

An Analysis of Convolution Neural Network for Image Classification using Different Models

Sushma L
B.M.S College of Engineering

Dr. K.P. Lakshmi
Professor
B.M.S College of Engineering

Abstract:- This paper presents an analysis of the performance of the Convolution Neural Networks (CNNs) for image identification and recognition using different nets. A variety of nets are available to test the performance of the different networks. Popular benchmark datasets like ImageNet, CIFAR10, CIFAR100 are used to test the performance of a Convolution Neural Network. This study focuses on analysis of three popular networks: Vgg16, Vgg19 and Resnet50 on ImageNet dataset. These three networks are first realized using Keras and Tensorflow for image classification on ImageNet dataset. The random set of annotated images from the internet are subjected to these three networks for classification and analyzed for accuracy. It is observed that that ResNet50 is able to recognize images with better precision compared to Vgg16 and Vgg19.

Keywords: *Convolution Neural Networks, Image Classification, ImageNet, Vgg16, Vgg19, ResNet50, Keras, Tensorflow.*

I.INTRODUCTION

Artificial Intelligence (AI) is a multidisciplinary science that includes in making smart machines capable to perform tasks that are analogous to human intelligence. It is an intelligence shown by machines, similar to the natural intelligence displayed by humans and animals to ideally validate and perform actions that will achieve a specific goal in a better way. Deep learning or deep structured learning is a part of a broader family of machine learning techniques based on artificial neural networks. In deep learning, artificial neurons in the networks examine the large dataset and it will inevitably determine the underlying patterns without human interference [1]. In deep learning, a model/network learns to perform classification tasks on images, text, or sound. Deep learning models can achieve state-of-the-art accuracy on image recognition and classification, sometimes exceeding human-level performance [2].

Computer vision has various applications like image classification [3], image recognition and object detection [4]. Among those, image detection [5] is considered as the fundamental application and forms the basis for other computer vision applications [6]. Nowadays internet has an abundance of images, which is very handy for the development of applications and algorithms. There are major advances in image labelling, object detection, image recognition areas across the world [7]. Artificial neural networks have been used extensively for object detection [4] and recognition and this work focuses on suggesting a suitable architecture/network for image recognition and image classification [8].

Datasets like ImageNet [9], LabelMe [10] etc. that have millions of high-resolution labelled images and are freely available to build neural network models. These images belong to thousands of different categories. In 2012 a large deep convolution neural network, called AlexNet [11], designed by Krizhevsky showed excellent performance on the ImageNet Large Scale Visual Recognition Challenge (ILSVRC) [12] [20]. The success of AlexNet has become the inspiration of different CNN model such as ZFNet [13], VGGNet [14], GoogleNet [15], ResNet [16], DenseNet [17], CapsNet [18], SENet [19] etc. in the following years.

ILSVRC commonly called as 'ImageNet' is an image dataset that is built according to the WordNet hierarchy. According to WordNet hierarchy, the words and similar word phrases are grouped to form "synonym set" or "synset". More than 100,000 synsets in WordNet are designed to provide on an average of 1000 images to demonstrate each synset in ImageNet. Every Images are human-annotated and quality-controlled. ImageNet [20] will offer tens of millions of efficiently sorted images for maximum of the models in the WordNet hierarchy.

In this work, pre-trained Vgg16, Vgg19 and Resnet50 have been realized using Keras and Tensorflow and is deployed using python for image detection. The selected models have been trained using millions of human-annotated images of ImageNet dataset. Random images from the internet are then used for classification. The results are analyzed with respect to accuracy of prediction.

II.CONVOLUTIONAL NEURAL NETWORK

A Convolutional Neural Network, a Deep Learning algorithm, can take an input image, assign importance to several aspects/objects in the image and be able to distinguish one from the other. A typical CNN is composed of single or multiple blocks of convolution and sub-sampling layers, after that one or more fully connected layers and an output layer as shown in fig. 1.

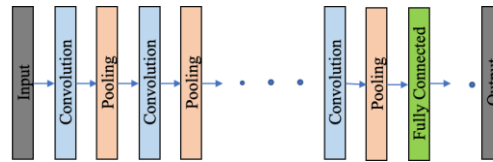


Fig. 1: Building block of a typical CNN

A. Convolutional Layer

The convolutional layer (conv layer) is the central part of a CNN. In a large Input image, a small section of the image is considered and we convolve them into a single output using a filter (Kernel). Fig. 2 shows typical convolutional operation.

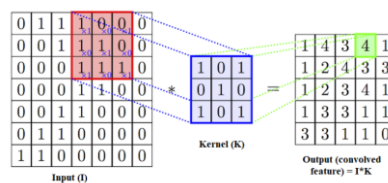


Fig. 2: Convolutional Layer

B. Sub-sampling or Pooling Layer

Pooling merely means down sampling of an image. It takes small region of the convolutional output as input and sub-samples it to produce a single output. Different pooling techniques like max pooling, mean pooling, average pooling is available. Max pooling takes largest of the pixel values of a region as shown in fig. 3.

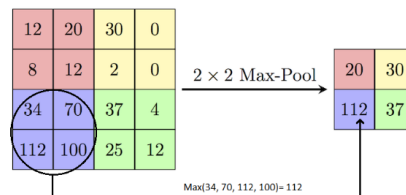


Fig. 3: Max Pooling operation

C. Non-Linear Layers

Neural networks and CNNs in precise rely on a non-linear “trigger” function to signal identification of likely features on every hidden layer. CNNs may use a range of certain functions —such as rectified linear units (ReLU) and continuous trigger (non-linear) functions—to proficiently implement this non-linear triggering.

