DIGITAL IMAGE PROCESSING

Project Report

TITLE:
FLOWER RECOGNITION SYSTEM USING CNN

DEPARTMENT:
SOFTWARE ENGINEERING

SUBMITTED BY:
HAFSA ANUM
SYEDA SAMANA BATOOL

ENROLLMENT NO:
2017-BSE-042
2017-BSE-066

SUBMITTED TO:
MAM MEHREEN SIRSHAR

SEMESTER:
VI-B

SESSION:
2017-2021

Contents

1

<u>FLOWER RECOGNTION SYSTEM</u>

1.   INTRODUCTION:

Digital image processing is used for extracting useful information from different images having different background appearance. By using image processing techniques, we can do various task like classification, clustering, image recognition, pattern or character recognition and object detection from an images or videos. The dataset of flowers contains the flower images are taken in natural environment where the shine of light changes with the time and weather. Thus, flower classification is an interesting and challenging problem.

Nature has many different kinds of flowers, similarity in some features is found between the flowers.

- o For example, many flowers share the red color. But they do not necessarily share the same shape. These similarities and differences highlight the difficulty of identifying each flower species automatically.

2.   OBJECTIVE:

There are almost 250,000 named species of flowering plants in the world. Most people don't have Knowledge about these flowers and in order to know about them, people usually have to use flowers guide books or use the relevant websites on the Internet to browse the information using keywords. Usually, these keyword searching is not practical for a lot of people. So, recognizing flowers from there images using normal ways like websites on the Internet using search engines and search keywords or via flowers guide books are not efficient and consuming a lot of time and hard to bring the right result. Hence its more effective to use an automated recognition system.

3.   ALGORITHM:

There are several numbers of methods and algorithms for recognition but we will use Convolutional Neural Networks (CNN) because neural networks are basically designed for pattern recognition.

i.   Convolutional Neural Network:

A **Convolutional Neural Network (CNN)** is a Deep Learning algorithm which can take in an input image, assign importance (learnable weights and biases) to various aspects/objects in the image and be able to differentiate one from the other.

ii.   A Convolutional Neural Network is Composed of:

- ### Convolutional layers
  The first ConvLayer is responsible for capturing the Low-Level features such as edges, color, gradient orientation, etc. With added layers, the architecture adapts to the High-Level features as well, giving us a network, which has the wholesome understanding of images in the dataset.

- ### ReLU layers
  ReLu refers to the Rectifier Unit, the most commonly deployed **activation** function for the outputs of the **CNN** neurons.

- ### Pooling layers
  The Pooling layer is responsible for reducing the spatial size of the Convolved Feature. There are two types of Pooling: Max Pooling and Average Pooling. **Max Pooling** returns the **maximum value** from the portion of the image covered by the Kernel. On the other hand, **Average Pooling** returns the **average of all the values** from the portion of the image covered by the Kernel.

- ### Fully connected layer
  Now that we have converted our input image into a suitable form for our Multi-Level Perceptron, we shall flatten the image into a column vector. The flattened output is fed to a feed-forward neural network and backpropagation applied to every iteration of training. Over a series of epochs, the model is able to distinguish between dominating and certain low-level features in images and classify them using the **SoftMax Classification** technique.

A classic **CNN architecture** would look something like this:

Input ->Convolution ->ReLU ->Convolution ->ReLU ->Pooling ->ReLU ->Convolution ->ReLU ->Pooling ->Fully Connected

---

A CNN **convolves** (not convolutes…) learned features with input data and uses 2D convolutional layers. This means that this type of network is ideal for processing 2D images. Compared to other image classification algorithms, CNNs actually use very little preprocessing. This means that they can **learn** the filters that have to be hand-made in other algorithms. CNNs can be used in tons of applications from image and video recognition, image classification, and recommender systems to natural language processing and medical image analysis.

iii.   Working Of CNN:

- starts with an input image
- applies many different filters to it to create a feature map

- applies a ReLU function to increase non-linearity
- applies a pooling layer to each feature map
- flattens the pooled images into one long vector.
- inputs the vector into a fully connected artificial neural network.
- processes the features through the network. The final fully connected layer provides the "voting" of the classes that we're after.
- trains through forward propagation and backpropagation for many, many epochs. This repeat until we have a well-defined neural network with trained weights and feature detectors

4.   LANGUAGE:

There are many efficient languages for image processing but we are using **python** because it's easy, has a lot of functions and support open source libraries. As python is a general-purpose programming language with a number of potential advantages for image handling and data presentation. These include NumPy for the array manipulation needed in image processing, and matplotlib, for display of graphs and data plots etc.

