Deep Learning Architectures: A Hierarchy in Convolution Neural Network Technologies

Shruti Karkra, Priti Singh, Karamjit Kaur, and Rohan Sharma

Abstract Albeit deep learning has chronicled roots and has been applied to computer vision task since 2000 but a decade ago, neither the expression "Deep learning" nor the methodology was well known. This dormant field regains consciousness when a highly influential paper "Image Net Classification with Deep Convolutional Neural Networks by Krizhevsky, Sutskever and Hinton' in 2012" was published. Now, availability of abundance of data, computational power, and improved algorithms has contributed altogether and brought this technology to forefront in the field of machine learning. In this paper, we focus on growth of various convolution neural network architectures (deep learning architectures), from their predecessors up to recent state-of-the-art deep learning systems. The paper has three sections: (1) Introduction about neural networks along with necessary back ground information. (2) Hierarchy of classical and modern architectures; In this section, the existing methods are explained and their contribution and significance in field of machine learning are highlighted. At last, we point out a set of promising future works and draw our own conclusions.

Keywords Artificial neural network (ANN) · Machine learning · Deep learning · Convolution neural networks (CNN or CovNet) · Image classification · Computer vision

S. Karkra (✉) · P. Singh · K. Kaur · R. Sharma
Amity University Haryana, Gurgaon, India
e-mail: Shrutikarkra@gmail.com

P. Singh
e-mail: psingh@ggn.amity.edu

K. Kaur
e-mail: kkaur@ggn.amity.edu

R. Sharma
e-mail: rohans1497@gmail.com

V. C. Pandey et al. (eds.), *Advances in Electromechanical Technologies*,
Lecture Notes in Mechanical Engineering,
https://doi.org/10.1007/978-981-15-5463-6_39

1 Introduction

ANNs are the information processing systems, which are vigorously motivated by the way organic sensory system, (e.g., brain) works. Artificial neural networks are made up of large number of interrelating units called neurons. These neurons are laced in distributed manner to learn from the input in order to streamline the final output. The basic model and mathematical model of simple ANN are shown in Figs. 1 and 2. The information (generally multidimensional vector) is fed to the input layer and propagated to shrouded (hidden) layers. The middle layers then learn features and decision taken by these layers depends upon the former layer [1].

Study in artificial neural networks has started right around 100 years prior. For a long time, there was no broadly acknowledged organic model for visual neural systems, until practical and experimental works clarified the structure and function of the mammalian visual cortex [4]. Thereafter, theoreticians developed models which resemble to biological neural networks.

Till 1950s, Perceptron, Habbian, ADLINE, and MADALINE are the network models that proven to be the mile stone in the history of artificial neural network.

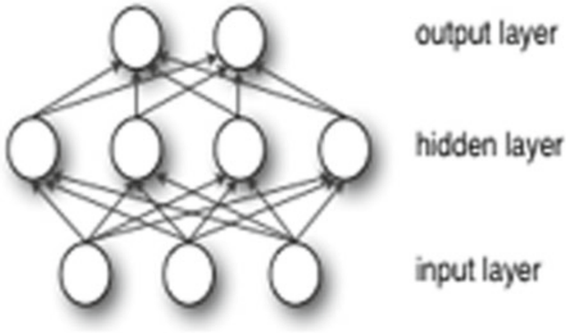


Fig. 1 Simple feed-forward neural network [2, 3]

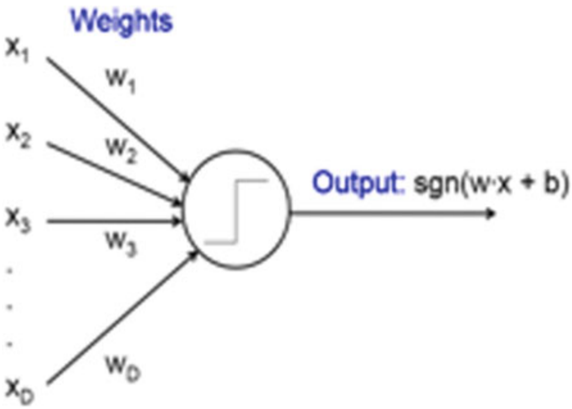


Fig. 2 Mathematical model for artificial neuron [2, 3]

