A Comparative study of Convolutional Neural Network Models

Group:05
[**Annon,Nihal Abedin**(18-37301-1),**Suparan Sharma**(18-37275-1),**Snigdho Dip Howlader**(18-36177-1),**Abir-Al-Arafat**(18-36138-1)]

[a]*Department of Computer Sciences, American International University-Bangladesh*

Abstract

In Image Classification, there are some very popular datasets that are used across research, industry, and hackathons. In this article, we will cover the top 5 pre-trained models for Image Classification are widely used in the industry as well. The individual models verily used for Computer Vision, Object detection using Deep learning and One of the most effective models for deep learning is the Convolutional Neural Network (CNN). these models can be explained in much more detail, but we have limited the article to give an overview of their architecture and implement it on a dataset.

1. Introduction

The human brain is quit good at recognizing and distinguish things in images. For example, given a picture of a cat and a dog, we can differentiate the two in nanoseconds and our brain recognizes the difference. It is as near to Artificial Intelligence as we can go if a computer duplicates the behavior. As a result, Computer Vision appears to replicate the human vision system – and there have been numerous milestones that have broken the barriers in this regard .

Moreover, nowadays machines can easily distinguish between different images, detect objects and faces, and even generate images of people who don’t exist! In this report, we are focusing about the models used for Computer Vision for Detectiong an Object using deep learning. The following are some of the prominent ones: VGG-16, ALexNet, ResNet50, Inceptionv3, EfficientNet.

Questions that are going to be discussed in these report are:

1. How the particular model works?
2. What are the Architectures of these models?
3. What are the similarities of these models?
4. What are the advantages using the particular model?

Despite the great achievements in image classification, deep learning has following challenges:

- Number of inputs to be considered and finding non-contributing columns;
- Number of hidden layers;
- Activation Functions;
- Optimization Algorithms;
- Decay function;
- Number of epochs ,

later we are going to discuss some of these challenges in this report with poper diagrams
And theoretical explanation.

2. Literature Review

Millions of pictures are created every day. Every image must be classified in this way so that they may be found more quickly and at a faster rate. Humans are
are capable of classifying pictures more quickly than computers. A basic categorization system consists of a camera mounted high above the area of interest,which captures and processes pictures. As a result, they can happen more readily and at a faster rate. Humans Are capable of classifying pictures more quickly than computers. A basic categorization system consists of a camera mounted high above the area of interest, which captures and processes pictures.Classification is a procedure to classify images into several categories,
based on their similarities. We can easily understand or analyses our surroundings by classifying the images. In the classification system user deal with a database and that database contains some patterns or images which are predefined or which are going to be classified. Image classification always is a critical but an important task for many Applications.

3. Proposed Method

[i] VGG-16: The VGG-16 is one of the most popular pre-trained models for image classification. Introduced in the famous ILSVRC 2014 Conference, it was and remains THE model to beat even today. Developed at the Visual Graphics Group at the University of Oxford, VGG-16 beat the then standard of AlexNet and was quickly adopted by researchers and the industry for their image Classification Tasks.
Here is the architecture of VGG-16:

