Crossref
Similarity Check
Powered by iThenticate

Reason- A Technical Journal
ISSN 2277-1654
Volume XX 2021

A BRIEF JOURNEY OF CONVOLUTIONAL NEURAL NETWORKS FROM 2012 TO 2017

Deepsubhra Sarkar[1], Pushkal Guha Choudhury[2], Abhisek Dutta[3], Sourish Sarkar[4] and Satyendra Nath Mandal[5]*

Department of Information Technology, Kalyani Government Engineering College, Kalyani- 741235
Email: [1]deepsubhra31@gmail.com, [2]pushkalgc007@gmail.com, [3]mailabhishekdutta@gmail.com, [4]sourishironsides@gmail.com, [5]satyenkgec@gmail.com *Corresponding author

Paper submitted on September 23 2021, Accepted after revision on March 17 2022
DOI:10.21843/reas/2021/84-96/212376

Abstract: With the recent advancements in the field of Deep Learning, largely due to the increase in computational power, the long-standing problem of large-scale image classification has been solved up to a considerable threshold. Since the advent of AlexNet in 2012, there has been an increase in the research and development of various Convolutional Neural Network based model to solve problems in the field of Computer Vision. This may be primarily attributed to the success of AlexNet in successfully classifying 1000 class subset of ImageNet with considerable accuracy. Consequently, many more Convolutional Neural Networks were introduced including VGG Net, Inception Net, ResNet, DenseNet, etc. each with better performance than its predecessor. This paper serves as a brief guide to the architecture of ResNet and some of its predecessors. It will also discuss DenseNet and ResNeXt as improvements of ResNet architecture.

Keywords: Computer Vision, Convolutional Neural Network, Deep learning, Image classification, ResNet, AlexNet.

1. INTRODUCTION

Image classification or generally speaking computer vision has been very difficult for computer scientist for a long time. Effort has been made to present convolutional neural network which is one of the most effective solutions to this problem. In a nutshell, this paper tries to capture a slice of commonly used convolutional neural network architecture and present it in a way that can serve as a quick-start guide to anyone new to this field. The paper also highlights the journey of neural network to bring in more context to the subject. The model-oriented development and usage of various CNNs are the primary areas of focus.

Towards the very dawn of computing, an IBM researcher named Arthur Samuel began working on a different approach to program computers to complete tasks. He referred to it as machine learning. In 1962 he wrote an essay on the topic of "Artificial Intelligence: A Frontier of Automation" [1]. In his essay, he describes a traditional Machine Learning problem to be very difficult. He further reasons that, this difficulty could be attributed to the necessity of describing the algorithm's steps minutely rather than the complexity of the problem in hand.

As it is evident from his observations a step-by-step traditional algorithmic based hard coded approach is out of question. There is a need to find a way to let the computer figure out a way for itself to make sense of the data by giving examples of some sort. To put in layman terms, the computer will be shown a few images of an apple and tell it is apple instead to trying to describe apple as a somewhat round spherical object of red color. Needless to say, but this is somewhat of an oversimplified representation of a much complex problem. So, Arthur Samuel introduces the concepts of weights and talks about a process through which these weights could be automatically adjusted to produce the desired results. Thus, the program would learn to maximize its performance by modifying the weights. This whole concept could the represented in this diagram (Fig. 1).

This concept forms the very basis of modern neural network. In present day context, weight is defined as the parameter that transforms the input data within the hidden layers of network. In other words, it could be described as strength of connections between nodes.

To formalize this idea the following terms namely loss function and label were introduced. Loss function is used to analyze the performance of neural network. It is calculated as the difference between actual results, i.e., label and predicted value. After this we update the weights such that it reduces the loss or in other words give more accurate results. Also, after introduction of Neural Network, the term architecture is often used interchangeably with model.

Thus, the conceptual flowchart is updated again [2] to reflect the Neural Network jargon (Fig. 2). This provides a basic high- level overview of any modern neural network model [1-20].

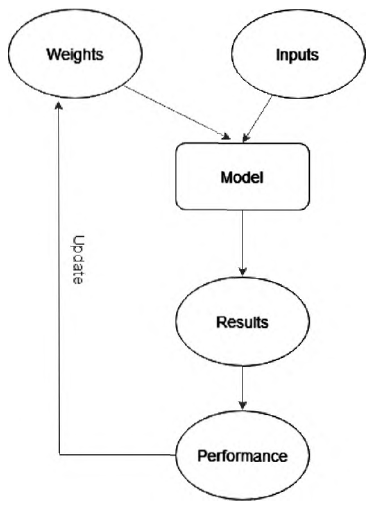

Fig. 1. Arthur Samuel's idea of Machine Learning [1]