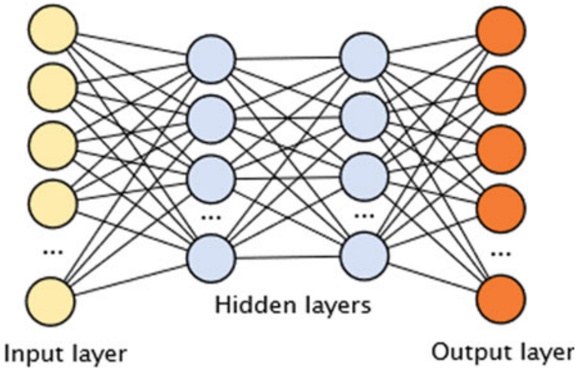


Fig. 3 Deep neural network with several hidden layers [5]

Thereafter, invention of back propagation in 1960s and convolution neural networks in late 90s has totally changed the way we look at neural networks. Convolution neural networks were developed in 1998, named as Le-Net(First CNN), and since then, these are applied to visual tasks. However, despite a few scattered applications, they were dormant until the mid-2000s. But due to abundance of data, efficient algorithms and computational power exponential growth have been noticed since 2012. Table stated below mentions the brief history of neural networks and highlighting the key events:

Figure 3 is the basic structure of number of common neural networks. It is a feed forward network with input, two hidden layers, and one out layer. Neural networks, which have more than one hidden layer (as shown below) are called deep neural networks. Deep neural networks (DNN) are also referred as deep learning networks. These are different from the traditional single-layer-hidden neural networks because of number of hidden layers or their depth.

Based upon the previous layer output in deep learning neural networks, each layer of neurons trains on particular set of features. The deeper you go, more complex and composite features can be recognized. The most significant and robust deep learning network is **convolution neural network (CNN)**.Convolutional neural network is a class of deep learning techniques which turned out to be which has become overruling in various computer vision tasks and is drawing attention over various domains. CNN is made up of numerous layers, such as convolution layers, pooling layers, and fully connected layers, and is intended consequently and adaptively learn spatial features through a back propagation algorithm [6, 7] (Fig. 4).

Complete CNN architecture is summarized as below

1. Input layer	Feature extraction
2. Convolution layer	
3. Pooling layer	
4. Fully connected layer	Classification/prediction/recognition, etc.

CNN accepts three-dimensional input and transforms it through all connected layers into a set of class scores given by the output layer. First of all, the convolution

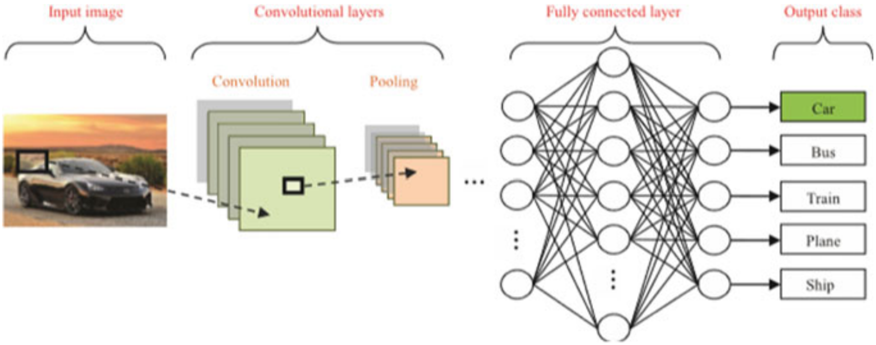


Fig. 4 Layer wise detailed CNN architecture [8]

operation is performed between the input image pixels and window filter (kernel, generally of size 3 × 3 or 5 × 5). The output obtained thus is called activation map or feature map. After this, nonlinear activation function is applied on feature map called ReLu operation. It replaces all the negative pixel values with zero and introduces nonlinearity. Then, pooling is done to reduce the size of receptive field. Reducing the size of receptive field decreases the training time and massive calculation of parameters. The last layer in CNN is the classification layer (Fully connected layer). Here, the higher-order features are transformed to class scores or probabilities. Back propagation algorithm is then used for training the CNN. Training of Covnet (CNN) is done to lessen the differences between estimated target predictions and actual ground truth label [8].