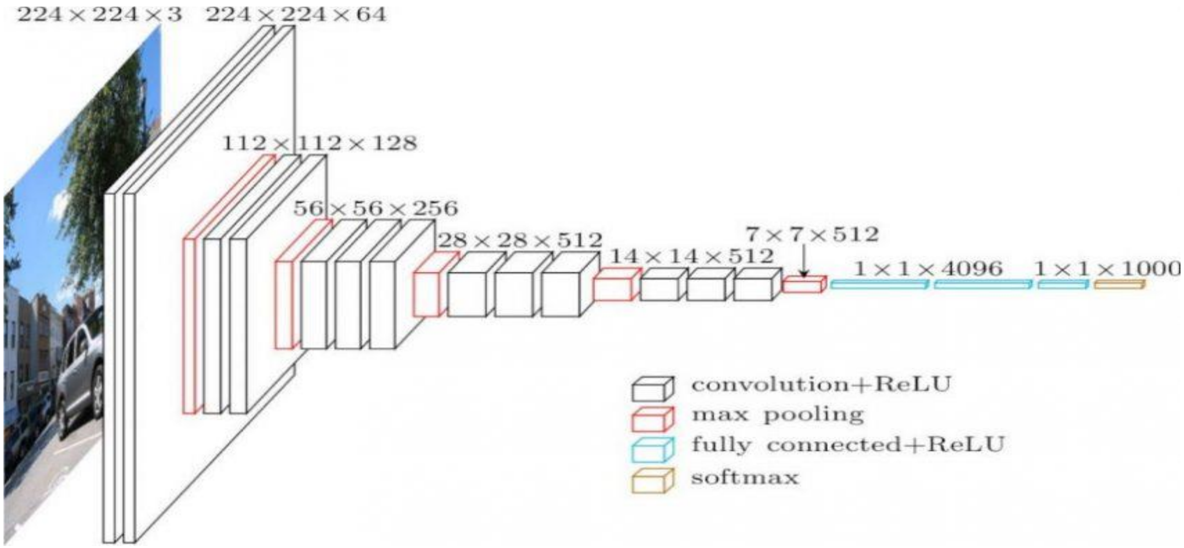


Fig: 7 Architecture of VGG-16

The precise structure of the VGG-16 network shown in Figure. 7. is as follows:

- The first and second convolutional layers are comprised of 64 feature kernel filters and size of the filter is 3×3. As input image (RGB image with depth 3) passed into first and second convolutional layer, dimensions changes to 224x224x64. Then the resulting output is passed to max pooling layer with a stride of 2.

- The third and fourth convolutional layers are of 124 feature kernel filters and size of filter is 3×3. These two layers are followed by a max pooling layer with stride 2 and the resulting output will be reduced to 56x56x128.

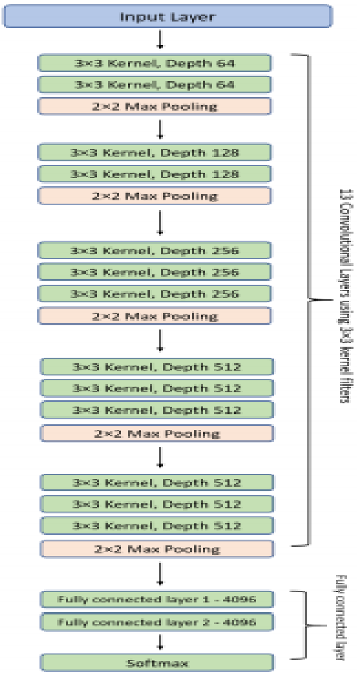


Fig. 7. VGG-16 model architecture – 13 convolutional layers and 2 Fully connected layers and 1 SoftMax classifier

- The fifth, sixth and seventh layers are convolutional layers with kernel size 3×3. All three use 256 feature maps. These layers are followed by a max pooling layer with stride 2.

- Eighth to thirteen are two sets of convolutional layers with kernel size 3×3. All these sets of convolutional layers have 512 kernel filters. These layers are followed by max pooling layer with stride of 1

- Fourteen and fifteen layers are fully connected hidden layers of 4096 units followed by a softmax output layer (Sixteenth layer) of 1000 units.

Now, let us about the dataset of Cat and Dog Images. The original training dataset on Kaggle has 25000 images of cats and dogs and the test dataset has 10000 unlabelled images. Since our purpose is only to understand these models, We took 5000 data images of cats and dogs out of 25000 images for training and 2000 images for validation.Now using these dataset running with basic CNN model –Model accuracy and loss, wee get the following graphs:

Fig:1.1 Training and validation accuracy after using CNN

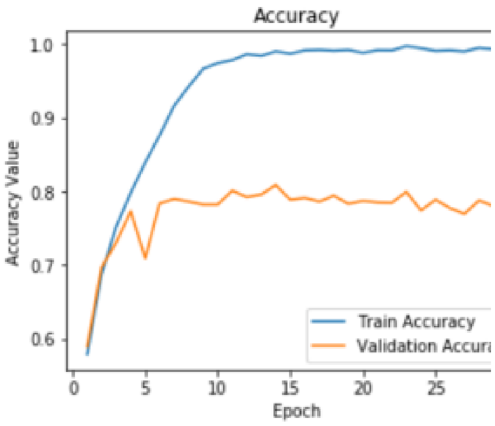
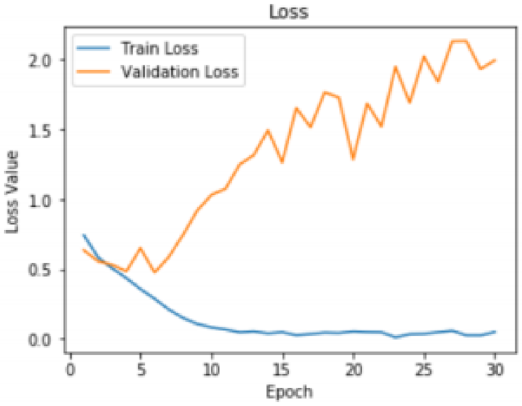
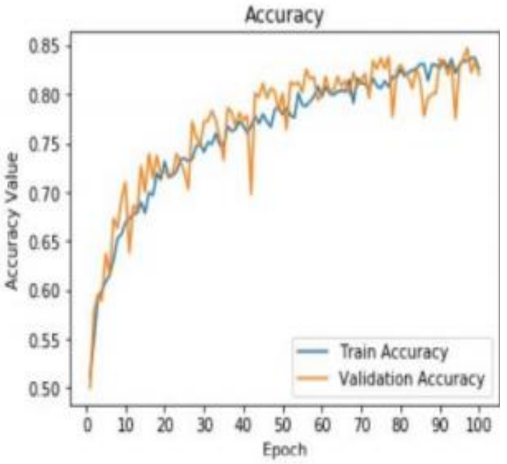
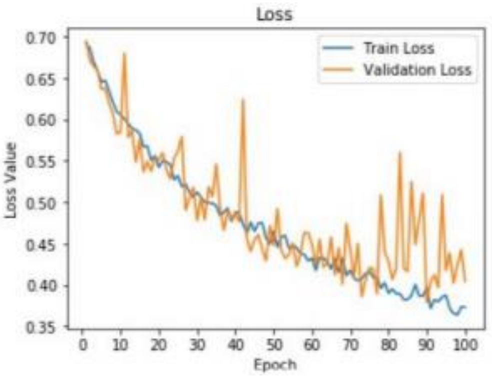


Fig:1.2 Training and validation loss after using CNN



And after running the Dataset running with CNN with image augmentation , we can have the graphs of Accuracy and Loss.





As you can see, we were able to achieve a validation Accuracy(85% approx.) with just 100 epochs and without any major changes to the model. This is where we realize how powerful transfer learning is and how useful pre-trained models for image classification can be. A caveat here though – VGG16 takes up a long time to train compared to other models and this can be a disadvantage when we are dealing with huge datasets.

[ii] ALexNet : AlexNet is a convolutional neural network that has had a significant influence on machine learning, particularly in the application of deep learning to machine vision. Convolutions, max pooling, dropout, data augmentation, ReLU activations, and SGD with momentum were all part of it. After each convolutional and fully-connected layer, it added ReLU activations.

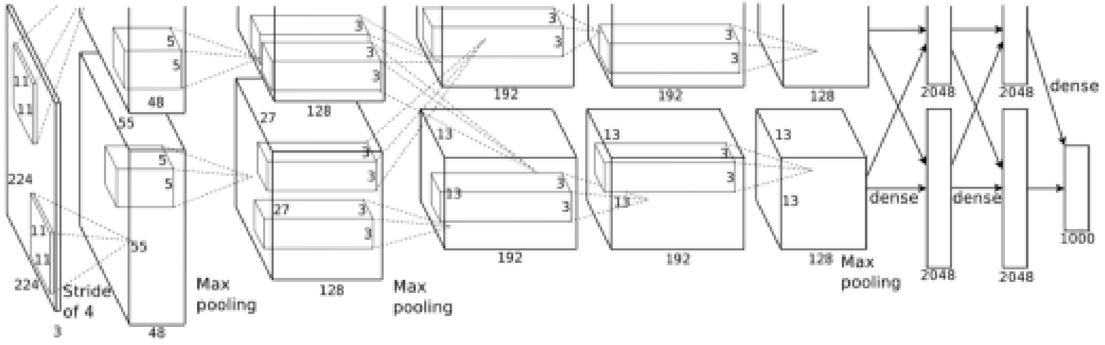


Fig: Illustration of AlexNet's architecture.