ReLU -- A ReLU implements the function $y = \max(x, 0)$, thus the input and output sizes of this layer will be the same. It rises the nonlinear properties of the decision function and of the overall network without affecting any of the receptive fields of the convolution layer. ReLU layer functionality is demonstrated in Figure 4, with its transfer function plotted over the arrow.

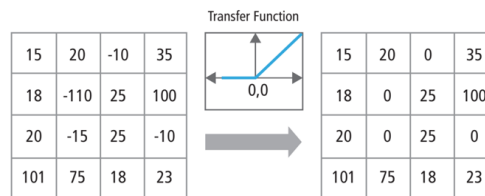


Fig. 4: Pictorial representation of ReLU functionality

D. Fully-connected Layer (FC Layer)

Fully connected (FC) layers are always used as the last layers of a CNN. This layer takes input from all neurons in the previous layer and performs arithmetic sum of the preceding layer of features with individual neuron in the current layer to generate output as shown in fig. 5.

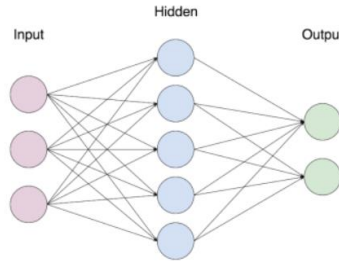


Fig. 5: Fully-connected layer

III. DIFFERENT MODELS OF CNN AVAILABLE FOR IMAGE CLASSIFICATION

VGGNet – Visual Geometry Group

Simonyan and Zisserman used deeper configuration of AlexNet [21], and they proposed it as VGGNet [14]. VGGNet comes in two different flavors, VGG16 and VGG19, where 16 and 19 are the total number of layers in each of them correspondingly. VGGNet was born out of the need to reduce the number of parameters in the CONV layers and improve on training time [27]. The variations of VGGNet are discussed below:

A. VGG-16

VGG16, a flavor of VGG net, is a Convolution Neural Net (CNN) architecture was used to win ILSVR(ImageNet) contest in 2014[14]. The most unique thing about this VGG16 is that instead of taking a large number of hyper-parameters they concentrated on having convolution layers with 3x3 filter and with a stride 1 and continually used maxpool layer of 2x2 filter of stride 2 and same padding. It follows this procedure of max pool and convolution layers steadily throughout the complete architecture. In the end it has 2 fully connected layers (FC) followed by a SoftMax for the output. VGG16[14], the 16 here refers to it has 16 layers that have weights.

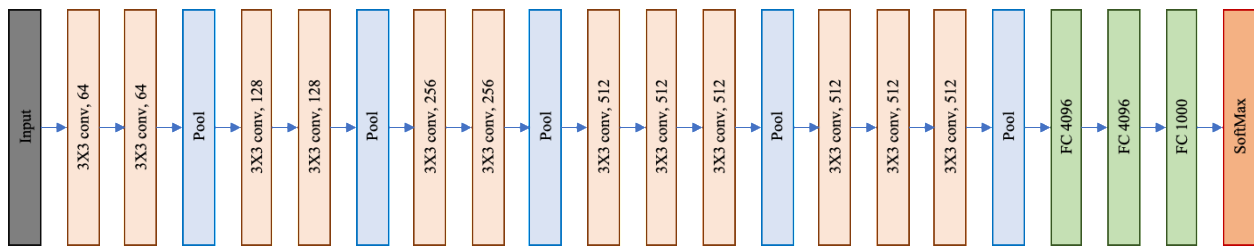


Fig. 6: VGG16 Architecture.

The input to the network is image of dimensions (224, 224, 3) [22]. The first two layers have 64 channels of 3*3 filter size and same padding and then a max pool layer of stride (2, 2). Again, two layers which have convolution layers of 128 and filter size (3, 3), a max pooling layer of stride (2, 2). Then there are 2 convolution layers of filter size (3, 3) and 256 filter. There are 2 sets of 3 convolution layer and a max pool layer. Each have 512 filters of (3, 3) size with same padding. Followingly there are 3 fully connected layer [23], the first layer takes input from the last feature vector and outputs a (1, 4096) vector, second layer also outputs a vector of size (1, 4096) but the third layer outputs 1000 channels for 1000 classes of ILSVRC challenge, then after the output of 3rd fully connected layer is passed to softmax layer in order to normalize the classification vector [24].

B. VGG 19

The visual geometry group network (VGGNet) is a deep neural network with a multilayered operation. The VGGNet is based on the CNN model and is applied on the ImageNet dataset [21]. VGG-19 is useful due to its simplicity as 3×3 convolutional layers are mounted on the top to increase with depth level. To reduce the volume size, max pooling layers were used as a handler in VGG-19[14]. Two FC layers were used with 4096 neurons. In the training phase, convolutional layers were used for the feature extraction and max pooling layers associated with some of the convolutional layers to reduce the feature dimensionality. In the first convolutional layer, 64 kernels (3×3 filter size) were applied for feature extraction from the input images. Fully connected layers were used to prepare the feature vector.

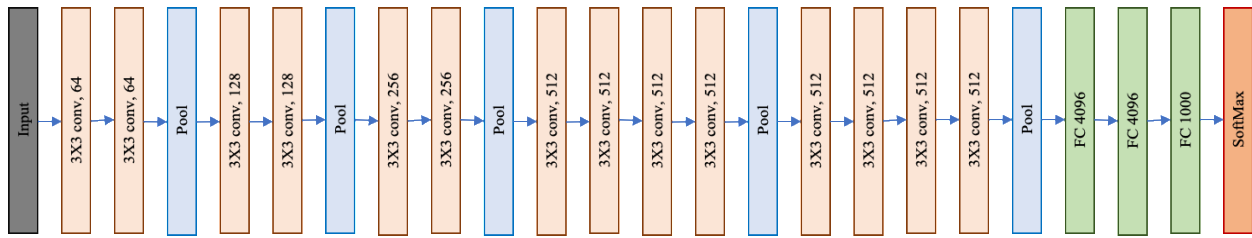


Fig. 7: VGG19 Architecture.

