2020 2nd International Conference on Advances in Computing, Communication Control and Networking (ICACCCN) | 978-1-7281-8337-4/20/$31.00 ©2020 IEEE | DOI: 10.1109/ICACCCN51052.2020.9362804

Image Classification Using Deep Neural Network

Vaibhav Tiwari[1], Chandrasen Pandey[2], Ankita Dwivedi[3],Vrinda Yadav[4]
Computer Science and Engineering
Centre for Advanced Studies, AKTU, Lucknow, India
vaibhavtiwari09@gmail.com[1], developer.chandrasen@gmail.com[2], ankitadwivedikit007@gmail.com[3], vrinda@cas.res.in[4]

***Abstract*—Image Classification is widely used in various fields such as Plant leaf disease classification, facial expression classification. To make bulky images handy, image classification is done using the concept of a deep neural network. The proposed work implemented the VGG16 model to classify an image into one of the categories like living and non-living that is further classified into several classes like an animal, human, selfie, group photos, place, wallpaper, vehicles, etc. The paper contributes a methodology for a more accurate classification of images instead of image feature extraction or image segmentation. The proposed work established a promising accuracy of 99.89%.**

***Index Terms*—Deep Neural Network, VGG, Image Classification, Convolutional Neural Network (CNN)**

I. Introduction

Deep Neural Network is a widely used technique in different fields such as Self Driving, Health Care, Automatic Machine Translation, etc. Classification of images with the help of a deep neural network is a recent approach to achieve good results. This paper presents a comparative study of different neural networks that can be used to classify the image datasets in different classes with the help of pre-trained VGG models. Feature extraction of images is computationally expensive, the pre-trained neural network is the better choice for image classification[4]. In this paper, we used VGG16 as a pre-trained model to classify the images into five classes, each class with approx 2000 images. The image dataset is first preprocessed and then the VGG16 model is trained. Images may differ in size, hence they are preprocessed before they can be used to train the VGG16 model. Specifically, convolutional neural networks trained on large volumes of annotated data are capable of achieving the state of the art results. This paper will begin with a discussion of the baseline convolutional neural network architecture. We will analyze the results from experiments with VGG16 and also explore several different performances of pre-trained models.

The following characterizes the fundamentals needed for comprehension of the CNN.

A. Convolution Layer

The convolution layer is made up of a set of independent filters. Each filter slides over the image and creates feature maps that get different aspects of an image.CNN uses convolutions to joined fetch features from the local domain of given input. Most CNNs comprise a consortium of convolutional, pooling, and affine layers. CNN's offer fantastic performance on visual identification jobs, where they achieve the state of the art.

Pooling Layer

Pooling was at first developed to assist to make CNN layers put up distortions, as in the scale-invariant feature transform (SIFT) descriptor with a 4×4 sum pooling grid. This layer allows features to move relative to each other resulting in the deep meeting of features even in the light of small distortions. There are some other profits of pooling, as a reduction of the spatial dimension of the feature map degrading the number of parameters. This simplifies the overall complexity of the model. Though sum and max-pooling are a bit outdated as mostly, nowadays, stridden convolution is used. The motive of stridden convolution is to jump some domains during the convolution operation consequently resulting in capable convolution operation with the reduced spatial dimension of the output.

B. Fully Connected (FC) Layer

The fully connected layer in the CNN symbolises the feature vector for input. This feature vector or tensor or layer takes information that is important to the input. When the network trains, this feature vector then uses for classification, regression, or make an input into another network like Recurrent Neural Network for translating into another type of output, etc. It is also being used as an encoded vector. During training, this feature vector is used to calculate the loss and helps the network to make it trained. The convolution layers before the fully connected layer keep information related to local features in the input image i.e. edges, blobs, shapes, etc. Every Convolutional layer keeps many filters that symbolize one of the local features. The fully connected layer keeps composite and aggregated information from all the important convolution layers.

