNN).

Therefore, the SVM get trained from the patterns of the filtered images. In the process of model building the RBF kernel is used as it is used for non-linear data and also mostly used in image classification related problem.

Support Vector Machine (SVM) is a supervised machine learning algorithm which can be used for both classification or regression challenges.

However, it is mostly used in classification problems. In the SVM algorithm, in the plot each data item as a point in n-dimensional space (where n is number of features you have) with the value of each feature being the value of a particular coordinate. Then, performing classification by finding the hyper-plane that differentiates the two classes very well.

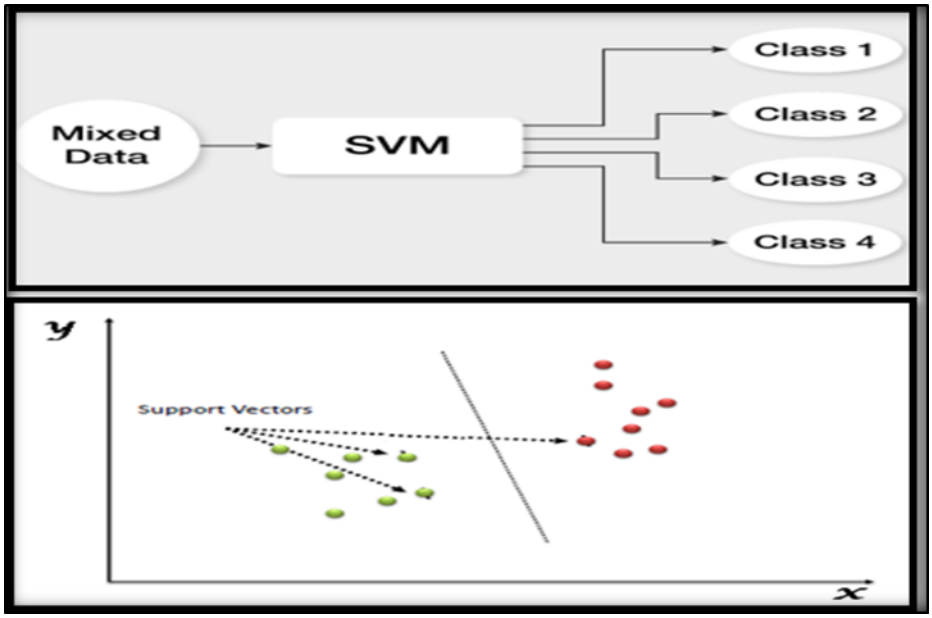


Fig 5 Graphical representation of SVM approach

Random-Forest Classifier

The random forest combines hundreds or thousands of decision trees, trains each one on a slightly different set of the observations, splitting nodes in each tree considering a limited

number of the features. The final predictions of the random forest are made by averaging the predictions of each individual tree.

Support Vector Machine (SVM):

Applying various computer vision techniques to the image dataset contains various features that are linear and non-linear. As SVM is mostly preferred algorithm for classification as it creates a best fit hyperplane to classify the classes. The random forest is a model made up of many decision trees. Rather than just simply averaging the prediction of trees (which we could call a "forest"), this model uses two key concepts that gives it the name RANDOM:

1. Random sampling of training data points when building trees
2. Random subsets of features considered when splitting nodes