Eight layers make up the architecture: five convolutional layers and three fully linked layers. But that isn't the only thing that distinguishes AlexNet. these are some of the features used that are new approaches to convolutional neural networks:

- **ReLU Nonlinearity:** AlexNet uses Rectified Linear Units (ReLU) instead of the tanh function, which was standard at the time. ReLU's advantage is in training time; a CNN using ReLU was able to reach a 25% error on the CIFAR-10 dataset six times faster than a CNN using tanh.
- **Multiple GPUs:** Back in the day, GPUs were still rolling around with 3 gigabytes of memory (nowadays those kinds of memory would be rookie

numbers). This was especially bad because the training set had 1.2 million images. AlexNet allows for multi-GPU training by putting half of the model's neurons on one GPU and the other half on another GPU. Not only does this mean that a bigger model can be trained, but it also cuts down on the training time

- **Overlapping Pooling:** CNNs traditionally "pool" outputs of neighboring groups of neurons with no overlapping. However, when the authors introduced overlap, they saw a reduction in error by about 0.5% and found that models with overlapping pooling generally find it harder to overfit
- **The Overfitting Problem:** AlexNet had 60 million parameters, a major issue in terms of overfitting. Two methods were employed to reduce overfitting:
  - **Data Augmentation:** The authors used label-preserving transformation to make their data more varied. Specifically, they generated image translations and horizontal reflections, which increased the training set by a factor of 2048. They also performed Principle Component Analysis (PCA) on the RGB pixel values to change the intensities of RGB channels, which reduced the top-1 error rate by more than 1%.
  - **Dropout:**This technique consists of "turning off" neurons with a predetermined probability (e.g. 50%). This means that every iteration uses a different sample of the model's parameters, which forces each neuron to have more robust features that can be used with other random neurons. However, dropout also increases the training time needed for the model's convergence.

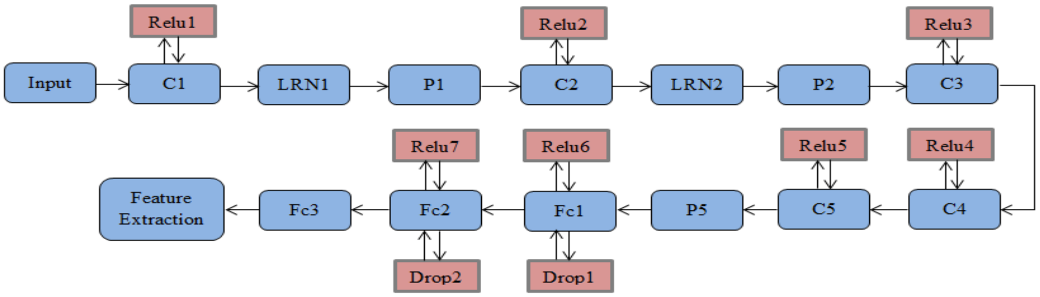
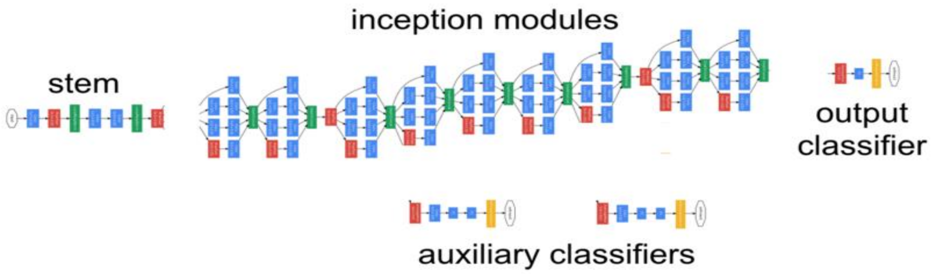


Fig.3 The flow chart of feature extraction based on CNN

Deep learninghas become a hot area of machine learning research, and ithas been made a significant breakthrough in image recognition, speech analysis, target detection and other fields.In this paper, we extract the last layer features of the deep convolutional neural network for scene classification by using the Alex-Net model in the deep learning framework, which makes the classification accuracy can be further improved, so a stronger generalization performance and higher efficiency scene image classification model is constructed.In future work, we will introduce the relationship between different scenes in the training set to reduce redundant information, so as to further improve the performance of scene classification.