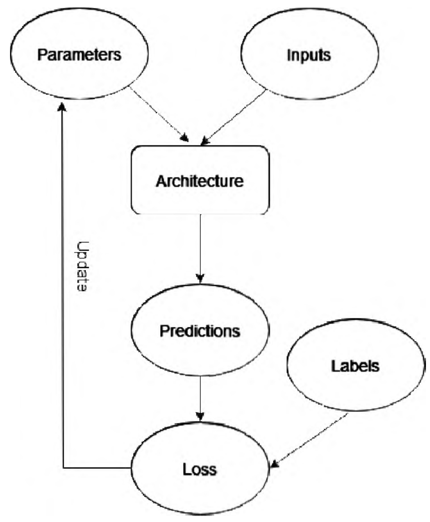

Fig. 2. Modern view of Neural Networks [2]

Aim of This Paper : The decade of 2010-2019 was the dawn of deep learning in the field of computer vision on a large scale. The aim of this article is to document this period of twilight, which can be a source of valuable significance for future ahead. With the immense development in computing power and A.I. the scope of growth and usability for computer vision is immense in every field, be it Medical, Retail, Military, Transport.

1.1. History of Artificial Neural Network

In 1959, Stanford researchers Bernard Widrow and Marcian Hoff developed two models named as ADALINE and MADALINE [3]. ADALINE is a single layer ANN. It was based on the neuron concept proposed by McCulloch and Pitts in ADALINE consists of a weight, a bias and a summation function. MADALINE or Many ADALINE is basically a multilayer network of ADALINE units. In 1972, Kohonen [20] and Anderson [19] both independently developed a network similar to an array of analog ADALINE circuits without realizing it. They used matrix mathematics to approach the problem.

In 1974, Werbos [16] introduced back-propagation algorithm to train multi-layer networks. This proved to be a major mile-stone. In 1988, Kunihiko presented Neo-cognitron [17] which was capable of visual pattern recognition. It was in 1998 that Yann Lecun et al. [18] made Convolutional Neural network with back-propagation algorithm for document analysis. Finally in 2012, Alex krizhe-vesky AlexNet [3] which marked the advent of modern Convolutional Neural Network.

1.2. Structure of Artificial Neural Network

Any Artificial Neural Network is essentially a collection of units known as artificial neuron. The structure of these artificial neuron is inspired by the neurons present in human brain (Fig. 3). The connections between neurons are comparable to synapses in the

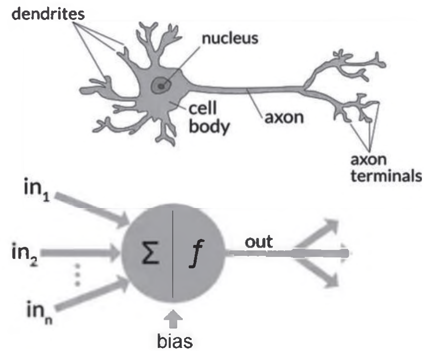

Fig. 3. Biological Neuron and Artificial Neuron [11]

brain [11]. These connections are able to transmit signal to each other much like synapses in our brain. Instead of a signal that biological neurons receive artificial neurons receive a real number. Then the artificial neuron computes a weighted average and passes it through a non-linear function to formulate an output which is in turn passed to the next neuron. The connections between neurons are called edges. Each edge has a weight assigned to it. This weight is then adjusted in order to “learn”. The increase or decrease of weight affects the strength of input signal. Neurons may also implement some kind of threshold function. Although, most modern neural networks use linear activation functions. The neurons are arranged in layers. Typically, a neural net-work is an ensemble of different type of layers each having some special tasks [12]. The first layer is called input layer and the last layer output layer, everything in between is referred to as hidden layer (Fig.4).

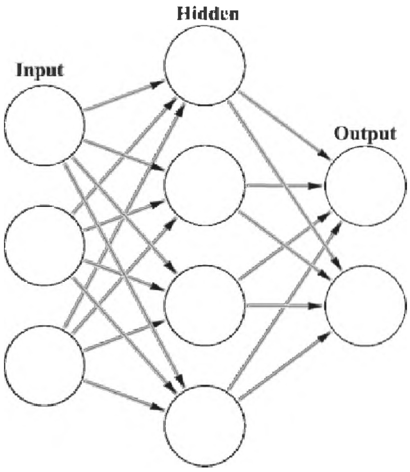

Fig. 4. Artificial Neural Network [12]

1.3. Deep Learning

Deep Learning is a branch of machine learning primarily inspired by the way human brain functions. It, mainly consists of algorithms that are basically different arrangement of Artificial Neurons. The word Deep is used here to signify the large Artificial Neural Networks. In comparison to other Machine Learning methods Deep Learning is much more computationally intensive. However, the key difference between Machine Learning and Deep Learning is in the process of feature extraction. Generally, Machine Learning techniques use handmade features by applying several feature extraction algorithms including Scale Invariant Feature Transform, Histogram Oriented Gradient (HOG), Empirical mode decomposition and many more. Finally comes the leaning algorithms like SVM (support vector machine), Random Forest, PCA (Principal Component Analysis), Linear Decrement Analysis, etc. consists of the various algorithms used on the extracted features to perform classification [1-8].