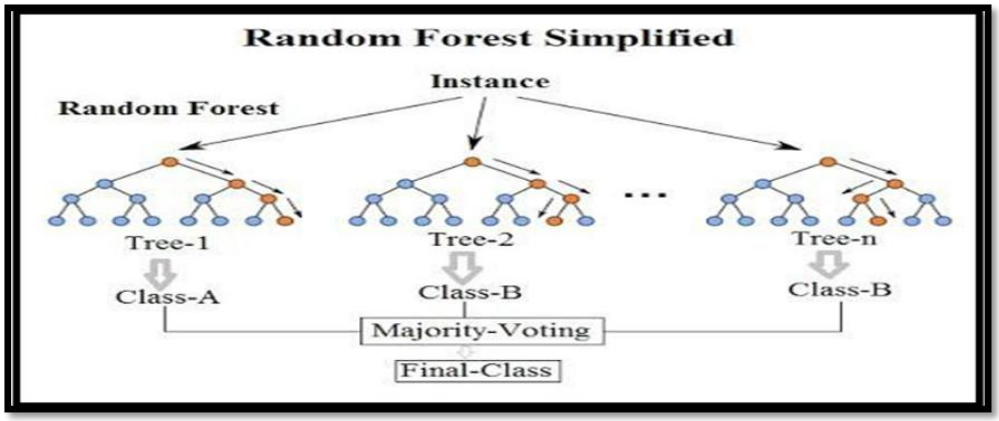


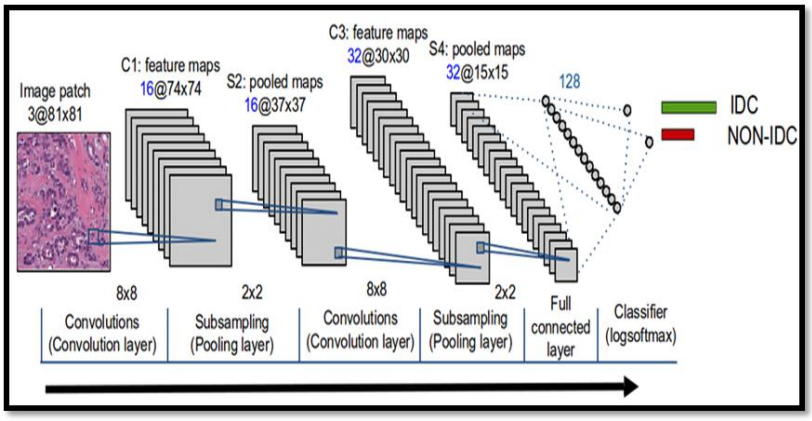
Fig 6 Working Model of Random-Forest

Convolutional neural network

A Convolutional Neural Network is a special type of an Artificial Intelligence implementation which uses a special mathematical matrix manipulation called the convolution operation to process data from the images. A convolution does this by multiplying two matrices and yielding a third, smaller matrix. The Network takes an input image, and uses a filter (or kernel) to create a feature map describing the image. In the convolution operation, we take a filter (usually 2x2 or 3x3 matrix) and slide it over the image matrix. The corresponding numbers in both matrices are multiplied and and added to yield a single number describing that input space. This process is repeated all over the image. This can be seen in the following animation. This is a 2-D representation of calculations happening in 3 dimensions. This is what is actually happening. We use different filters to pass over our inputs, and take all the feature maps, put them together as the final output of the convolutional layer. We then pass the output of this layer through a non-linear activation function. The most commonly used one is ReLU. The next step of our process involves further reducing the dimensionality of the data which will lower the computation power required for training this model. This is achieved by using a Pooling Layer. The most

commonly used one is max pooling which takes the maximum value in the window created by a filter. This significantly reduces the training time and preserves significant information.