C. Number of Parameters

The convolution layer consists of 2 types of parameters: biases and weights. The summation of weight and biases is an overall wide variety of parameters in the convolution layer. The variability of parameters is affected by kernel size. For a convolution layer with the clear out the length of 3×3, enter of length 25 with 3 channels, the quantity of weights in this layer is $3 \times 3 \times 25 \times 3$. The dimension of the input facts is the number of biases. Thus, the wide variety of parameters is $3 \times 25 \times 3 + 25$ on this convolution layer. There can be no parameters since the hyperparameters include stride, pool size, and zero paddings for the pooling layer. In a DNN (Deep Neural Network) shape, the last pair of layers is often the FC (Fully Connected) layer. The numeral of biases is the variation

ISBN: 978-1-7281-8337-4/20/$31.00 ©2020 IEEE

of neurons in the presentation layer. Instead of a pair of layered (DNN) Deep Neural Network, the wide variety of parameters is the summation of parameters in every layer.

The formation of the paper follows as Section II talk about the applied models to classify the images and the comparative study of all other models applied in the experiment. Section III states the implementation of the VGG16 model. Section IV represents the experimental estimated results that show the accuracy of the proposed model, Section V comprises the conclusion and the future work of this paper.

II. Applied Models

A. Visual Geometry Group (VGG16)

The Visual Geometry Group at Oxford University acquires the 16 layers VGG network for the state of the art results in the competition named ILSVRC-2014. The greater depth of its network is the fundamental characteristic of its structural design. In this the RGB images go through the five blocks of convolution layers and in every block, there are 3x3 numbers of channels/filters. The CNN layer is padded and stride fixed to 1 so that spatial resolution is conserved after convolution (which means that for 33 filters there is the padding of 1 pixel). The blocks are divided by the max- pooling layer. The stride is 2 in max-pooling which is executed over 22 windows. All the five blocks of the convolution layer are followed by FC (fully connected) layer. The output of the final layer gives class probabilities in soft max-layers. The full design architecture is illustrated in Fig 1.

B. Comparative Study

In this paper, we firstly use the Baseline CNN (Convolutional Neural Network) Model to classify the images in two domains, the first one is a living being and the other is a non-living being. Baseline CNN Model achieves accuracy by 55.075%. Based on the study[8][9], we use different models and increase the number of classes in each model. Three-Block VGG Model we classify the images dataset in three classes the first is nature, the second is animal and the third one is vehicles. Three-Block VGG Model has three Convolutional layers and 128 filters in the last layer and we use sigmoidal as the activation function. The accuracy of the model increases by up to 74.561%.

To achieve more accuracy, we used Data Augmentation in the same VGG3 Model. In the Data Augmentation, the training dataset is artificially expanded such that the images are scaled 10% horizontal and vertical. In this model, after increasing one more class named as a group, there is a gradual decrease in the accuracy level 61.0% From our observations we use the VGG 16 model with softmax activation function and loss function sparse categorical Cross entropy and achieved accuracy level of 98.97%.In this model, we classify the images into 5 different classes named nature, animal, vehicle, group photo, and selfie.