Feature Extraction Process

In case of Deep Learning, the process of feature extraction is automated and is represented in multiple layers. Hence, the name Deep Neural Network. Since most NNs contain multiple hidden layers other than the input and output layer. This is the chief advantage of using deep learning instead of machine learning models. It is required since the process of definition of feature extraction manually is not possible in case of Deep NNs since it would have to be done at multiple layers. Also, the two-step process of Feature extraction and classification is now unified in a single step. This process of feature extraction is powered by back-propagation.

2. CONVOLUTIONAL NEURAL NETWORK

Since the paper is focusing on the journey towards ResNet, the following section will examine several popular state-of- the-art CNN models which were predecessor to ResNet and finally ResNet will be discussed in this section. Most of the multi-layered Convolutional Neural Networks comprises of some particular group of layers which are common to most, this includes the convolution layer, fully connected layers, the sub-sampling layer, and the SoftMax layer. In general, any typical CNN model is basically an arrangement of numerous convolutional layers and sub-sampling layers succeeded by fully connected layers and at last comes the SoftMax layer [3-10].

In the initial stages of visual recognition, the dataset sizes were small compared to the current standards. Back then, the main problem for the image classification models was over-fitting. But as the time progressed the sizes of datasets increased drastically and along with that the resolution of the images increased a lot. It created lot of trouble for the classic feed forward networks.

2.1. Image classification using Neural Networks

To perform image classification using neural networks we need to perform a multitude of transformations and computational techniques in order of achieve the end results. It involves feature extraction which is mostly done using convolutional layers, dimensionality reduction using pooling layers and soon. The chief components of any neural network model for image classification are as follows–

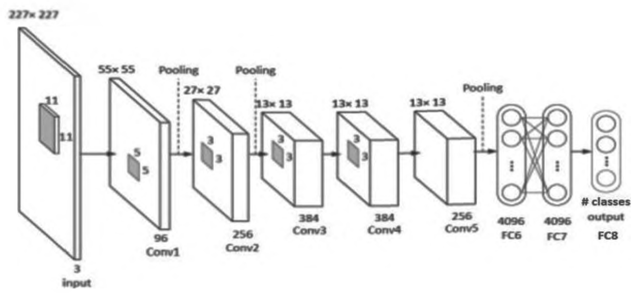

Fig. 5. AlexNet architecture diagram [3]

1) **Feature extraction using Convolutional Layers** - The most vital part of any neural network, made for image classification are the convolutional layers. They are kernels that extract spatial features of the input image. Generally square kernels of odd lengths are used. The most common dimension being 3 x 3 x N where N is the number of layers.
2) **Dimensionality Reduction**-Max- pooling and Average-pooling are the two most common strategies adopted to reduce the size of the inputs. But before sending this to the Fully Connected layer the 2D inputs are passed through a Flattening Layer and converted into a long 1D array.
3) **Activation functions and Normalization** - Activation functions like ReLU and Sigmoid functions serve as activation functions. These are generally present in con- junction with convolutional layers. Normalization like batch norm and group norm are used to improve the model training procedure.

2.1.1. AlexNet (2012)

In the year 2012, AlexNet with its state-of-the-art architecture shook the field of visual recognition completely. It significantly outperformed its prior competitors and won the ImageNet Scale Visual Recognition Challenge that year.

ImageNet SVRC: 2012 (ImageNet Scale Visual Recognition Challenge) had a huge dataset, containing around 1.4 million images in total. On this dataset AlexNet reduced the top-1 error and top-5 error to 36.7% and 15.3% respectively, which was remarkable compared to the second-place top-5 error rate of 26.2%.

AlexNet (Fig. 5) was one of the first models which tried to utilize the depth of a convolution network with its 5 convolution layers and 3 fully connected weighted layers and for output a 1000-way SoftMax was used. Some key features of this Architecture [3]:

1) ReLU Non-Linearity: Previously the standard of activation function was non-

linear saturating functions like tanh or sigmoid functions which suffer from vanishing gradient problem. But in AlexNet instead of that non- linear saturating Rectified Linear Unit (ReLU) was used, this reduced the model training time considerably also reduced the error rate by nearly 25%. In top of that, Local Normalization was used to prevent unnecessarily high learned variables.