Fig 7 Working Model of CNN



Model Deployment

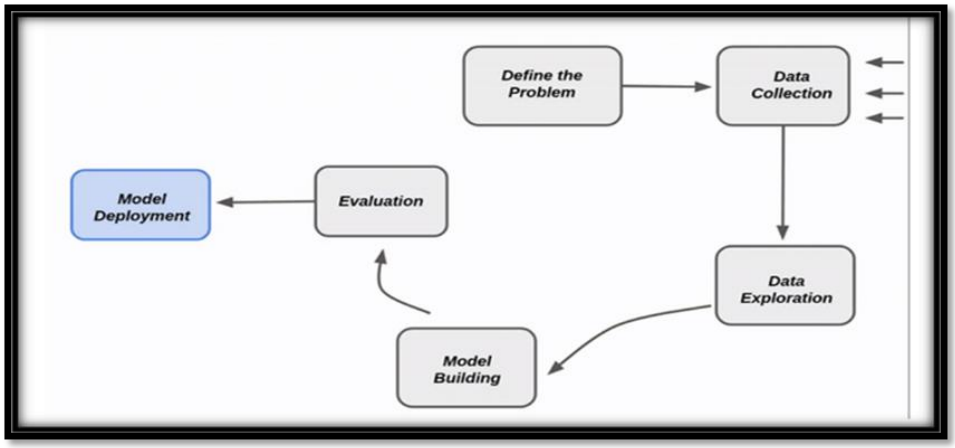


Fig 8 Model Deployment of CNN

The final stage is deployment and now here created a flask for serving the model. To creating a web app that Identifies 5 different types of flowers, we need two routes on our flask app:

1. An index page route for the users to select the image.
2. A predict route to make inferences from our saved model.

The predict function should take an image given by users and send it to the model. The last step of deploying into the cloud, created an interface which enables users to choose images from the browser. Here, JavaScript & CSS is being used and render a canvas on the HTML page.

Model Deployment is one of the last stages of any machine learning project. In a typical machine learning and deep learning project, we usually start by defining the problem statement followed by data collection and preparation, understanding of the data and model building.

Performance Analysis and Result

To measure the performance of the classification algorithms can be obtained through accuracy, precision, recall, and F-measure.

Evaluation Metrics

Evaluation metrics are used to measure the quality of the statistical or machine learning model. There are many different types of evaluation metrics available to test a model. These include classification accuracy, logarithmic loss, confusion matrix, and others.

Accuracy – This shows that how many data points are predicted correctly.

Precision – This refers to the metric for binary classifier measures the correctness among all positive labels.

Recall – It also used for binary classifier measures how many +ve labels are successfully predicted amongst all +ve labels.

F1-measure – It is used to measure a tests accuracy. It's a harmonic mean between the precision & recall ranging [0,1].

Confusion matrix – It gives an output like a matrix and describes the complete performance of the model having parameter such as TP, TN, FP&FN.

Micro average - is the precision/recall/f1-score calculated for all the classes.

Macro average - is the average of precision/recall/f1-score.

Weighted average - is just the weighted average of precision/recall/f1-score

Metrics	F1-Score	Support
Accuracy	0.89	827
Macro Avg	0.76	827
Weighted Avg	0.89	827

Table 1 Measuring the Model Performance

Table 2 Evaluation of the Model Performance

Flowers	Precision	Recall	F1-Score	Support
Daisy	0.91	0.98	0.94	402
Rose	0.88	0.90	0.89	192
Dandelion	0.90	0.85	0.87	156
Sunflower	0.84	0.69	0.76	52
Tulip	0.50	0.24	0.32	25

Metrics	F1-Score	Support
Accuracy	0.89	827
Macro Avg	0.76	827
Weighted Avg	0.89	827

Table 3 Performance measures

Experimental Results

For the experimentation following hardware and software have been used

Hardware Specification:

1. Processor – Intel(R)Core™ i5-8250U Central Processing Unit @ 1.60GHz 1.80Ghz

2. RAM – 8.00GB, x64-based processor

Software Specification*:*

1. Jupyter notebook /Google collab/ Sypder

2. Tensorflow framework

3. Windows 10 edition

The training and testing were performed using the Anaconda Software. All results reported have been taken after 10 epochs. During training of our model, we have used relu activation function in our hidden layer and it is proven that relu (and leaky relu) are always provide a good result when we use it for our hidden layer, the reason of applying nonlinear function to our model is to introduce the complexity of the data as our real world data are nonlinear in nature. For optimizing our model, we have Adam optimizer and we can use any optimizer, Adam is the right choice after testing with several optimizer like RMSprop and SGD. And for loss we have taken Mean squared error value. If error is less our model is good for the data. And after several testing and tuning parameters such as batch size, epochs, we found some results as follows;