[iii] Inceptionv3: The year 2014 has been iconic in terms of the development of really popular pre-trained models for Image Classification. While the above VGG-16 secured the 2nd rank in that years' ILSVRC, the 1st rank was secured by none other than Google – via its model GoogLeNet or Inception as it is now later called as.

The original paper proposed the Inceptionv1 Model. At only 7 million parameters, it was much smaller than the then prevalent models like VGG and AlexNet. Adding to it a lower error rate, you can see why it was a breakthrough model. Not only this, but the major innovation in this paper was also another breakthrough – the Inception Module Architecture.



The Inceptionv2 model was a major improvement on the Inceptionv1 model which increased the accuracy and further made the model less complex. In the same paper as Inceptionv2, the authors introduced the Inceptionv3 model with a few more improvements on v2.

The following are the major improvements included:

- Introduction of Batch Normalisation
- More factorization
- RMSProp Optimiser

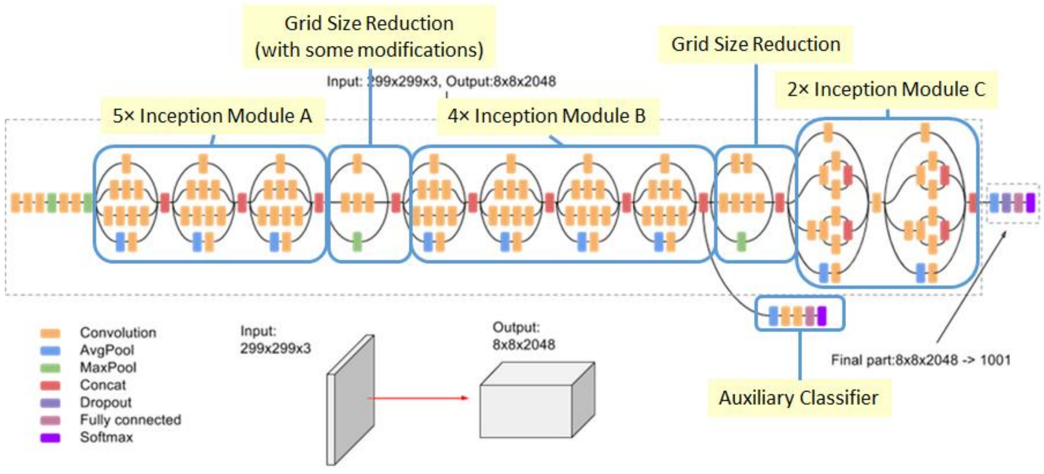


Fig: Inceptionv3[updates]

As you can see that the number of layers is 42, compared to VGG16's paltry 16 layers. Also, Inceptionv3 reduced the error rate to only 4.2%.

[*While it is not possible to provide an in-depth explanation of Inception in this article, you can go through this comprehensive article covering the Inception Model in detail:* ***Deep Learning in the Trenches: Understanding Inception Network from Scratch***]

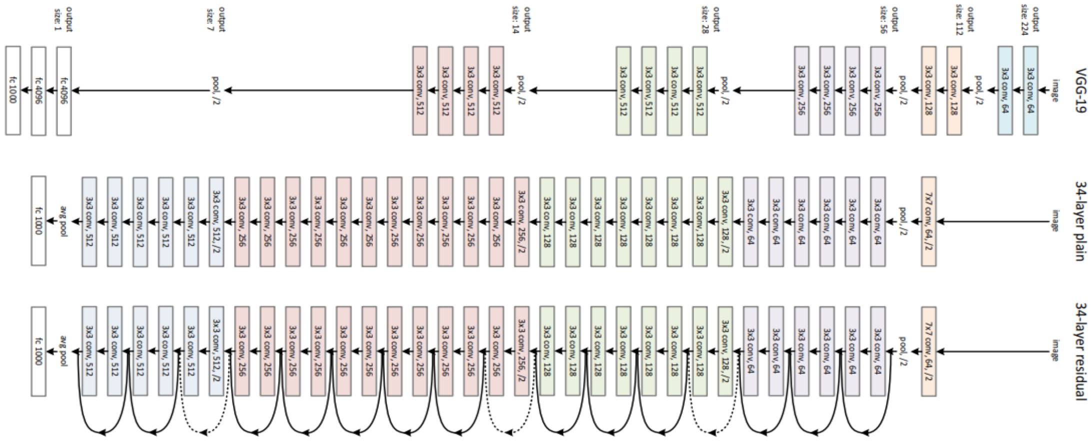
As a result, we can use the previous Dataset that we get 96% Validation accuracy in 100 epochs. Also note, how this model is much faster than VGG16. Each epoch is taking around only 1/4th the time that each epoch in VGG16.

