e-ISSN: 2582-5208
International Research Journal of Modernization in Engineering Technology and Science
Volume:02/Issue:05/May-2020 www.irjmets.com

IDENTIFICATION OF BUTTERFLY SPECIECS USING VGG-16

Samruddhi N Hegde[*1], Manjesh R[*2], P Sindhura[*3], Reema Rodrigues[*4], Sheethal A[*5]

[*1,3,4,5]BE Student, Computer Science & Engineering, Srinivas Institute of Technology, Mangalore, India.

[*2]Professor, Computer Science & Engineering, Srinivas Institute of Technology, Mangalore, India.

ABSTRACT

Butterflies are abundant species on the earth, and the task of identification of butterflies is complex. How to apply image processing methods to automatic identification of butterfly species is a hot issue in current research. In this paper, the problem of automatic detection and classification of butterfly species using buttery photographs is studied. A model needs to be trained first and training too deep it's not good because if done so then even background will be trained. So that small change in background it gives results as not a butterfly. Pooling method used and also tensor flow used for training CPU and GPU. Therefore only like 75% to 80% images should be trained remaining directly should be tested. This paper uses pre-trained model VGG-16 with Convolutional Neural Network (CNN) to classify butterfly images.

KEYWORDS: Convolutional Neural Network (CNN), Image Processing, Pooling, VGG-16, Tensor flow.

I. INTRODUCTION

In the world there are around 18,000 butterfly species were exists. These different butterflies are having different characters like unique patterns, large wings, etc., so they are named using there external morphological features. Butterflies plays a very important role in the environment by pollination and also it is a prey to many predators to balance the ecosystem. Currently, for the sake of further reveal the evolution of the species ecological status, scientists are focused and maintaining the diversity of species in each ecosystem, as their number have dramatically decreased. Therefore, the classification of species is crucial but complicated and difficult. Traditional methods of insect's taxonomy differentiate species of butterflies by analyzing the color and size of the wing spot, the wing veins and other anatomical features. Many automatic methods have been developed to help entomologists with identification. Here VGG-16 method is used which is shown in figure 1. First the image is to be trained then the data need to be augmented. Then the augmented data is to be pre-trained using the VGG-16, which pre-trained data need to be validated and feed the data to testing, after testing the butterfly species is identified.

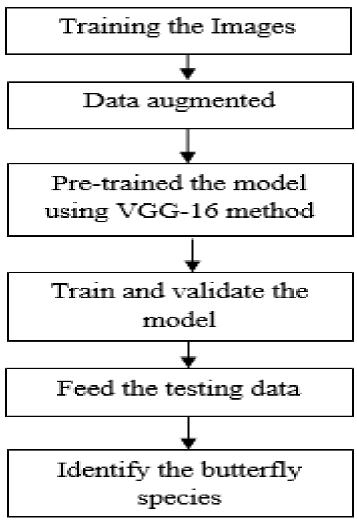

Fig-1: Block diagram

e-ISSN: 2582-5208
International Research Journal of Modernization in Engineering Technology and Science
Volume:02/Issue:05/May-2020 www.irjmets.com

II. LITERATURE SURVEY

As a part of this we a basic convolutional neural network from scratch will be developed, training should be done based on image dataset after which the evaluation of the model is the most necessary method to be applied. Hereafter the model accuracy for the images could be done by image data augmentation technique .Therefore at the final stage the pre-trained model VGG-16 will be leveraged which will be already pre-trained on a large dataset with variable range of categories for the feature extraction and the classification of the images. A research project Image Net is to develop a big database of annoted images. Pre-trained models like InceptionV1, Inception V2, VGG-19 and VGG-16 are already pre-trained, which comprises of diverse classes of images on ImageNet. These models are built from scratch and will be trained by the usage of high GPU's over millions of images consisting of thousands of image categories. As the model will be trained on large dataset, it will learn a fine representation of low level features like rotation, lighting, spatial, edges, shapes and these features can be shared over to enhance the knowledge sharing and act as a feature extractor in variant computer vision problems for newer images. These images could be of exactly different classes from the dataset, but the pre-trained model should still be able to extract features relevantly from images based on the transfer learning principle. In this paper the power of transfer learning will be unleashed by using an effective feature extractor, pretrained model -VGG-16 to classify classes even with fewer images for training.