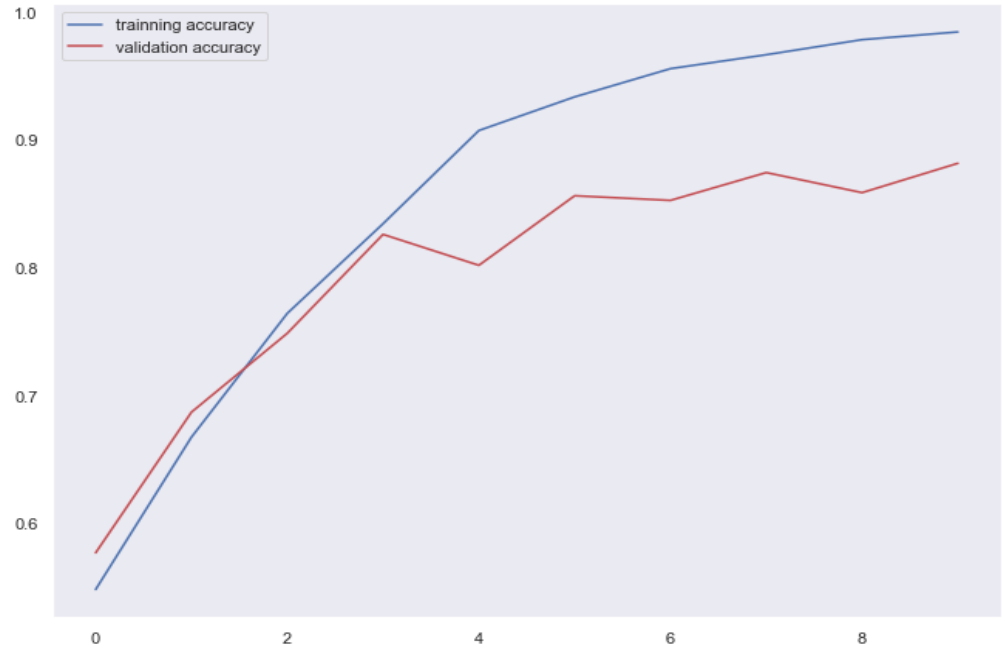


Fig.9. Graph of training vs. validation accuracy

Observation: This graph represents the training accuracy vs. validation accuracy. Here training accuracy and validation accuracy both intersect at a point and nearly close to each other.