[iv] ResNet50 : Just like Inceptionv3, ResNet50 is not the first model coming from the ResNet family. The original model was called the Residual net or ResNet and was another milestone in the CV domain back in 2015.

The main motivation behind this model was to avoid poor accuracy as the model went on to become deeper. Additionally, if you are familiar with Gradient Descent, you would have come across the Vanishing Gradient issue – the ResNet model aimed to tackle this issue as well.

Here is the architecture of the earliest variant: ResNet34(ResNet50 also follows a similar technique with just more layers)

This network uses a 34-layer plain network architecture inspired by VGG-19 in which then the shortcut connection is added. These shortcut connections then convert the architecture into residual network.



You can see that after starting off with a single Convolutional layer and Max Pooling, there are 4 similar layers with just varying filter sizes – all of them using 3 * 3 convolution operation. Also, after every 2 convolutions, we are bypassing/skipping the layer in-between. This is the main concept behind ResNet models. These skipped connections are called 'identity shortcut connections" and uses what is called residual blocks:

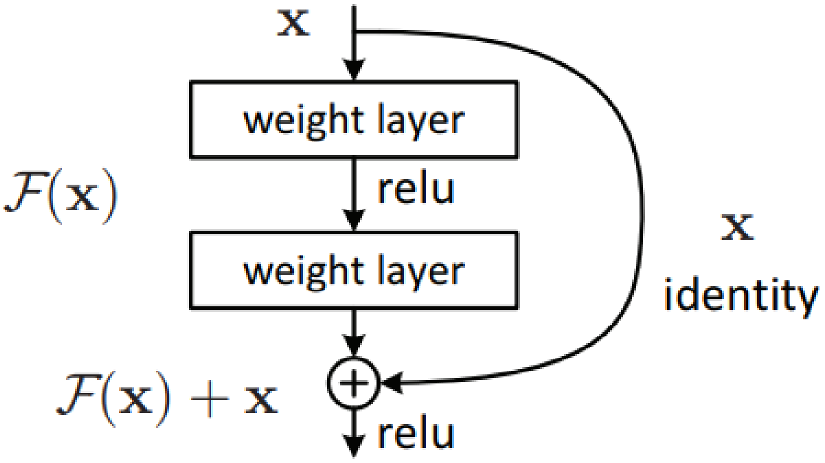

x
weight layer
$\mathcal{F}(\mathbf{x})$
relu
weight layer
x
identity
$\mathcal{F}(\mathbf{x}) + \mathbf{x}$
relu

Fig: residual blocks

In layman's words, the ResNet authors argue that fitting a residual mapping is considerably easier than fitting the real mapping, and that it should be used in all layers. Another important aspect to notice is that the ResNet creators believe that the more layers we stack, the better the model will perform.

Contrary to what we observed in Inception, this is basically identical to VGG16 in that it just stacks layers on top of each other. The underlying mapping is changed by ResNet.

Remarkably, ResNet not only has its own variants, but it also spawned a series of architectures based on ResNet. These include ResNeXt, ResNet as an Ensemble, etc. Additionally, the ResNet50 is among the most popular models out there and achieved a top-5 error rate of around 5%

The following is the link to the paper:[**Deep Residual Learning for Image Recognition**]

[v] EfficientNet: Finally, we get to the most recent model among these four that has made ripples in this field, and it is, of course, from Google. The authors of EfficientNet suggest a novel scaling approach called Compound Scaling. The long and short of it is that older models, such as ResNet, took the traditional technique of arbitrarily scaling the dimensions and layering on more and more layers.

However, the paper proposes that if we scale the dimensions by a fixed amount at the same time and do so uniformly, we achieve much better performance. The scaling coefficients can be in fact decided by the user.