C. ResNet-50

From the past two CNNs, it is seen nothing but a rising number of layers in the design, and for achieving better performance of the design. But “with the network depth rising/increasing, accuracy grows saturated and then reduces rapidly.” The guys from Microsoft Research focused this problem with ResNet — by means of skip connections (a.k.a. shortcut connections, residuals), whilst building deeper models. ResNet architecture is one of the initial adopters of batch normalization [25].

The deeper CNN stacked up with more layers suffers from vanishing gradient problem. To solve this problem, pre-trained shallower model is used with additional layers to perform identity mapping. So that the performance of deeper network and the shallower network should be similar. The deep residual learning framework [26] is introduced as a solution to the degradation problem. Hence, included residual mapping ($H(x) = F(x) + x$) instead of desired underlying mapping ($H(x)$) into the network and named the model as ResNet [26].

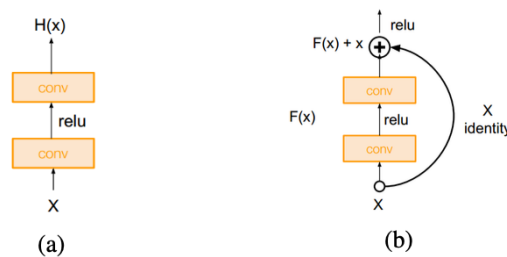


Fig. 8: (a) Plain layer (b) Residual block [17]

ResNet architecture consists of stacked residual blocks of 3×3 convolutional layers. They have periodically doubled the number of filters and used a stride of 2. Fig. 8a and 8b shows a plain layer and residual block. The architecture of Resnet50 is as shown in the fig. 9. The input to the network is image of dimensions (224, 224, 3) and this image is given to four stages as shown in the figure. Then the image undergoes average pooling, this output is subjected to fully connected that outputs 1000 channels for 1000 classes of ILSVRC net.

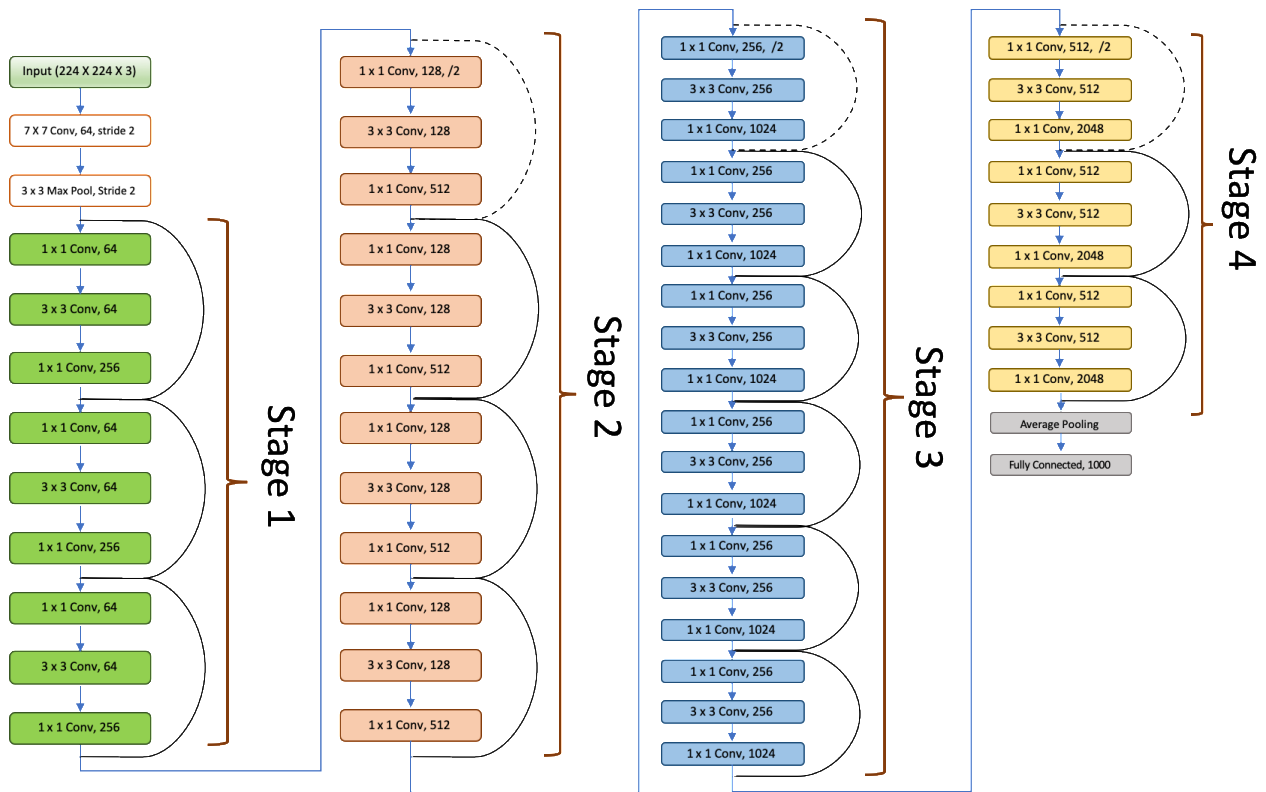


Fig. 9: RESNET50 Architecture.