2) Multiple GPUs: Due to the large nature of the database the neurons are distributed among two GPUs faster parallel computation. This reduced the training time and top-5 and top-1 error rate. The kernels in the third convolution layer are connected to kernels of the second convolution layers residing on both the GPUs. Fully Connected layers on both the GPUs are connected with each other.
3) Overlapping Pooling: In CNN's, usually depth of the output increases with each layer to capture the spatial features of an image more accurately. That results in a huge amount of data. To address this issue pooling is used to reduce the size of each output; it eases the computation quite a lot. To preserve the features better in AlexNet overlapping pooling is used.
4) Dropout: Over fitting is one of the main problems while training models like this. To reduce the over fitting problem, one of the methods is Dropout. In dropout, some random neurons are made inactive by setting their respective weights to zero, which in turn forces other neurons to step-up. This results in less codependency among neurons and less chances of over-fitting. In case of AlexNet, Dropout was used only in the first two fully connected layers.
5) Data Augmentation: Another way of reducing overfitting is to use different versions of the same image preserving the labels. Data Augmentation in images are mainly done in two ways
   a) Image translations and horizontal reflections, extracting random patches of the images etc. This increases the size of the dataset, though they are highly inter-dependent.
   b) In images of real life objects the lighting and contrast may vary time to time. To solve this problem in second type of image augmentation the color, contrast and brightness of the image is varied to reduce overfitting. After using this technique in AlexNet the top-1 error was reduced.

2.1.2. VGG (2014)

In 2014, ImageNet Visual Recognition Challenge the GoogleNet came first, while the VGGNet [4] invented by Visual Geometry Group of Oxford university was the first runner up with 7.3% top-5 error rate. The main aspect of VGGNet is its simplicity and depth. Karen Simonyan and Andrew Zisserman of the University of Oxford was the inventor of VGG. They tried variations with different number of layers from 11 to 19. Among those, VGG16 with 16 weighted layers gave the best result.

While building the ConvNet architecture their main aim was to reduce the number of parameters. For that they used a greater number of small filters rather using one large filter. For instance, combination of three 3-by-3 Convolution layer will give the same field as one 7-by-7 filter. But the number of parameters will be lot less.

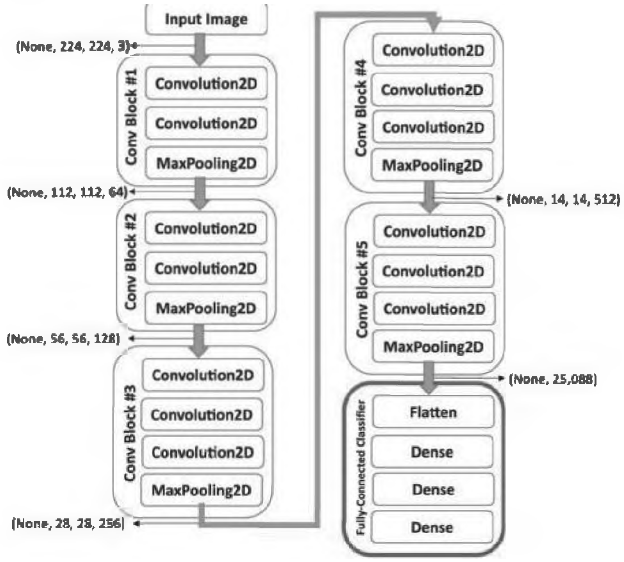

Fig. 6. Schematic diagram of VGG-16 [10]

Keeping this in mind while building the ConvNet unlike previous models like AlexNet (which used 11-by-11 filters) or ZFNet (that used 7-by-7 filter), used a greater number of 3-by- 3 that gave the architecture more depth and a smaller number of parameters to train. This in turn results in less chances of over-fitting.

The most successful VGG16 model [10] had 13 Convolution layers, 3 Fully connected layers with 4096 neurons each, and a SoftMax layer which gives output of 1000 labelled classes (Fig.6). VGG was trained on 4 GPUs 1.3 million training images. The training procedure mainly follows the same process as AlexNet with minor adjustments. On 50K validation images and 100K test images the VGG with 16 layered D-configuration gave top -1 error and top-5 error of 25.6% and 8.1% respectively.

This model was very important as it showed the path for the upcoming CNN models. Due to its simplistic model, larger depth and less chances of over-fitting this model was used as a base for upcoming lots of models.