Though this scaling technique can be used for any CNN-based model, the authors started off with their own baseline model called EfficientNetB0:

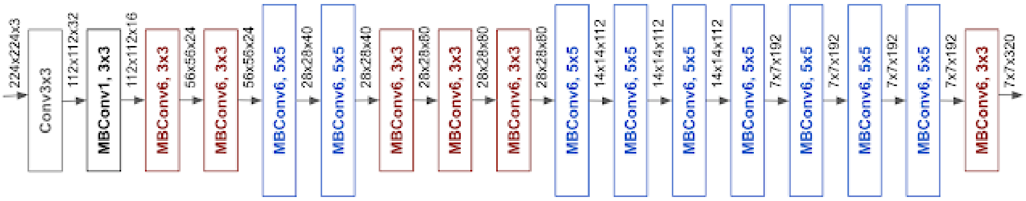


Fig: The EffecientNet-B0 general architecture

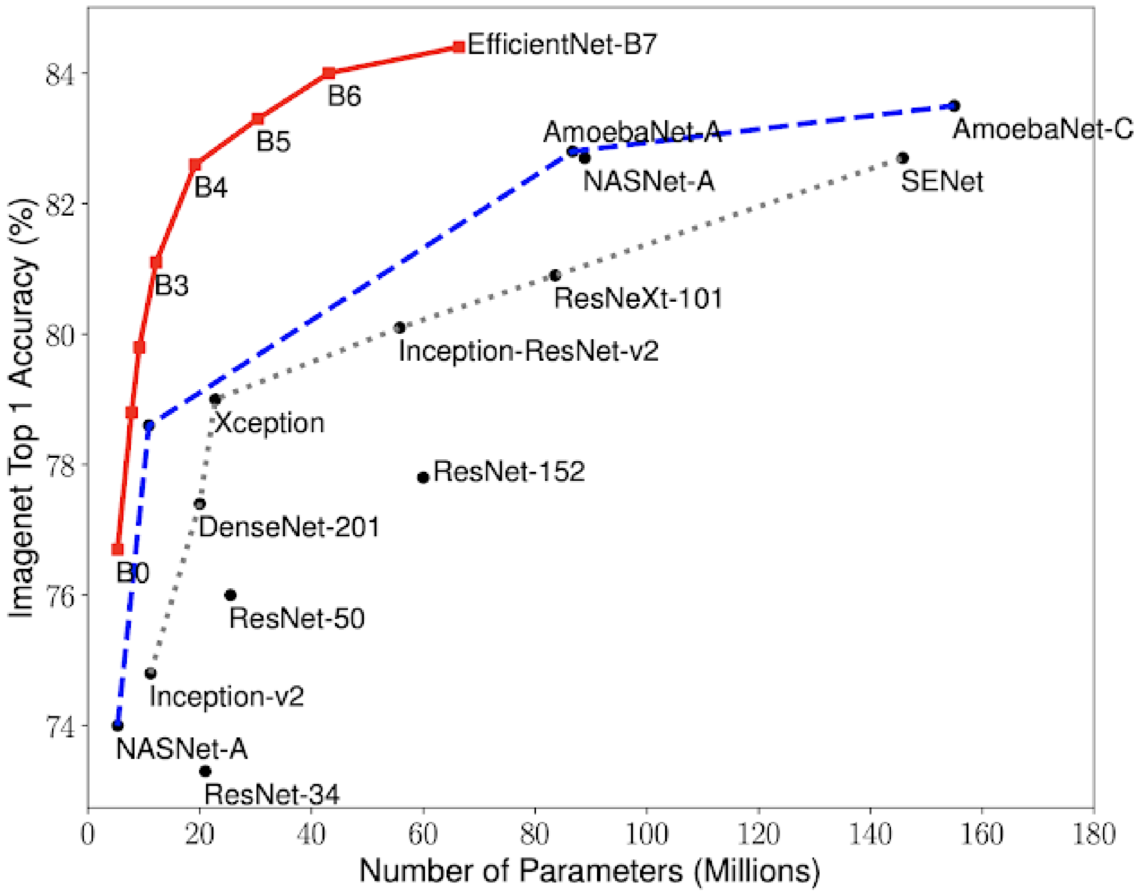
EfficientNet Performance

We have compared our EfficientNets with other existing CNNs on ImageNet. In general, the EfficientNet models achieve both higher accuracy and better efficiency over existing CNNs, reducing parameter size and FLOPS by an order of magnitude. For example, in the high-accuracy regime, our EfficientNet-B7 reaches state-of-the-art 84.4% top-1 / 97.1% top-5 accuracy on ImageNet, while being 8.4x smaller and 6.1x faster on CPU inference than the previous Gpipe. Compared with the widely used ResNet-50, our EfficientNet-B4 uses similar FLOPS, while improving the top-1 accuracy from 76.3% of ResNet-50 to 82.6% (+6.3%).