As mentioned in Table 1, there are many convolution architectures that have been released since 2012. Practically, all CNN structures pursue a similar general plan standard of progressively applying convolutional layers to the input, and then down sampling the spatial dimensions and learning the feature maps [9]. Earlier, the system models contain just stack of convolutional layers, but the recent architectures develop new and imaginative ways for building convolutional layers and help in more accurate and competent learning. LeNet, AlexNet, ZfNet, and VggNet are the standard networks. DenseNet, GoogLeNet, Inception (All four version), Exception, ResNext, Network in Network, and many more are the advanced networks. In the next section, a few classic architecture and main modern architecture are defined (Fig. 5).

2 LeNet (1998) [11, 12]

It is the oldest convolutional network which is designed for handwritten and machine-printed character recognition. Main features are:

- Use average pooling

Table 1 Important contribution toward neural network and deep learning architectures [10]

Model/architecture/network	Year and contribution
MCP model: first artificial neural network	1943, McCulloch & Pitts
Hebbian learning rule	1949, Hebb
Perceptron	1958, Frank Rosenblatt
Backpropagation	1974, Werbos
Boltzmann machine	1985, Ack-ley, Hinton & Sejnowski
Restricted Boltzmann machine	1986, Smolensky
Recurrent neural network	1986, Jordan
LSTM	1997, Hochreiter & Schmidhuber
Le Net (starting age of deep learning)	1998, Lecun
Deep belief network	2006, Hinton
AlexNet, starting age of CNN	2012, Krizhevsky, Sutskever, & Hinton
ZFNET	2013, Matthew Zieler, RobFergus
Google	2014, Google
Resnet	2015, Kaiming He, Microsoft
Inception V3-V4	Google 2016
Exception and many more	2017

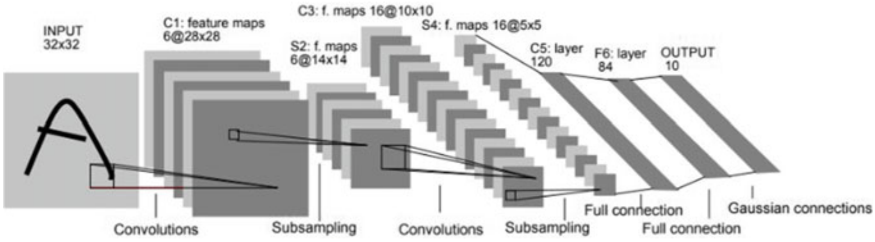


Fig. 5 Layer wise LeNet architecture for handwritten digit recognition [11]

- Nonlinear activation function is Sigmoid or tanh.
- Use of FC layers at the end
- Le Net is trained on approx. 60 K training image (use MNIST database)

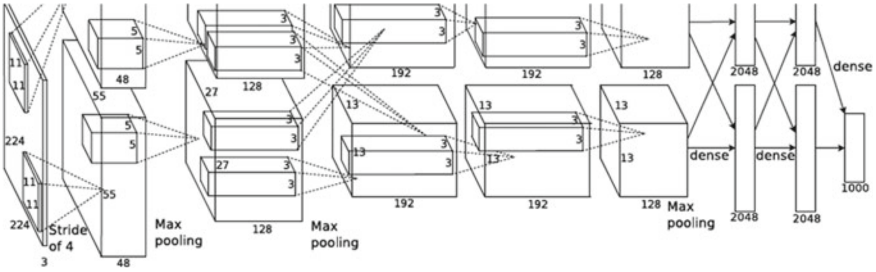


Fig. 6 Layer wise AlexNet architecture [15]

3 AlexNet (2012) [13, 14]

ILSVRC 2012 winner. AlexNet expressively outclassed all the earlier contenders and won the competition by challenge by reducing the top-5 error from 26-15.3%. System architecture of AlexNet and LeNet is fundamentally same except the former is more profound, with more number of filters, and more number of convolution layers, AlexNet uses imagenet database and is trained on two Nvidia Geforce GTX 580 GPUs for six days. Main features are:

- Use Max pooling
- Use Relu for nonlinearity function (faster than tanh function)
- Use data augmentation techniques to enlarge the data base, hence more data and large model with seven hidden layers and 60 M parameters (Fig. 6).