2.1.3. InceptionNet (2014)

The idea of was to basically to gather different information from same input feature map using kernels of different size. Instead of just depending on 3-by-3 kernels like most convolutional neural network, it makes different branches with each branch having different kernel size [14]. The different kernel size that InceptionNet used are 1-by-1, 3-by-3 and 5-by-5 in their model GoogleNet which won the ILSVRC 2014 competition.

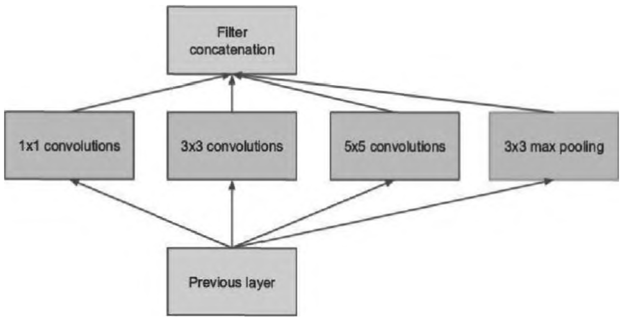

Fig. 7. Diagram showing naive version of Inception module [14]

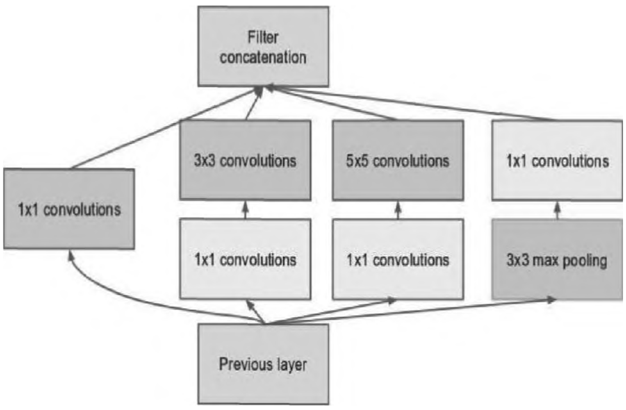

Fig. 8. Diagram showing Inception module with dimensionality reduction [14]

After initially operating under the naive approach (Fig. 7), it was seen to be computationally expensive to run the 5-by-5 kernels. So, to make the model more efficient the concept of dimensionality reduction was introduced where before the 5- by-5 and 3-by-3 kernel convolutions a 1-by-1 convolution was performed to reduce the number of channels (Fig. 8). This leads to better results with lesser computation cost.

Inception module is basically a combination of 1-by-1 convolution with 64 filters, 3-by-3 convolution with 128 filters, 5-by-5 convolution with 32 filters and 3-by-3 max pooling with 32 filters with same padding. In inception module all combinations are done in single layer. 1-by-1 convolutional layer used to reduce computational complexity as well as depth channel. Advantage of inception net is to reduce computational cost without changing the speed and accuracy.

2.1.4. ResNet (2015)

The years following the introduction of AlexNet, the first CNN based architecture, saw the rise in the number of layers in the DNNs in order to increase accuracy (Fig. 9). It was observed that after a maximum threshold number of layers, the accuracy did not seem to increase, rather the architectures with a lower number of layers were performing better. This took place because of the vanishing gradient problem, where the gradient of the loss function approaches zero making it impossible to reduce the error in these networks and to solve this problem, was introduced in the year 2015 [15].

Residual Blocks

ResNets use the concept of skip connections or short-cut connections to solve the vanishing gradient problem [15]. Up until this point, each layer in a conventional DNN feeds into the next layer. In ResNets, each layer

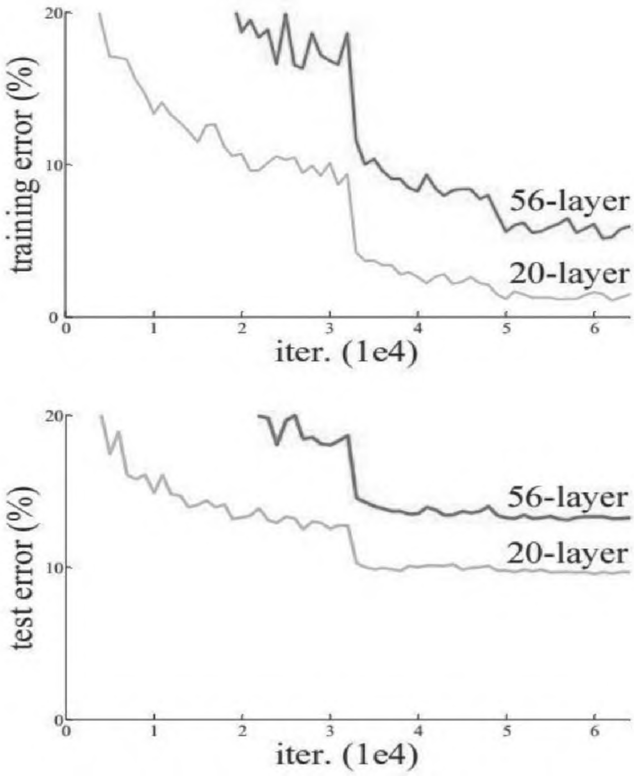

Fig. 9. Diagram comparing plain 56 layers vs plain 20 layers network training and test error