III. METHODOLOGY

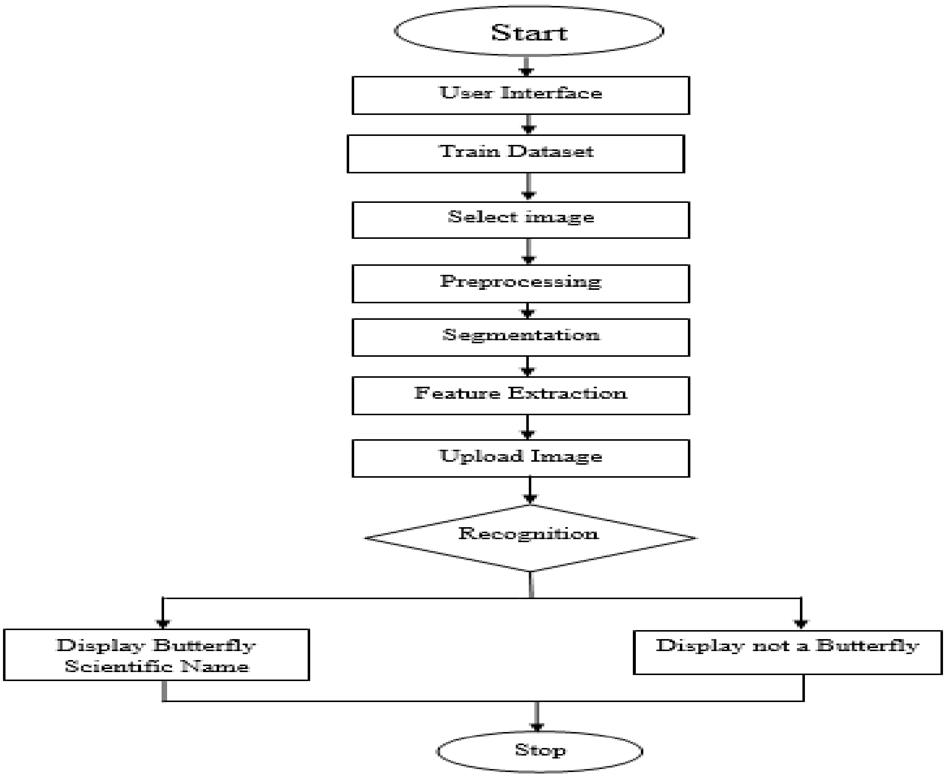

Fig-2: System Architecture of Identifying the Butterfly Species using VGG-16

The system is implemented using Anaconda a standard platform and coded using jupyter notebook. The aim of this project is to classify the images of butterflies using the VGG-16,which pre-trained model to evaluate the performance of the classification. A system flowchart symbolically shows how data flows throughout a system and how event controlling decisions are made. Initially the image is input, and the pre-processing of the input image takes place followed by segmentation. Later the images are classified according to their extracted feature. Finally, if input image is butterfly then the name of the butterfly is displayed else not a butterfly is displayed. Following figure shows the proposed process flow of the butterflies images classification. The proposed flow of application begins with uploading the image of butterfly. Image of butterfly will going through resize and cropping to 128x128 pixels. To classify image that has been resized, a trained VGG-16 model is used which produces the final outcome of the butterfly classification subsequently. After process of VGG-16 prediction is completed. The maximum probability will be calculated using array operations. Lastly, identified image of butterfly with their label or not a butterfly will be displayed. A total of eight hundred and thirty two images of butterflies were collected from online database. The size of images obtained are from different kind of sizes but then resize to 128x128 pixels. There are totally 10 classes of butterfly species each having its own description. Tensor Flow is an open source programming library for numerical calculation utilizing information stream charts. Nodes in the graph represent mathematical operations while the diagram edges speak to the multidimensional information clusters (tensors) conveyed between them. The versatile plan empowers you to send figuring to no less than one CPUs or GPUs in a work region, server, or PDA with alone. Tensor Flow has many deep learning libraries, including Keras. Keras is an exceedingly measured library written in Python and is fit for running either Theano or Tensor Flow. While Tensor Flow allows for training on both a CPU and GPU. Our model is implemented using CPU. There are five main layers used in each network architecture: A convolutional layer is analogous to a novel classifier, but its weight matrix size is not equal to that of the layer but is much smaller from it. In this layer the weight matrix is called a filter and it is "moved" over the whole layer. Pooling is a method of introducing a nonlinear function in a classifier which enables it to represent the whole training data. Max pooling is the most commonly used pooling techniques, some other techniques include average pooling. Dropout consists of randomly setting a fraction of input units to 0, or closing of a fraction of activation function by dropping their weights to 0 during training time. Dropout is the most effective way of dealing with over fitting. Generally 25% to 50% percent are dropped out and is commonly implemented at the final layers of the network. ReLu is the activation function that is also used for introducing nonlinearity in the network. ReLu activation function is defined as max (x, 0). A Fully Connected Layer is the same as in an ordinary Neural Network. Neurons in this layer have connections to all other activation layers, allowing to activations to be computed using matrix multiplication. In addition to the layers used above, the loss function used in all models was a softmax loss function. The optimiser used in the network is stochastic gradient descent (SDG) with momentum. As compared to gradient descent, in which we update the weights after a complete traversal of training data, in stochastic gradient descent we update the weights after a batch of training data. It is observed that by the use of stochastic gradient descent the time required for the convergence is much smaller as compared to gradient descent. The momentum term ensures that the model does not get stuck at some local minimum and tries to ensure a global optimum. The network model build (including pre-trained VGG-16), using an L2 regularization rate of 0.01 and Gaussian initialization for weight initialization. In addition, the model is trained using batches of size 12 (due to the external memory limitation). The architecture of the model is described below.