5.   ENVIRONMENT:

We will use Jupiter notebook (Anaconda). The **Jupiter** Notebook is a web-based interactive computing platform. The notebook combines live code, equations, narrative text, visualizations, interactive dashboards and other media. Moreover, it's one of the best platforms for executing python code.

6.   DATASET:

Taken from Kaggle. This dataset contains **3242** images of flowers. The data collection is based on the data flicr, google images, yandex images. You can use this dataset to recognize flowers from the photo.

The pictures are divided into five classes:

      i. Dandelion

     ii. Sunflower

    iii. Rose

    iv. Tulip

     v. Daisy

Photos are not reduced to a single size; they have different proportion.

7. SYSTEM ARCHITECTURE:

i.   Importing Various Modules:

We have imported NumPy, CV2, Matplotlib, label encoder etc.

ii.   Preparing the Data:

We have Used train_test_split function to not divide the dataset manually. It will make random partitions for the dataset.

### iii.   Modelling:

We modeled the system using CNN because it is useful for **image** recognition.

iv.   Evaluating the Model Performance:

As we have divided the dataset randomly. On the basis of this our model is accurately predicted the different flower species

8. OUTPUT:

We have compared the flowers with the misclassified indexes (means with the dataset our system is not trained). So, by comparing the classes the system is giving following input with matching flower species.
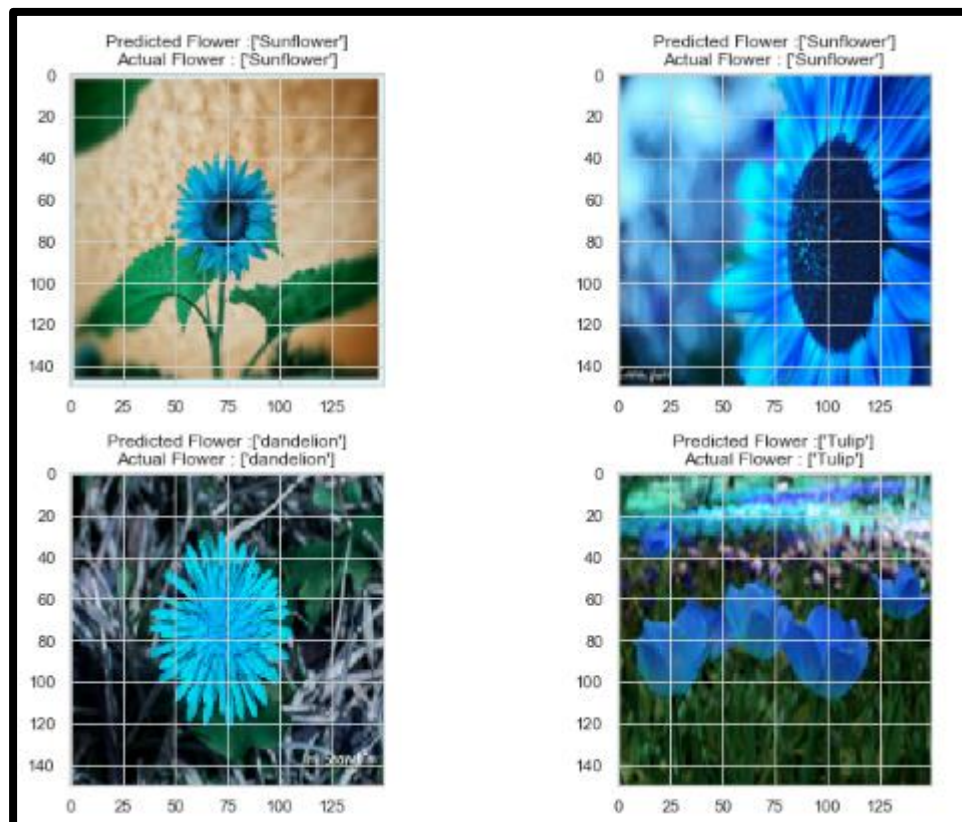
9.   RESULT:

Th Flower Recognition Model is accurately predicting the different species of flowers and giving output by matching the species as actual flower and the predicted flower type. This Automated flower recognition will be helpful for the people with limited experience in flower species to recognize the flowers.

****