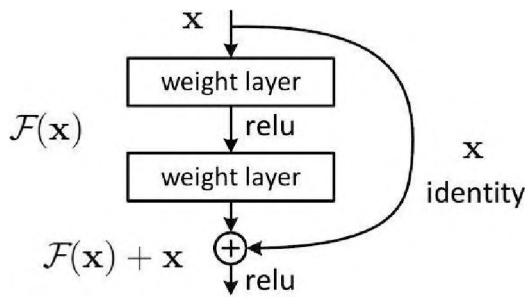

Fig. 10. Diagram showing a residual block [15]

acts as an input for the next layer and also directly into farther layers while skipping a few layers. If one takes any block from neural network whose input is x and output is H(x), one can represent the residual/difference as f(x) := H(x)-x, if one rearranges this, one gets the output distribution as H(x)=f(x)+x.

If one examines this, one will understand that the network is trying to learn the residual f(x), this makes it very easy for the layers to learn an identity function by assigning 0 to the value of the residual. In this way a very deep network can be turned into a smaller network if that is what the network requires to increase accuracy. Authors of the paper hypothesized that it was easier for the network to train a network with skipped connections than try to fit a very deep network.

3. FURTHER WORKS

Since at the time of writing this paper, it is 2021 and ResNet was introduced way back in 2015, it is obvious that better architecture has been proposed, as this field is intensively

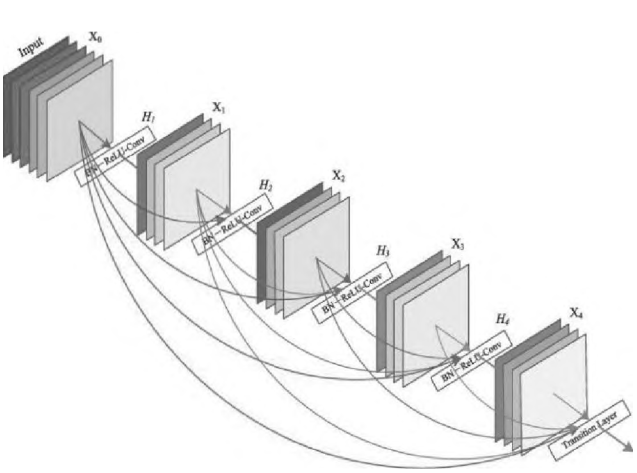

Fig. 11.Diagram showing a Dense block. Each layer is connected to all the previous

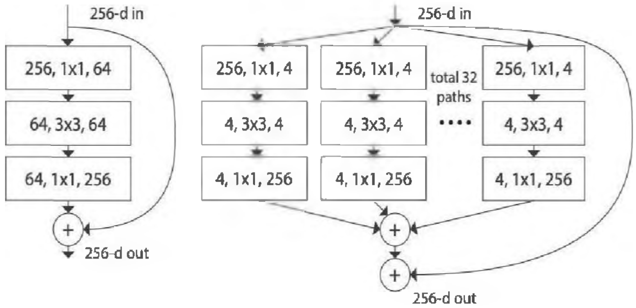

Fig. 12. Left: A ResNet block. Right: A ResNeXt block with 32

researched. So, there is a need to discuss a few more architectures briefly, which arguably draw their inspiration from ResNet architecture.

3.1. DenseNet (2016)

In general, convolutional neural networks with n layers have n connections i.e., each layer connected to the next layer. ResNet revolutionized this by introduction a skip connection. DenseNet, took this idea one step forward by connecting each layer to every other layer in a feed-forward manner, thereby having a total of n(n+1)/2 (Fig.11) However, the big difference between ResNet and DenseNet is that DenseNet does not add the output and input feature maps instead it concatenates them. Thus, to make every layer the inputs are the feature map of all the previous layers. In turn this layer and its own feature-maps forms the input to all the succeeding layers.

In DenseNet [7], information from all the previous layers is pre- served until the very last layer. Thus, the biggest advantage of DenseNet over most traditional Convolutional Neural Network architecture is that it solves the varnishing gradient problem due to its densely connected architecture.