IV.EXPERIMENTS AND RESULTS:

Keras, a deep learning API written in Python (latest version 3.7.6 is used) and TensorFlow 2.0 is used to realize the pre-trained models of Vgg16, Vgg19 and Resnet50. These models have been pre-trained with ImageNet dataset that has tens of millions of human annotated images. Different images belonging to different classes have been chosen from internet for image classification. We have chosen 50 different varieties of images like animals, birds, fruits, vegetables, object images like guitar, umbrella etc., for compiling the analysis of prediction using different nets.. Different models have given different prediction values for a given image.

For example, when Fig. 10 was subjected for image prediction, the network predicts it as an African Elephant by Resnet50, Vgg16 and Vgg19 with the prediction accuracy as 0.7855, 0.6805 and 0.7147 respectively. Similarly, Fig. 11 is predicted as an Umbrella with 0.9869, 0.7588, 0.7213 prediction accuracy by Resnet50, Vgg16 and Vgg19 nets respectively. Fig. 12 was subjected to the Resnet50, Vgg16 and Vgg19 nets, it is predicted as Pineapple with the prediction accuracy of 0.9998, 0.9958 and 0.9558 respectively.



Fig. 10: Elephant Image



Fig. 11: Umbrella



Fig. 12: Pineapple

V.DISCUSSIONS:

Image classification using three different architectures:

ResNet50, Vgg16 and Vgg19 architectures are trained based on ImageNet dataset to identify and classify images. A random set of 50 images are chosen for our analysis and these images are given to ResNet50, Vgg16 and Vgg19 architectures. Different models gives different prediction values for a given image. For every image, precision values are noted and compared with the other nets. For every image, the distribution of different precision values for Resnet50, Vgg16 and Vgg19 is plotted as in fig. 13, fig. 14 and fig.15 respectively. The x-axis in the chart represent the input set of 50 images and the y-axis represent the prediction value of the images.