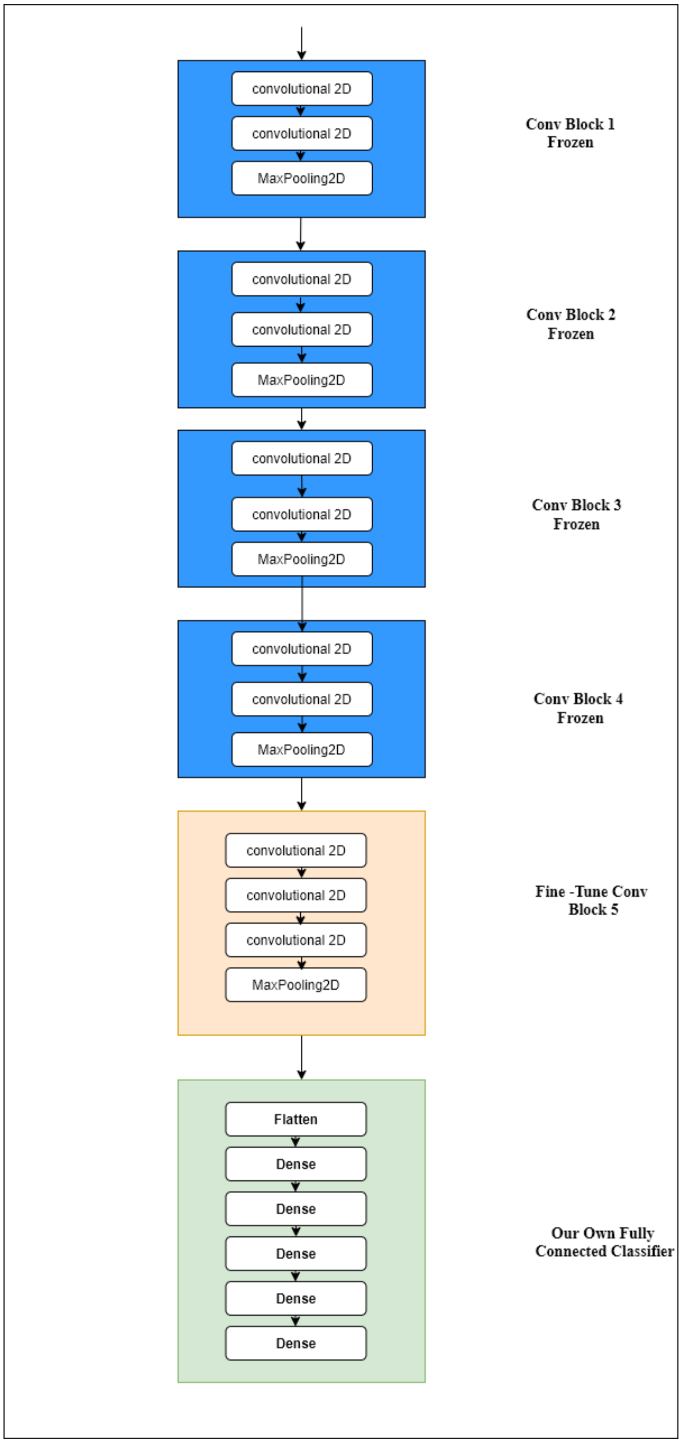

Fig.1 Flow diagram of Modified VGG 16

III. Implementation of VGG 16 Model

The VGG 16 models consist of 16 layers which give a good output in the challenge of ImageNet image classification. The model consists of 2 divisions: The feature extractor consist of VGG blocks and the classifier consist of FC and output layer. In this paper, a different classifier layer is added and utilize the feature extraction in design architecture. We didn't alter the weights of the convolutional layers, it is constant while training, and just trained new FC layers which will learn to infer the features extracted features aimed at binary classification from the design architecture or model. In the VGG-16 model, from the output side, the FC layer removed and added the new FC layers in output. The prerequisite of the design architecture is the specified input's shape and size which is (224, 224, and 3) in our case for the design model which referred that the updated design architecture lasts at the final max-pooling layer, then after the new classifier layer and a Flatten layer is added. The model ran in 10 epochs and attained an accuracy of 98.72%. This VGG16 model, trained on a particular ImageNet challenge dataset which is constituted to the input images with dimensions 224 x 224 pixels. The images are loaded from the image classification data set with this target size. Also, the images are expected to be centered by the design model. This is to get the mean pixel values from red, green, and blue channels and calculated on the ImageNet training dataset subtracted from the input. Using VGG 16 Model we made our own Fully Connected Classifier with one flatten layer and five dense layers as shown in Fig 1.

IV. EXPERIMENTAL RESULTS

An initial interesting point is that the common design principles of the VGG models since it performed best in the competition called ILSVRC 2014[10] and it is very simple and easy to comprehend and implement this modular construction of the architecture. The architecture includes the piling convolutional layers through minor filters of 3×3 along with the max-pooling layer. These layers composed and form a block, and the number of filters in every block augmented beside the network's depth in these blocks for example 32, 64, 128, and 256 for the first four blocks of the design model. To confirm the width and height of the output feature maps counterparts with the inputs, padding is utilized on convolutional layers. Commonly for finest practices, every layer uses the weight initialization and activation function called "ReLU". For example, a VGG-based design of three blocks has a single pooling layer and convolutional layer. We can see the accuracy of each model concerning its activation function and loss function in table I. Graphs are also shown depicting line plot for the accuracy and another for the loss of the design model on the training (red) and testing (blue) datasets. Fig. 2 denotes accuracy and loss of baseline model where the blue line denotes testing and the red line denotes training performance. Fig. 3 denotes the accuracy and loss of the VGG 3 model where the blue line denotes testing and the red line denotes training performance. Fig. 4 denotes the accuracy and loss of the VGG 16 model where the blue line denotes testing and the red line denotes training performance. Fig. 5 gives the final flow chart of the process involved in the experiment. Table II gives the number of images from various datasets [11], [12], [13], [14], [15] which is used in testing and training the data. The system on which the experiments are conducted is of 8th Generation i5, 8GB RAM, NVIDIA GeForce GTX 1050 Graphics and Anaconda Spyder software is used.