3.2. ResNeXT (2017)

ResNeXt [8] is another convolutional neural network which in a way generalizes the concept of ResNet by introducing another dimension called "cardinality" along with the general height and width of ResNet.

In ResNeXt instead of vertical expansion i.e.- increasing the number of channels and number of layers as done in most traditional CNNs focus is shifted to horizontal expansion. This idea of branching out was seen in InceptionNet too but their different sized kernels were used for different branches but here only same sized kernels are used.

It must be noted that after introduction of ResNeXt, traditional ResNet just becomes a special case of ResNeXt with unit cardinality.

3.3. MobileNetV2 (2019)

Convolution Networks with high accuracy, trade off on latency and demand of high computational power, is difficult to achieve on devices with low computational power, like mobile. MobileNet provides an architecture that requires low compute capability and low latency, but still maintaining good accuracy. MobileNetV2 made by Google exhibits [22] some innovative ideas, like –

1) Linear Bottleneck
2) Inverted Residuals
3) Depth-wise Separable Convolution

Culminating these ideas, along with the previous works MobileNetV2 maintains a balance between accuracy and latency even on smaller devices.

4. COMPARISON OF DIFFERENT CNNs

CNNs	Challenges	Findings	Limitations
AlexNet (2012)	ImageNet SVRC: 2012 Containing over 1.4 million images	One of the first to implement Deep Learning in Computer Vision with significant effects	Takes huge amount of computation power and time to train
VGG (2014)	ImageNet SVRC: 2014	Instead of using large filters, using multiple small filters gives more depth and also better results	Top-5 error was 7.3%
InceptionNet (2014)	Previous models were simplistic but accuracy was not sufficient	Instead of naive approach, using various dimensionality of Kernel extract various features	Due to a greater number of Layers the time required to train more
ResNet (2015)	Only increasing number of layers does not increase the accuracy, rather performance suffers	Instead of using linear approach of data flow through layers, skipping connections improves accuracy	ResNet needs more floating pointing operations than other latest architectures

5. PROS AND CONS OF CONVOLUTIONAL NEURAL NETWORKS

Pros and Cons of Convolutional Neural Networks. The Pros-

1) Convolution Neural Networks are very accurate models compared to its predecessor like HOG+SVM or SOFT+ SVM [21].
2) As these CNNs use machine learning for the purpose of image classification these models are very adaptive to the changes.
3) These initial CNNs worked as a foundation to develop advanced models which can be used in daily life to efficiently and accurately perform classification and detection in various fields.
4) A generalized CNN model can be easily used for a specialized and targeted use case without having to train the model from scratch using the concept of transfer learning [22].

The Cons-

1) Due to the numbers of the fully connected layer in a CNN, it is time consuming to train a network to successfully classify and image. Moreover, even the number of layers can affect the time required to perform the image classification.
2) To train a network the amount of computational power and storage capacity required is very huge and costly.
3) With CNN images can only be classified, but to perform Object Detection i.e., to determine the position of an object in image more complex networks like RCNN, Fast RCNN, Masked RCNN, YOLO etc. are required [23].

6. CONCLUSION

The paper provides a concise overview of the major Convolutional Neural Network based architectures used for the purpose of image classification. Focus has been kept on ResNet since it is a major improvement in the world of Convolutional Neural Networks. The introduction section provides with an alternate way of approaching the concept of Deep Neural Network by starting from a macro level overview. Thereby the reader approaches the subject of Neural Network in a Top- Down manner. This is inspired by the authors of FastAi [2]. Then the reader is provided with a history and basic construct of an Artificial Neural Network.

In the previous sections four major Convolutional Neural Net-work architectures- AlexNet, VGG, InceptionNet and ResNet, are discussed at length. In the end the conclusion is drawn by reviewing two architectural improvements after ResNet namely DenseNet and ResNeXt which extend and/or modify the idea of skip connections to

REFERENCES

[1] Samuel, L., Artificial Intelligence: A Frontier of Automation, The Annals of the American Academy of Political and Social Science, Vol. 340, No.1, pp.10-20, 1962.

[2] Howard, J. and Gugger, S., Deep Learning for Coders with Fastai and Pytorch, O'Reilly Media Inc., 2020.

[3] Krizhevsky, A., One Weird Trick for Parallelizing Convolutional Neural Networks, 2012. https://doi.org/10.48550/arXiv.1404.5997.

[4] Simonyan, K. and Zisserman, A., Very Deep Convolutional Networks for Large-Scale Image Recognition, arXiv: 1409. 1556, 2014.

