Flower Recognition Using Deep Learning

Rohit Sangale, Rushabh Jangada, Anshu De*, Nikhil Sanga, Prof. Sujit Deokar

Department of Instrumentation and Control Engiineering, Vishwakarma Institute of Technology,University of Pune,Pune, India

A B S T R A C T

We have developed a deep learning network for classification of different flowers. For this, we have used Visual Geometry Group's 102 category flower data-set having 8189 images of 102 categories from Oxford University. In this method we have used Convolutional Neural Network architecture for the classification purpose. By keeping all hyper-parameters for the architecture, we have found the Training Accuracy of and Testing Accuracy of . These results are extremely good when compared to random classification accuracy of 0.98%. This method for classification of flowers can be implemented in real-time applications and can be used to help botanists for their research as well as camping enthusiasts.

Keywords: deep learning network, Neural Network architecture, Flower Recognition

1. Introduction

Flowers are everywhere around us. They can feed insects, birds, animals and humans. They are also used as medicines for humans and some animals. A good understanding of flowers is essential to help in identifying new or rare species when came across. This will help the medicinal industry to improve. The system proposed in the paper can be used by botanists, campers and doctors alike. This can be extended as an image search solution where photo can be taken as an input instead of text in order to get more information about the subject and search accordingly for best matching results.

As the classification of flower species is an important task, it is already in research and many different approaches have been developed. Previously, methods like Deformable Part Models , Histogram of Oriented Gradients and Scale invariant feature transforms were used for feature extraction, linear classifiers and object detectors. Later the work was focused onsegmentation and classification using manual feature engineering. But nowadays, state-of-art performance is achieved by Convolutional Neural Networks. CNNs have fulfilled the demand of robustness and have removed the need of hand crafted features. They are similar to Artificial Neural networks but does not require feature engineering. Each neuron receives some inputs, performs a dot product and optionally follows it with a non-linear operation.

For using CNNs, a large amount of data is required for training. We have used the Visual Geometry Group's 102 category flower data-set used in having 8,189 images spread over 102 categories from Oxford University. We split 15% of the total images for validation set and 15% for test set.This makes the application less computational expensive.

2. Methodology

A CNN consists of an input and an output layer, as well as multiple hidden layers. The hidden layers of a CNN typically consist of convolutional layers, pooling layers, fully connected layers and normalization layers.

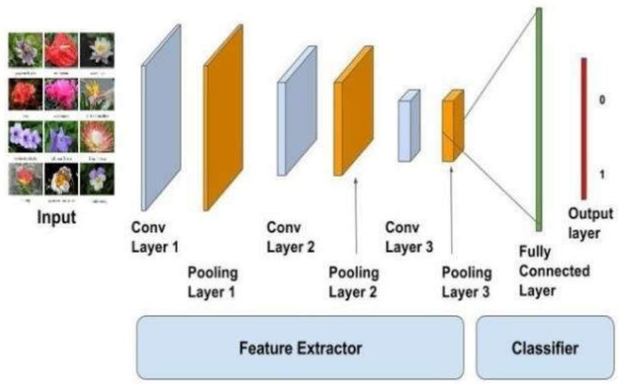

Fig 1 CNN

Image Input Layer

This layer consists of the input of the network i.e. image here.The size of input to our network is 64 * 64.

Convolutional

Convolutional layers apply a convolution operation to the input, passing the result to the next layer. The convolution emulates the response of an individual neuron to visual stimuli. Each convolutional neuron processes data only for its receptive field. A fully connected layer for a (small) image of size 100 x 100 has 10000 weights for each neuron in the second layer. The convolution operation brings a solution to this problem as it reduces the number of free parameters, allowing the network to be deeper with fewer parameters. For instance, regardless of image size, tiling regions of size 5 x 5, each with the same shared weights, requires only 25 learnable parameters. In this way, it resolves the vanishing or exploding gradients problem in training traditional multi-layer neural networks with many layers by using back propagation.

We have used total three Convolutional Layers all having input image of size 64*64.The Kernel of first layer is of the size 5 *5 and remaining two have of the size 3*3.The function used is Conv2D().

Pooling

Convolutional networks may include local or global pooling layer, which combine the outputs of neuron clusters at one layer into a single neuron in the next layer.For example, *max pooling* uses the maximum value from each of a cluster of neurons at the prior layer.Another example is *average pooling*, which uses the average value from each of a cluster of neurons at the prior layer.We have used Maxpooling in our project.The function that does maxpooling is MaxPooling2D**.** The Pool Size for pooling we defined is 2*2.