Image recognition using ResNet50 architecture

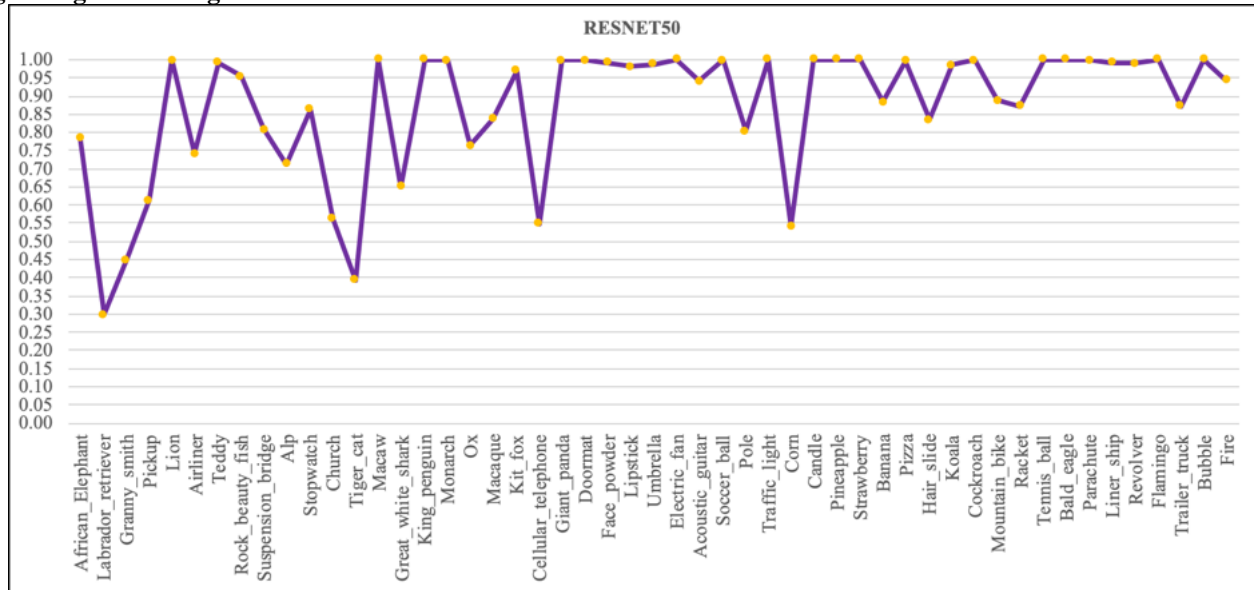


Fig. 13: Distribution of images prediction values on Resnet50 architecture

Image recognition using VGG16 architecture

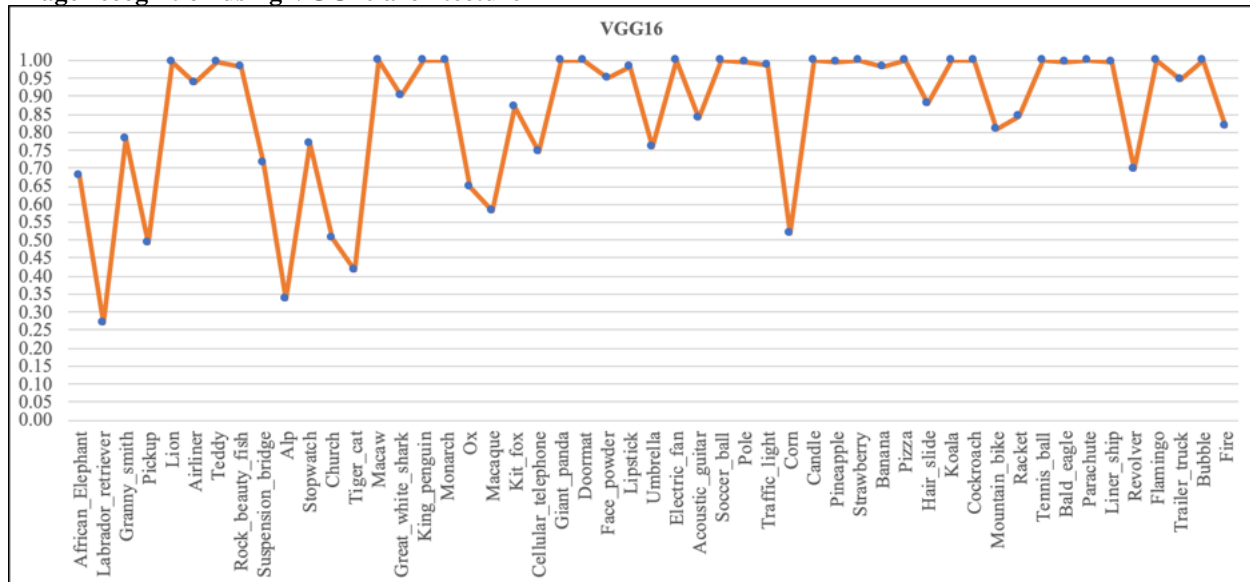


Fig. 14: Distribution of images prediction values on Vgg16 architecture.

Image recognition using VGG19 architecture

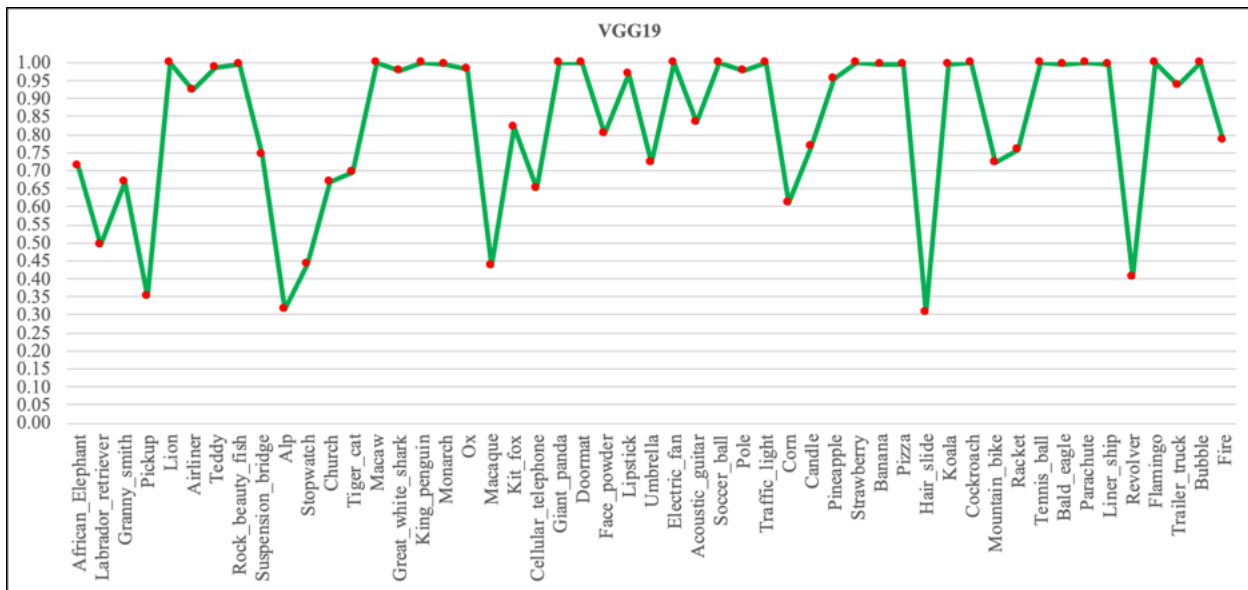


Fig. 15: Distribution of images prediction values on Vgg19 architecture.

Comparison of images prediction values – RESNET50, VGG16, VGG19

The fig. 14 shows the comparison of predicted values for every image. Each image is subjected to Resnet50, Vgg16 and Vgg19 architectures for image identification. Depending on the architecture and based on the dataset, CNN predicts the particular image and gives the result with prediction accuracy. Each architecture varies in the value of prediction of the same image. This variation for different precision values for all the images is shown in the fig. 17 for three different architectures.