4 VGGNet (2014) [1, 14]

VggNet scored ILSVRC 2014 competition. Like AlexNet, VGGNet utilizes only 3 × 3 convolutions, but more number of filters. It is trained on four GPUs for 2–3 weeks. It is the true deep learning neural network with 16 CONV/FC layers. VGGNet consists of 140 M parameters, which can be a bit challenging as well as costly to access. Main features are:

- Uniform architecture
- Large receptive fields swapped by consecutive layers of 3 × 3 convolutions filters.
- Keep the benefits of small filter sizes (minimal loss of spatial information)
- Number of filter increases to almost double after each pooling layer, hence spatial information decreases but depth of the network increases.
- Worked well on both image classification and localization tasks (Fig. 7).

The development of VggNet architecture has demonstrated that profundity (depth) of the system is the crucial component for good performance.

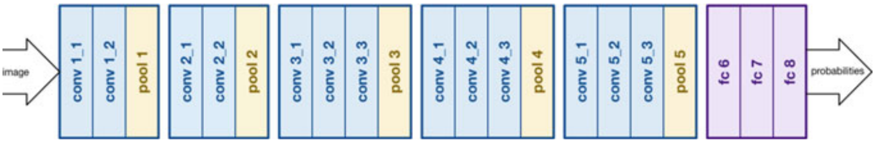


Fig. 7 Layer wise VGG Net architecture [1]

A drawback of the VGGNet is its increasing cost to access and utilize significantly more memory and parameters. Mostly, these parameters are defined in initial few layers; hence, FC layers can be expelled without no downsizing the architecture performance.

5 Inception (GoogLe Net) (2014)

Till VGG architecture, the accuracy and number of parameters are directly proportional to the depth or number of layers. But if training data is small then bigger models are more likely to over fit. With deep networks, the number of parameters increases hence the complexity and computational cost increase. Inception model proposes new idea of moving to sparsely connected architecture. This approach lets you to reduce error and maintain the "computational budget," while increasing the depth and width of the network.

Hence, just like any other architecture, reason behind the development of Inception was also to reduce error. Before this, the convolutional networks were only made deeper to increase the accuracy. But this lead to overfitting with limited data, and an exponential increase in the computational resources is required.

The inception network is the breakthrough achievement in improvement of CNN classifiers. The Inception network is complicatedly planned and uses many methods to improve performance (training speed, accuracy). GoogLeNet or Inception is the winner of the ILSVRC 2014. It achieved a top-5 error rate of 6.67%! Its persistent progression leads to the formation of many improved variants of the network. Firstly, the network familiarized a**s GoogLeN**et or Incepti**on-**v1. Next variant is inception V2, and in this model, the concept of factorization is introduced. Then idea of asymmetric factorization, and batch normalization is introduced in inception V3. Inception V4 and Inception-ResNet are explained together. Each version is an iterative improvement over the previous one [16].

5.1 Inception Module V1 (2014)

The Inception architecture of Google Net is designed to perform well even under strict constraints on memory and computational budget. Its architecture comprised of a 22-layer deep convolution layers. Inception V1 compute 5 M parameters, which are 12 times less than the parameters computed by alexNet use 60 M parameters [17]

With Inception, the network was not only made deeper but also wider, i.e., Instead of having just a single size filter at input, multiple filters of different sizes were introduced, thereby making the network wider. It performed convolution on three different sizes of filters (5 × 5, 1 × 1 and 3 × 3). After convolution operation, the Max pooling is applied; then, the outputs are concatenated to single vector output and sent to the next inception module. The number of input channels was limited by introducing an extra 1 × 1 convolution before 5 × 5 and 3 × 3 convolutions. The 1 × 1 convolutions are computationally cheaper than 5 × 5 convolutions. Hence, the less number of inputs helped in decreasing the computational requirement (reduce dimensionality). Inception reduced the computation by using sparsely connected neurons/layers, which significantly reduce the computation power by 84% in some cases while increasing the accuracy [18] (Figs. 8 and 9).