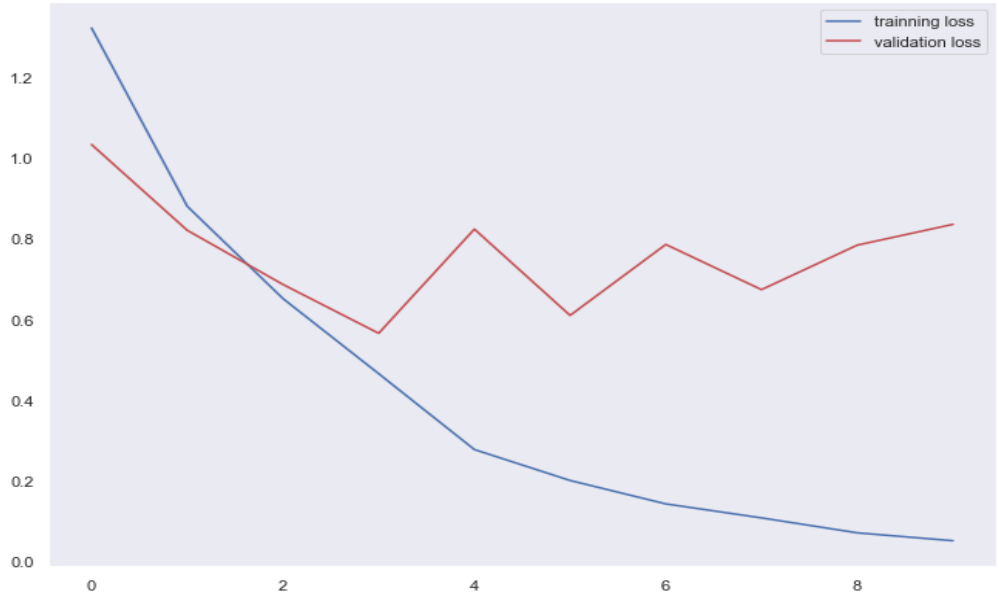


Fig.10. Graph of training loss vs. validation loss

Observation: The graph represents the training loss vs validation loss given by the model during training. After 10 epochs we found 0.06003 error (MSE) for our training and validation and in this case our model we have taken is CNN. For minimizing the error in testing we have several ways, either we can provide more training data or we can tune the hyperparameter. For this paper, we are going to tunning the parameter using Keras tuner to minimize the overall model error.

```
array([[  0, 204,   0,   1,   0],
       [  0, 254,   2,   0,   0],
       [  0, 190,   0,   0,   0],
       [  0, 174,   0,   0,   0],
       [  0, 252,   1,   0,   1]], dtype=int64)
```

Fig.11. Confusion matrix of SVM on gray-scale

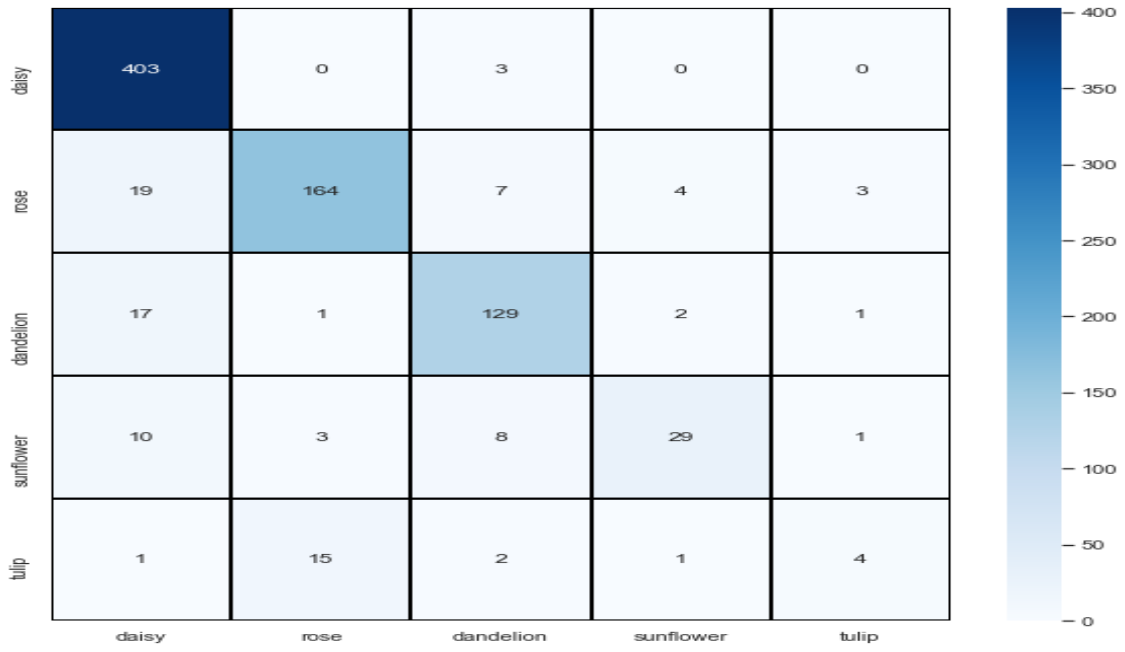
It gives matrix as an output and describes the complete model performance. Here in the SVM confusion the TP & TN is very low, so the model performance is very low. This the confusion matrix of SVM Model implemented on Gray-scale images having accuracy 24% which is very low, here SVM fails in feature extraction process of multi-class labelling .