[5] Szegedy, C., Liu, W., Jia, Y., Sermanet, P., Reed, S., Anguelov, D., Erhan, D., Vanhoucke, V. and Rabinovich, A., Going deeper with convolutions, 2015 IEEE Conference on Computer Vision and Pattern Recognition, pp.1-9, 2015.

[6] Kaiming, H., Xiangyu, Z., Shaoqing, R. and Jian, S., Deep Residual Learning for Image Recognition, IEEE Conference on Computer Vision and Pattern Recognition, pp.770-778, 2016.

[7] Huang, G., Liu, Z., and Weinberger, K.Q., Densely Connected Convolutional Networks, 2017 IEEE Conference on Computer Vision and Pattern Recognition, pp.2261-2269, 2017.

[8] Xie, S., Girshick, R.B., Dollár, P., Tu, Z. and He, K., Aggregated Residual Transformations for Deep Neural Networks, 2017 IEEE Conference on Computer Vision and Pattern Recognition, pp.5987-5995, 2017.

[9] Khvostikov, A., Aderghal, K., Benois-Pineau, J., Krylov, A. and Catheline, G., 3D CNN-Based Classification Using sMRI and MD-DTI Images for Alzheimer Disease Studies, Computer Vision and Pattern Recognition, 2018. https://arxiv.org/abs/ 1801.05968

[10] Gopalakrishnana, K., Khaitan, S. K., Choudhary, A. and Agrawal, A., Deep Convolutional Neural Networks with Transfer Learning for Computer Vision-Based Data-Driven Pavement Distress Detection, Construction and Building Materials, Vol. 157, pp.322-330, 2017.

[11] Abhi, S., What is the Differences between Artificial Neural Network (Computer Science) and Biological Neural Network, 2021. https://www.quora.com/What-is-the-differences-between-artificial-neural-network-computer-science-and-biological-neural-network, date of access: 14/03/ 2021.

[12] Srivastava, T., How Does Artificial Neural Network (ANN) Algorithm Work? Simplified!, https://www.scribd.com/ document/ 365270266/How-Does-Artifical-Neural-Network-ANN-Works-Simplified, 2014, date of access: 14/03/2021.

[13] Alom, M. Z., Taha, T. M., Yakopcic, C., Westberg, S., Sidike, P., Nasrin, M. S., Essen, B. C. V., Awwal, A. A. S. and Asari, V. K., The History Began from AlexNet: A Comprehensive Survey on Deep Learning Approaches, Deep Learning Monitor, 2018. https://deeplearn.org/arxiv/46836/the-history-began-from-alexnet:-a-comprehen sive-survey-on-deep-learning-approaches

[14] Widrow, B. and Lehr, M.A., 30 Years of Adaptive Neural Networks: Perceptron, Madaline, and Backpropagation, Proceedings of the IEEE, Vol. 78, No.9, 1990.

[15] McCulloch, W. S. and Pitts, W., A Logical Calculus of the Ideas Immanent in Nervous Activity, Bulletin of Mathematical Biophysics, Vol. 5, pp.115-133, 1943.

[16] Werbos, P., Beyond Regression: New Tools for Prediction and Analysis in the Behavioral Sciences, Ph.D. Thesis Submitted to Applied Mathematics Depart-ment, Harvard University, 1974.

[17] Fukushima, K., Neocognitron: A Hierarchical Neural Network Capable of Visual Pattern Recognition, Neural Net-works, Vol. 1, No.2, pp.119-130, 1988.

[18] Lecun, Y., Bottou, L., Bengio, Y. and Haffner, P., Gradient-Based Learning Applied To Document Recognition, Proceedings of the IEEE, Vol. 86, No.11, pp.2278-2324, 1998.

[19] Anderson, J.A., Mathematical Biosciences, Vol. 14, No. 3-4, pp.197-395, 1972.

[20] Kohonen, T., Lehtio, P., Rovamo, J., Hyvarinen, J., Bry, K. and Vainio, L., A Principle of Neural Associative Memory, Neuroscience, Vol. 2, No. 6, pp.1065-1076, 1977.

[21] Chaganti, S.Y., Nanda, I., Pandi, K.R., Prudhvith, T.G.N.R.S.N. and Kumar, N., Image Classification Using SVM and CNN, 2020 International Conference on Computer Science, Engineering and Applications, pp.1-5, 2020.

[22] Hussain, M., Bird, J.J. and Faria, D.R., A Study on CNN Transfer Learning for Image Classification, Advances in Computational Intelligence Systems, pp.191-202, 2019.

[23] Joseph, R. and Farhadi, A., Yolov3: An Incremental Improvement, pp.1-6, 2018.