ReLU layer

ReLU is the abbreviation of Rectifier Linear Unit. This layer applies the non-saturating Activation Function. It increases the nonlinear properties of the decision function and of the overall network without affecting the receptive fields of the convolution layer.

Fully connected layer

Finally, after several convolutional and max pooling layers, the high-level reasoning in the neural network is done via fully connected layers.

Neurons in a fully connected layer have connections to all activations in the previous layer, as seen in regular neural networks. Their activations can hence be computed with a matrix multiplication followed by a bias offset.

Fig 2: Sample Images from database

3. Observations & Results

RESULTS FROM EPOCH 1 To 5

```
Epoch 1/50
122/122 [==============================] - 2s
15ms/step - loss: 1.5783 - acc: 0.3279 - val_loss: 1.5805 - val_acc: 0.3455

Epoch 2/50
122/122 [==============================] - 2s
13ms/step - loss: 1.5223 - acc: 0.3934 - val_loss: 1.6709 - val_acc: 0.3455

Epoch 3/50
122/122 [==============================] - 2s
13ms/step - loss: 1.4860 - acc: 0.3934 - val_loss: 1.8134 - val_acc: 0.2545

Epoch 4/50
122/122 [==============================] - 2s
13ms/step - loss: 1.5030 - acc: 0.3361 - val_loss: 1.4897 - val_acc: 0.3636

Epoch 5/50
122/122 [==============================] - 2s
13ms/step - loss: 1.3206 - acc: 0.4754 - val_loss: 1.5405 - val_acc: 0.3455
```

RESULTS FROM EPOCH 45 To 50

```
Epoch 45/50
122/122 [==============================] - 2s
13ms/step - loss: 6.4706e-05 - acc: 1.0000 - val_loss: 3.9607 - val_acc: 0.4727

Epoch 46/50
122/122 [==============================] - 2s
13ms/step - loss: 5.7887e-05 - acc: 1.0000 - val_loss: 3.9892 - val_acc: 0.4727

Epoch 47/50
122/122 [==============================] - 2s
13ms/step - loss: 5.2743e-05 - acc: 1.0000 - val_loss: 4.0203 - val_acc: 0.4909

Epoch 48/50
122/122 [==============================] - 2s
13ms/step - loss: 5.0711e-05 - acc: 1.0000 - val_loss: 4.0247 - val_acc: 0.4727

Epoch 49/50
122/122 [==============================] - 2s
13ms/step - loss: 4.5728e-05 - acc: 1.0000 - val_loss: 4.0538 - val_acc: 0.4909

Epoch 50/50
122/122 [==============================] - 2s
13ms/step - loss: 4.2984e-05 - acc: 1.0000 - val_loss: 4.0671 - val_acc: 0.4909
```

4. Conclusion

SOME KEY HIGHLIGHTS OF THIS EXAMPLE ARE:

- Code to train, evaluate, and deploy image classification models.

- Demo images provided, but easily adapt0able to use own image dataset.
- State-of-the-art expert features implemented to train high accuracy models.
- Interactive model development with Keras in Python.

Acknowledgements

We would like to express our special thanks of gratitude to our Guide Prof. Sujit Deokar as well as our Head Of The Department Dr.Shilpa Sondkar and our Co-Ordinater Prof. Archana Chaudhari who gave us the golden opportunity to do this wonderful project on the topic Flower Recognition Using Deep Learning , which also helped doing a lot of Research and We came to know about so many new things we are really thankful to them..

References

Nilsback, M., Zisserman, A.: ‘Automated flower classification over a large number of classes’. Proc. Sixth Indian Conference on Computer Vision, Graphics & Image Processing, Bhubaneswar, India, December 2008, pp. 722–729 6

Nilsback, M., Zisserman, A.: ‘A visual vocabulary for flower classification’. Proc. IEEE Conference on Computer Vision and Pattern Recognition, New York, NY, June 2006, 2, pp. 1447–1454

Miao Jinquan, Cao Weiqun (2014). Extensible Flower Species Recognition. Journal of Image and Graphics, 19(11), 1630--1638.

YIN Baocai, WANG Wentong, WANG Lichun (2015). Review of Deep Learning Research. Journal of Beijing Polytechnic University, 1, 48—59

Hsu, T., Lee, C., Chen, L.: ‘An interactive flower image recognition system’, Multimedia Tools and Applications, 2011, 53, (1), pp. 53–73.

Shelhamer, E., Long, J., Darrell, T.: ‘Fully Convolutional Networks for Semantic Segmentation’, IEEE Transactions on Pattern Analysis and Machine Intelligence, 2017, 39, (4), pp. 640–651