Fig.12. Confusion Matrix of SVM color images

```
array([[502,   0,   0,   0,   0],
       [ 80, 163,   0,   0,   0],
       [110,   0,  86,   0,   0],
       [ 43,   0,   0,  14,   0],
       [ 35,   0,   0,   0,   1]], dtype=int64)
```

But when we use coloured images the accuracy increase that is 74% in SVM.

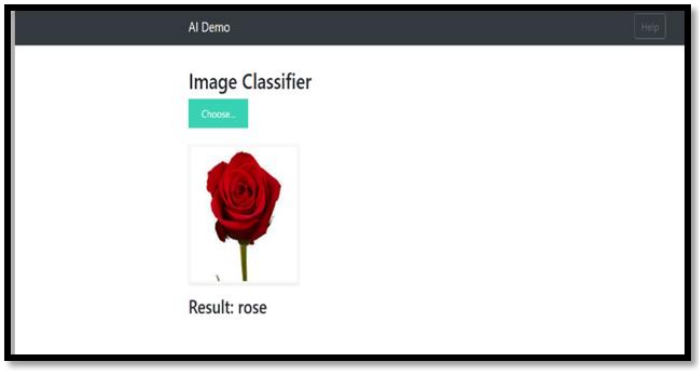
Fig. 13 Confusion Matrix



Observations: The confusion matrix of CNN model is very good as compared to SVM & Random-Forest Model as here FN&FP such as Type 1 & Type 2 error is very low.

The micro framework web application which does not require particular tools or libraries. Flask is a micro web framework written in Python. It is classified as a microframework because it does not require particular tools or libraries. It has no database abstraction layer, form validation, or any other components where pre-existing third-party libraries provide common functions. Flask API is a drop-in replacement for Flask that provides an implementation of browsable APIs.

Fig14 Micro web app



Conclusion

This Flower identification and classification using CNN is very effective model especially for image classification and also helps to do feature extraction automatically where as in Machine Learning algorithms manually feature extraction can be done. Machine learning Algorithms not worked so well as there was no tunning of hyperparameter and also it cannot classify multi class related problems .For this we need advance ML algorithms. But here CNN classify finely without any hyperparameter tunning and also worked easily .At last, deployment of model using flask framework that is micro web application because it does not really use particular tools or libraries .

References

1. Das, Madirakshi, R. Manmatha, and Edward M. Riseman. "Indexing flower patent images using domain knowledge." *IEEE Intelligent Systems and their Applications* 14.5 (1999): 24-33.
2. Shi, Lin, Zhigang Li, and Dingli Song. "A Flower Auto-Recognition System Based on Deep Learning." *IOP Conf. Series: Earth and Environmental Science*. Vol. 234. 2019.
3. Saitoh, Takeshi, and Toyohisa Kaneko. "Automatic recognition of wild flowers." *Systems and Computers in Japan* 34.10 (2003): 90-101.
4. Nilsback, M-E., and Andrew Zisserman. "A visual vocabulary for flower classification." *2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'06)*. Vol. 2. IEEE, 2006.
5. Sunpetchniyom, Treepop, Saowaluk Watanapa, and Rungkarn Siricharoenchai. "Flower search by image on mobile phone." *2012 6th International Conference on New Trends in Information Science, Service Science and Data Mining (ISSDM2012)*. IEEE, 2012.

6. Cho, Siu-Yeung, and Peh-Ti Lim. "A novel virus infection clustering for flower images identification." *18th International Conference on Pattern Recognition (ICPR'06)*. Vol. 2. IEEE, 2006
7. Cıbuk, Musa, et al. *"Efficient deep features selections and classification for flower species recognition." Measurement 137 (2019): 7-13.*