They also propose the Compound Scaling formula with the following scaling coefficients:

- Depth = 1.20
- Width = 1.10
- Resolution = 1.15

This formula is used to again build a family of EfficientNets – EfficientNetB0 to EfficientNetB7. The following is a simple graph showing the comparative performance of this family vis-a-vis other popular models:



Model Size vs. Accuracy Comparison. EfficientNet-B0 is the baseline network developed by AutoML MNAS, while Efficient-B1 to B7 are obtained by scaling up the baseline network. In particular, our EfficientNet-B7 achieves new state-of-the-art 84.4% top-1 / 97.1% top-5 accuracy, while being 8.4x smaller than the best existing CNN.

As you can see, even the baseline B0 model starts at a much higher accuracy, which only goes on increasing, and that too with fewer parameters. For instance, EfficientB0 has only 5.3 million parameters!

As a result – we got a whopping 98% accuracy on our validation set in only 100 epochs.

The following is the link to the paper: **EfficientNet: Rethinking Model Scaling for Convolutional Neural Networks**

4. Results

- ✓ VGG19 is able to correctly classify the the input image with a probability of 85%(approx.)
- ✓ ResNet correctly classify the the input image with 94.48% (approx.) accuracy.
- ✓ Using the data set with AlexNet we get 96% Validation accuracy in 100 epochs.
- ✓ We obtain 96 percent Validation accuracy in 100 epochs using the data set and Inceptionv3.
- ✓ Applying EfficientNet in only 100 epochs, we achieved a remarkable 98 percent accuracy on our validation set.

In terms of the data gathered, we can quickly determine the model's correctness and which one to utilize as the condition we require.

5.Discussion

We have only provided an overview of the top 5 pre-trained models for image classification and how to implement them. Here is a handy table for you to refer these models and their performance

Model	Year	Top-1 Accuracy	Top-5 Accuracy
VGG-16	2014	74.5%	91.2%
ResNet	2015	77.15%	93%
AlexNet	2016	77.1%	80.2%
Inceptionv3	2015	78.8%	89.3%
EfficientNet	2019	84.4%	98%

Although various researchers in the field of computer vision have varied methods for putting up tests, the following tendencies may be seen in general. We go through how the pictures are pre-processed, what kind of data augmentation is utilized, how the optimization mechanism works.

6. Conclusion

Pros:

- ✓ Deep Convolutional Neural Networks represent current state-of the-art techniques in image classifation, object detection and localization
- ✓ Powerful CNN models are VGG-16, ALexNet, ResNet50, Inceptionv3, EfficientNet and so on.

- Open-source libraries for deploying applications with CNN very fast
- Convolutional Neural Networks can share pre-trained weights, which is the base for transfer learning

Cons:

- The interpretation and mechanism of CNN are not clear, we don't know why they work better than previous models
- Large numberof training data and annotations are needed, which may not be practicalin some problems,

References

[1]A survey of image classification methods and techniques for improving classification performance D. Lu &Q. WengDepartment of Geography, Geology, and Anthropology , Indiana State University , Terre Haute, IN 47809, USA.

[2] The Architecture that Challenged CNNs.

[3] 3D CNN-based classification using sMRI and MD-DTI images for Alzheimer disease studies.

[4] Residual Networks (ResNet) – Deep Learning.

[5] ImageNet Classification with Deep Convolutional Neural Networks.

[6] Convolutional neural networks are fantastic for visual recognition tasks
.

[7] Python image search *by* **Adrian Rosebrock** *on* March 20, 2017.

Annon,Nihal Abedin	Introduction, Literature Review, VGG-16, ResNet50
Suparan Sharma	AlexNet, Discussion
Snigdho Dip Howlader	Inceptionv3,Result
Abir-Al-Arafat	EfficientNet, Conclusion