Table-1

Model	
VGG-16	124x124x32
Max Pool	62x62x32
VGG-16	60x60x64
Max Pool	30x30x64
VGG-16	28x28x128
Max Pool	14x14x128
VGG-16	12x12x128

Max Pool	6x6x128
Flatten	1x1x512
Dropoot(10%)	1x1x512
Dense	1x1x256
Dense Softmax	1x1x10

IV. RESULTS AND DISCUSSION

From the above aspects we have observed that there are vast varieties of species of the butterflies found. So it is very to classify all of them. We have used the domain Machine Learning implemented the VGG-16 method which is used because it is one of the most efficient methodology under machine learning which has delivered about 92% accuracy.

V. CONCLUSION

By the analysis of the varies butterfly species we have come to conclusion saying that we have created a model which is trained on about 85% of the dataset, and tested on remaining 15% of the dataset. This Model Identifies the Butterfly class of that particular image which is fed as input. This Model also ensures that it does not identify the objects other than Butterfly. Future work should mainly focus on the proposed VGG-16 method could be employed for real-time butterfly not by considering images of butterfly. Also can be develop a web app or mobile app where the user can use this application anytime anywhere for butterfly species identification. And multi-classification, which focuses on rare species of butterflies and unknown species, can also be explored.

ACKNOWLEDGEMENTS

The authors would like to thank Mr. Manjesh R, Assistant Professor, Department of Computer Science & Engineering, Srinivas Institute of Technology for his valuable suggestions during the planning and development of this project work. And also we thank our parents for their support and encouragement throughout our work.

VI. REFERENCES

[1] K. Simonyan and A. Zisserman. Very deep convolutional networks for large-scale image recognition. Int. Conf. on Learning Representations, San Diego, CA, 2015.

[2] L. Wang et al.. Training deeper convolutional networks with deep supervision. arXiv: 1505.02496, 2015.

[3] M. Abadi et al. Tensorflow: Large-scale machine learning on heterogeneous distributed systems. arXiv: 1603.04467, 2016.

[4] M. D. Zeiler and R. Fergus. Visualizing and understanding convolutional neural networks. European Conf. on Computer Vision, Zurich, Switzerland, 2014.

[5] F. Perronnin, J. Sanchez, and T. Mensink. Improving the Fisher kernel for large-scale image classification. In Proc. ECCV, 2010.

[6] G. Huang, Z. Liu, and K. Q. Weinberger. Densely connected convolutional networks. arXiv preprint arXiv: 1608.06993, 2016.

[7] ATC (1996). Seismicvaluation and retrofit of Concrete buildings, D. C. Ciresan et al. Flexible, high performance convolutional neural networks for image classification. Int. Joint Conf. on Artificial Intelligence, Barcelona, Spain, 2011.

[8] R. Collobert, K. Kavukcuoglu, and C. Farabet. Torch7: A MATAB-like environment for machine learning. Conf. on Neural Information Processing Systems: BigLearn Workshop, Granada, Spain, 2011.

e-ISSN: 2582-5208
International Research Journal of Modernization in Engineering Technology and Science
Volume:02/Issue:05/May-2020 www.irjmets.com

[9] A. G. Howard. Some improvements on deep convolutional neural network based image classification. CoRR, abs/1312.5402, 2013.

[10] M. D. Zeiler and R. Fergus. Visualizing and understanding convolutional neural networks. European Conf. on Computer Vision, Zurich, Switzerland, 2014.