Images	Image Names	Prediction Accuracy of 50 images			Deviation in prediction accuracy			Images	Image Names	Prediction Accuracy of 50 images			Deviation in prediction accuracy		
		RESNET50	VGG16	VGG19	ResNet50 ~ Vgg16	ResNet50 ~ Vgg19	Vgg16 ~ Vgg19			RESNET50	VGG16	VGG19	ResNet50 ~ Vgg16	ResNet50 ~ Vgg19	Vgg16 ~ Vgg19
	African Elephant	0.7870	0.6806	0.7148	0.1064	0.0722	0.0342		Umbrella	0.9869	0.7588	0.7213	0.2281	0.2656	0.0375
	Labrador Retriever	0.2974	0.2691	0.4952	0.0282	0.1978	0.2260		Electric Fan	0.9999	0.9985	0.9997	0.0014	0.0002	0.0012
	Granny Smith	0.4485	0.7819	0.6689	0.3334	0.2204	0.1130		Acoustic Guitar	0.9415	0.8381	0.8352	0.1034	0.1063	0.0029
	Pickup	0.6136	0.4900	0.3529	0.1236	0.2607	0.1371		Soccer Ball	0.9992	0.99999	0.99999	0.0008	0.0008	0.0000
	Lion	0.9994	0.9973	0.9997	0.0021	0.0003	0.0024		Pole	0.8028	0.9961	0.9784	0.1933	0.1756	0.0177
	Airliner	0.7416	0.9369	0.9258	0.1953	0.1842	0.0111		Traffic Light	0.9999	0.9882	0.9997	0.0117	0.0002	0.0115
	Teddy	0.9925	0.9937	0.9874	0.0012	0.0051	0.0063		Corn	0.5427	0.5167	0.6123	0.0260	0.0696	0.0956
	Rock Beauty fish	0.9543	0.9806	0.9969	0.0264	0.0427	0.0163		Candle	0.9999	0.9978	0.7663	0.0021	0.2336	0.2315
	Suspension Bridge	0.8088	0.7148	0.7456	0.0940	0.0632	0.0307		Pineapple	0.9998	0.9958	0.9558	0.0040	0.0440	0.0400
	Alp	0.7128	0.3373	0.3170	0.3755	0.3958	0.0203		Strawberry	0.9999	0.9997	0.9995	0.0002	0.0004	0.0002
	Stopwatch	0.8639	0.7688	0.4392	0.0951	0.4247	0.3296		Banana	0.8843	0.9814	0.9934	0.0971	0.1091	0.0120
	Church	0.5616	0.5041	0.6692	0.0574	0.1077	0.1651		Pizza	0.9968	0.9994	0.9934	0.0026	0.0034	0.0060
	Tiger Cat	0.3928	0.4184	0.6968	0.0256	0.3040	0.2784		Hair Slide	0.8356	0.8794	0.3052	0.0438	0.5304	0.5742
	Macaw	0.9999	1	0.9999	0.0001	0.0000	0.0001		Koala	0.9847	0.9988	0.9964	0.0141	0.0117	0.0024
	Great White Shark	0.651	0.901	0.9767	0.2500	0.3257	0.0757		Cockroach	0.9983	0.9995	0.9996	0.0012	0.0013	0.0001
	King Penguin	0.9999	1	1	0.0001	0.0001	0.0000		Mountain Bike	0.8878	0.8104	0.7217	0.0774	0.1661	0.0887
	Monarch	0.9978	0.9992	0.9943	0.0014	0.0035	0.0049		Racket	0.8728	0.844	0.758	0.0288	0.1148	0.0860
	Ox	0.7632	0.6467	0.9799	0.1165	0.2167	0.3332		Tennis Ball	1	0.9999	0.9999	0.0001	0.0001	0.0000
	Macaque	0.8377	0.5799	0.4356	0.2578	0.4021	0.1443		Bald Eagle	0.9999	0.9958	0.9966	0.0041	0.0033	0.0008
	Kit fox	0.9692	0.8726	0.8213	0.0966	0.1479	0.0513		Parachute	0.998	0.9999	0.9996	0.0019	0.0016	0.0003
	Cellular Telephone	0.5492	0.746	0.6529	0.1968	0.1037	0.0931		Liner Ship	0.9929	0.9971	0.9952	0.0042	0.0023	0.0019
	Giant Panda	0.999	0.999	0.999	0.0000	0.0000	0.0000		Revolver	0.9899	0.6983	0.4067	0.2916	0.5832	0.2916
	Doormat	0.999	0.999	0.999	0.0000	0.0000	0.0000		Flamingo	0.9999	0.9999	0.9999	0.0000	0.0000	0.0000
	Face Powder	0.991	0.951	0.805	0.0400	0.1860	0.1460		Trailer Truck	0.8721	0.9472	0.9355	0.0751	0.0634	0.0117
	Lipstick	0.981	0.9798	0.97021	0.0012	0.0108	0.0096		Bubble	1	1	0.9999	0.0000	0.0001	0.0001

Fig. 16: Distribution of images with their prediction values

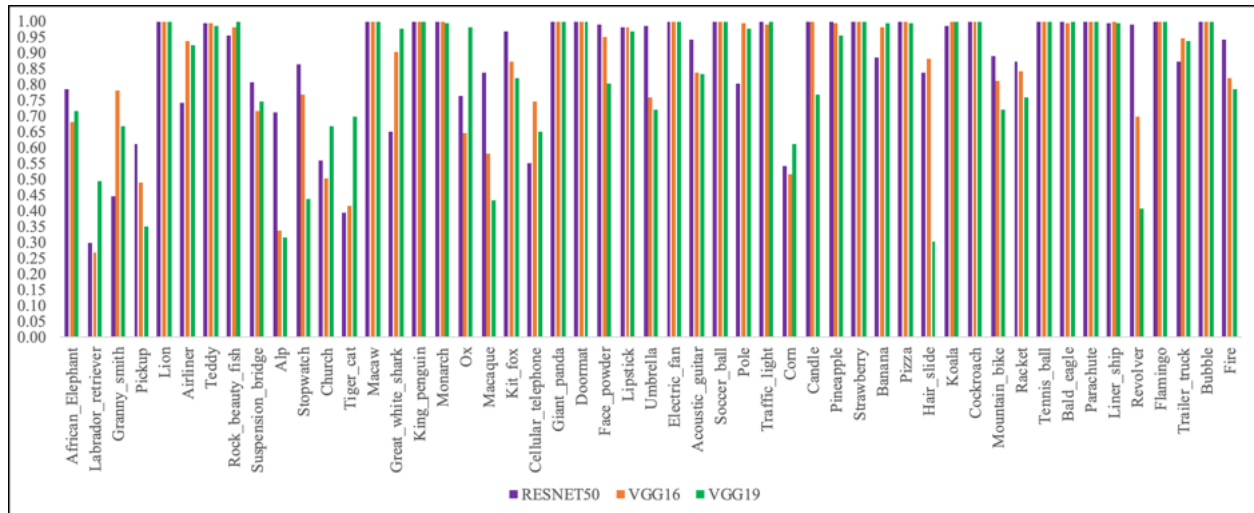


Fig. 17: Comparison of image prediction values – RESNET50, VGG16, VGG19

The CNN is realized using different architectures for different images and their precision values are analyzed. Based on the work, ResNet50 gives the highest accuracy for the given image when compared to Vgg16 and Vgg19. The network depth of the ResNets are larger and achieves better accuracy while being computationally more efficient than VGGNet. The architecture of Resnet50 is similar to the VGGNet consisting mostly of 3X3 filters except the shortcut connection is inserted in residual network. This residual network overcomes the vanishing gradient problem or degradation problem and hence giving a better accuracy compared with Vgg16 and Vgg19. Based on the pre-trained architectures a particular image is recognized. This precision accuracy depends on how well the networks are trained in the particular data set. The graph in fig. 18 shows the highest accuracy using ResNet50 architecture for all the images.