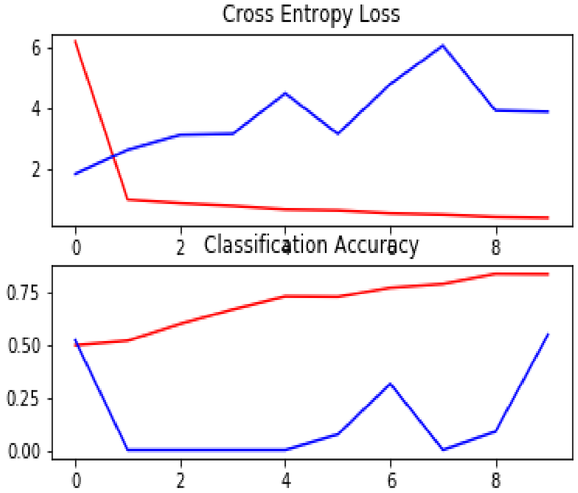

Fig.2 Baseline CNN Model

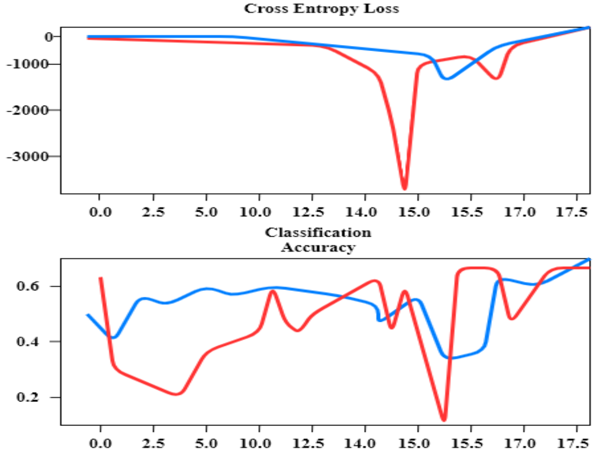

Fig.3 VGG 3 Model

TABLE I
COMPARISON OF DIFFERENT MODELS

Model Used	***Accuracy in(%)***	***Activation Function***	***Loss Function***
Baseline CNN	55.075	Sigmoid	binary_crossentropy
VGG 3	74.561	Sigmoid	binary crossentropy
VGG3+Data Aug..	61.404	Sigmoid	binary_crossentropy
VGG 16	98.97	Softmax	sparse categorical

V. CONCLUSIONS AND FUTURE WORK

In this study, the datasets are made from different sources and then all the images are categorically classified using VGG16. The images are classified into the two categories, firstly living and non-living and then further classified into subcategories such as living classified as nature, animal, human, and the human class categorized into group and selfie images.

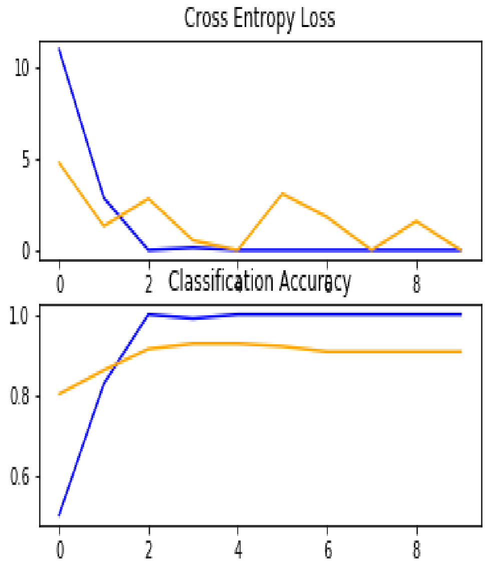

Fig. 4 VGG 16 Model

TABLE II DATASET USED

Name of Dataset	***Testing***	***Training***
Nature	2780	1524
Animal	15141	5080
Group	3576	1500
Selfie	20011	10277
Vehicle	3593	2401
Total	45101	20782