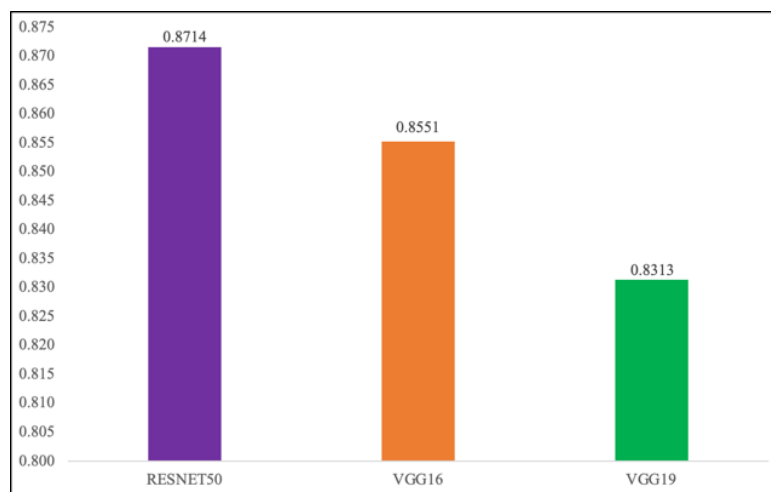


Fig. 18: Average of the predicted values of all the images

VI.CONCLUSION

In this paper, a deep learning convolutional neural network based on Keras and Tensorflow is deployed using python for image classification. This paper analyzed the prediction accuracy of three different Convolutional Neural Network (CNN) on most popular ImageNet dataset. The main purpose of this paper is to find out the accuracy of the different networks on same dataset and evaluating the consistency of prediction by each of this CNN. We have presented a detailed prediction analysis for comparing the networks' performance for different images. The images are subjected to Vgg16, Vgg19 and Resnet50 networks and the prediction values are evaluated. The analysis shows that ResNet50 able to recognize images with better precision compared to Vgg16 and Vgg19. The future enhancement will focus on other high-level nets. Future works will be dedicated to investigating the other pre-trained models for the classification task.

VII.REFERENCES:

- [1] Sachchidanand Singh, Nirmala Singh "Object Classification to Analyze Medical Imaging Data using Deep Learning", International Conference on Innovations in information Embedded and Communication Systems (ICIIECS), ISBN 978-1-5090-3295-2, pp. 1-4, 2017