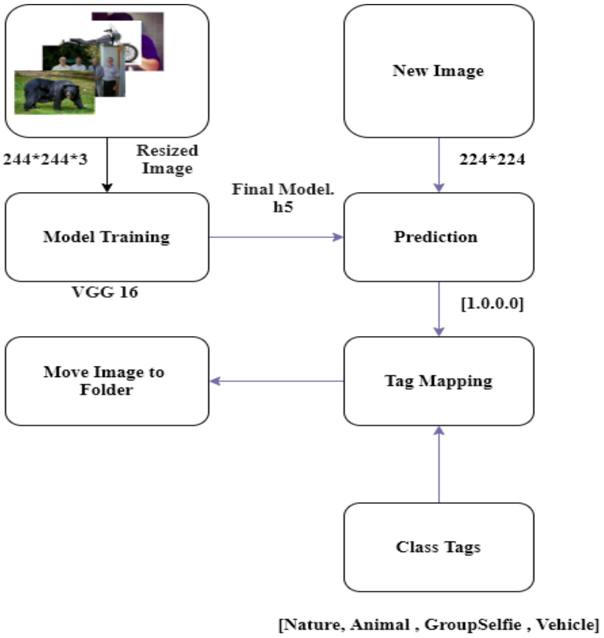

Fig.5 Process Flow

The non-living classified into the vehicle. Also using VGG16 we get 99.89% accuracy with minimal loss. It takes 1 hour 45 minutes for 1 Epoch while we ran 10 epochs to train our model, it took approx. 17 hours for the given dataset. Future work includes reducing the training time for the model and using VGG19 but the challenge is the accessibility of open-source big datasets for training model and the high computational time. It might be possible to carry out further research in this area only using GPU servers, unlike personal computing devices.

REFERENCES

[1]. D. Erhan, C. Szegedy, A. Toshev and D. Anguelov, ”Scalable Object Detection Using Deep Neural Networks,” 2014 IEEE Conference on Computer Vision and Pattern Recognition, Columbus, OH, 2014, pp. 2155-2162

[2]. Day O ,Khoshgoftaar T M . A survey on heterogeneous transfer learning[J]. Journal of Big Data, 2017, 4(1):29.

[3]. Wang Wenpeng, Mao Wentao, He Jianliang, et al. Smoke Recognition Method Based on Deep Migration Learning [J]. Computer Applications, 2017 (11): 144-149+161(in Chinese).

[4]. Wang Liwei, Li Jiming, Zhou Guomin, Yang Dongyong. Application of depth transfer learning in hyperspectral image classification[J/OL]. Computer engineering and application: 1-8 [2019-01-25] (in Chinese).

[5]. Planas, Santiago, et al. ”Performance of an ultrasonic ranging sensor in apple tree canopies.” Sensors11.3 (2011): 2459-2477

[6]. LI Yandong,HAO Zongbo,LEI Hang. Survey of convolutional neural network[J]. Journal of Computer Applications, 2016, 36(9): 2508-2515.

[7]. Xie J, Ding C, Li W, et al. Audio-only Bird Species Automated Identification Method with Limited Training Data Based on Multi-Channel Deep Convolutional Neural Networks[J]. 2018.

[8]. Tindall, Lucas and Cuong Manh Luong. “Plankton Classification Using VGG 16 Network.” (2017).

[9]. Elson, Jeremy and Douceur, John (JD) and Howell, Jon and Saul, JaredAsirra: A CAPTCHA that Exploits Interest-Aligned Manual Image Categorization,Proceedings of 14th ACM Conference on Computer and Communications Security (CCS).2007.

[10]. Olga Russakovsky*, Jia Deng*, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael Bernstein, Alexander C. Berg and Li Fei-Fei. (* = equal con- tribution) ImageNet Large Scale Visual Recognition Challenge. IJCV, 2015

[11]. Nature-http://arti.vub.ac.be/research/colour/data/imagesets.zip.

[12]. Animal- https://www.kaggle.com/search?q=ANIMALS

[13]. Group-
http://chenlab.ece.cornell.edu/people/Andy/ImagesOfGroups.html

[14]. Selfie - https://www.crcv.ucf.edu/data/Selfie/

[15]. Vehicle- https://www.kaggle.com/jessicali9530/stanford-cars-dataset