- [2] Li Deng and Dong Yu "Deep Learning: methods and applications" by Microsoft research [Online] available at: <http://research.microsoft.com/pubs/209355/NOW-Book-Revised- Feb2014-online.pdf>
- [3] Wang, Y., & Wu, Y. "Scene Classification with Deep Convolutional Neural Networks."
- [4] Zhou, B., Khosla, A., Lapedriza, A., Oliva, A., & Torralba, A. (2014) "Object detectors emerge in deep scene cnns."
- [5] Srinivas, S., Sarvadevabhatla, R. K., Mopuri, K. R., Prabhu, N., Kruthiventi, S. S., & Babu, R. V. (2016) "A taxonomy of deep convolutional neural nets for computer vision."
- [6] ANIMAL DETECTION USING DEEP LEARNING ALGORITHM N. BANUPRIYA, S. SARANYA, RASHMI SWAMINATHAN, SANCHITHAA HARIKUMAR, SUKITHA PALANISAMY
- [7] An analysis of Convolutional Neural Networks for Image Classification. International Conference on Computational Intelligence and Data Science (ICCIDS 2018) by Neha Sharma, Vibhor Jain, Anju Mishra.
- [8] L. Liu, P. Fieguth, G. Zhao, M. Pietikäinen, and D. Hu, "Extended local binary patterns for face recognition," *Inf. Sci. (Ny)*, vol. 358–359, pp. 56–72, Sep. 2016.
- [9] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei, "ImageNet: A Large-Scale Hierarchical Image Database," in *CVPR09*, 2009.
- [10] B. C. Russell, A. Torralba, K. P. Murphy, and W. T. Freeman, "Labelme: A database and web-based tool for image annotation," *International Journal of Computer Vision*, vol. 77, no. 1, pp. 157–173, May 2008. [Online]. Available: <https://doi.org/10.1007/s11263-007-0090-8>
- [11] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," in *Advances in Neural Information Processing Systems 25*, F. Pereira, C. J. C. Burges, L. Bottou, and K. Q. Weinberger, Eds. Curran Associates, Inc., 2012, pp. 1097–1105. [Online]. Available: <http://papers.nips.cc/paper/4824-imagenet-classification-with-deep-convolutional-neural-networks>.
- [12] O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, Z. Huang, A. Karpathy, A. Khosla, M. Bernstein, A. C. Berg, and L. Fei-Fei, "Imagenet large scale visual recognition challenge," *Int. J. Comput. Vision*, vol. 115, no. 3, pp. 211–252, Dec. 2015. [Online]. Available: <http://dx.doi.org/10.1007/s11263-015-0816-y>
- [13] M. D. Zeiler and R. Fergus, "Visualizing and understanding convolutional networks," in *Computer Vision – ECCV 2014*, D. Fleet, T. Pajdla, B. Schiele, and T. Tuytelaars, Eds. Cham: Springer International Publishing, 2014, pp. 818–833.
- [14] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," *CoRR*, vol. abs/1409.1556, 2014. [Online]. Available: <http://arxiv.org/abs/1409.1556>
- [15] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, and A. Rabinovich, "Going deeper with convolutions," in *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2015.
- [16] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2016.
- [17] G. Huang, Z. Liu, and K. Q. Weinberger, "Densely connected convolutional networks," *CoRR*, vol. abs/1608.06993, 2016. [Online]. Available: <http://arxiv.org/abs/1608.06993>
- [18] S. Sabour, N. Frosst, and G. E. Hinton, "Dynamic routing between capsules," *CoRR*, vol. abs/1710.09829, 2017. [Online]. Available: <http://arxiv.org/abs/1710.09829>
- [19] J. Hu, L. Shen, and G. Sun, "Squeeze-and-excitation networks," *CoRR*, vol. abs/1709.01507, 2017. [Online]. Available: <http://arxiv.org/abs/1709.01507>
- [20] O. Russakovsky et al. Imagenet large scale visual recognition challenge. arXiv:1409.0575, 2014.
- [21] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," in *Advances in Neural Information Processing Systems 25*, F. Pereira, C. J. C. Burges, L. Bottou, and K. Q. Weinberger, Eds. Curran Associates, Inc., 2012, pp. 1097–1105. [Online]. Available: <http://papers.nips.cc/paper/4824-imagenet-classification-with-deep-convolutional-neural-networks>. Pdf
- [22] J. Yosinski, J. Clune, Y. Bengio, and H. Lipson, "How transferable are features in deep neural networks," in *Advances in neural information processing systems*, 2014, pp. 3320–3328.
- [23] P. K. Sonawane and S. Shelke, "Handwritten Devanagari Character Classification using Deep Learning," in *2018 International Conference on Information, Communication, Engineering and Technology (ICICET)*, 2018, pp. 1–4.
- [24] S. Lu, Z. Lu, and Y.-D. Zhang, "Pathological brain detection based on AlexNet and transfer learning," *J. Comput. Sci.*, vol. 30, pp. 41–47, 2019.
- [25] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," *CoRR*, vol. abs/1512.03385, 2015.
- [26] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2016.
- [27] <https://towardsdatascience.com/the-w3h-of-alexnet-vggnet-resnet-and-inception-7baaecccc96>
- [28] Image Classification using Convolutional Neural Networks by Muthukrishnan Ramprasath and M. Vijay Anand International Journal of Pure and Applied Mathematics Volume 119 No. 17 2018.
- [29] Image Classification Using Convolutional Neural Networks by Deepika Jaswal, Sowmya V., K. P. Soman - International Journal of Advancements in Research & Technology, Volume 3, Issue 6, June-2014.
- [30] Image Classification with Deep Learning and Comparison between Different Convolutional Neural Network Structures using Tensorflow and Keras by Karan Chauhan, Shrwan Ram - International Journal of Advance Engineering and Research Development, Volume 5, Issue 02, February -2018.
- [31] Advancements in Image Classification using Convolutional Neural Network by Farhana Sultana, Abu Sufian, Paramartha Dutta - 2018 Fourth International Conference on Research in Computational Intelligence and Communication Networks
- [32] Benchmark Analysis of Popular ImageNet Classification Deep CNN Architectures by Mustafa Alghali Elsaid Muhammed.
- [33] A Comprehensive Study of ImageNet Pre-Training for Historical Document Image Analysis by Linda Studer, Michele Alberti, Vinaychandran Pondenkandath, Pinar Goktepe, Thomas Kolonko, Andreas Fischer, Marcus Liwicki, Rolf Ingold - 2019 International Conference on Document Analysis and Recognition (ICDAR).
- [34] Automated brain image classification based on VGG-16 and transfer learning by Taranjit Kaur, Tapan Kumar Gandhi - 2019 International Conference on Information Technology (ICIT).
- [35] Compressed Residual-VGG16 CNN Model for Big Data Places Image Recognition by Hussam Qassim, Abhishek Verma, David Feinzimer.
- [36] VERY DEEP CONVOLUTIONAL NETWORKS FOR LARGE-SCALE IMAGE RECOGNITION by Karen Simonyan & Andrew Zisserman - Published as a conference paper at ICLR 2015
- [37] A Switch State Recognition Method based on Improved VGG19 network by Yongjun Xia, Min Cai, Chuankun Ni, Chengzhi Wang, Shiping E, Hengxuan Li - 2019 IEEE 4th Advanced Information Technology, Electronic and Automation Control Conference (IAEAC 2019).
- [38] Improving a 3-D Convolutional Neural Network Model Reinvented from VGG16 with Batch Normalization by Nontawat Pattanajak and Hossein Malekmohamadi - 2019 IEEE SmartWorld, Ubiquitous Intelligence & Computing, Advanced & Trusted Computing, Scalable Computing & Communications, Cloud & Big Data Computing, Internet of People and Smart City Innovation.
- [39] Face Recognition Using Light-Convolutional Neural Networks Based on Modified VGG16 Model by Anugrah Bintang Perdana, Adhi Prahara.
- [40] Real-Time Object Detection on 640x480 Image With VGG16+SSD by Hyeon-Ju Kang - 2019 International Conference on Field-Programmable Technology (ICFPT).