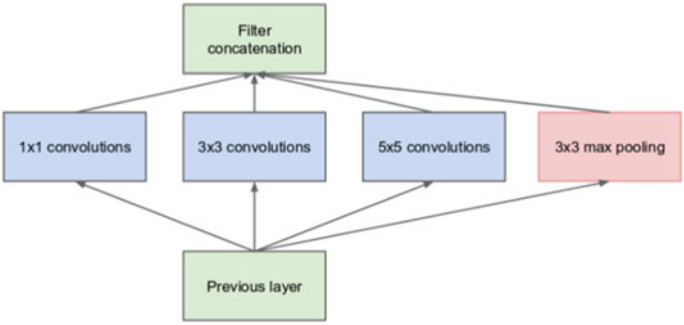


Fig. 8 Inception module versiion1 with naïve version [18]

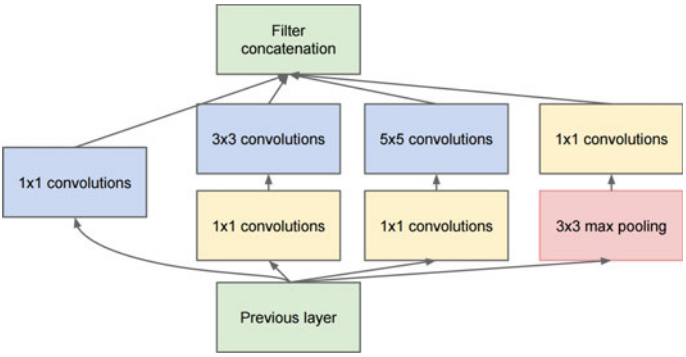
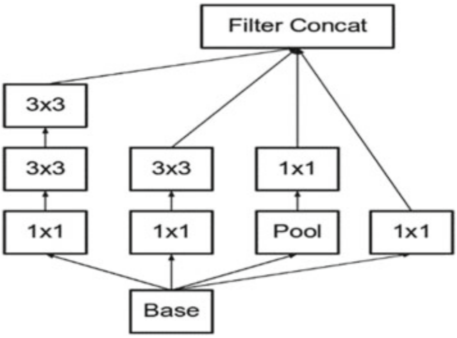


Fig. 9 Inception module with dimension reduction [18]

Fig. 10 Inception module after convolution and filter factorization. Here, the left-most 5 × 5 convolution of the old inception module is now represented as two blocks of 3 × 3 convolutions [17]



5.2 *Inception V2 (2015)*

Inception v2 is the upgradation of inception V1. In inception module decrease in dimensions causes loss of information, which is known as representational bottle neck. Inception Version 2 was built considering the problem of Representational Bottleneck. This occurred mostly with very deep convolutional networks. With each layer, the size of image was reduced by a fraction, resulting in a smaller image. Hence, after every level, the information we receive from an image is reduced [16]. To retain the spatial information, smart factorization method is used which made inception v2 more accurate and more efficient in terms of computational complexity. In factorization method, the 5 × 5 convolution was broken down into two 3 × 3 convolution, as two 3 × 3 convolutions were faster and more cheaper to compute as compared to a single 5 × 5 convolution. Factorization procedure is illustrated below (Fig. 10).

5.3 *Inception VersionV3 (2015)*

It is one of the modern architectures that attains a new state of art in terms of accuracy on ILSVRC image classification standard [19]. Inception V3 is the first Runner Up for image classification in ILSVRC 2015 [19]. To increase the speed and efficacy of inception module, asymmetric factorization is done, i.e., the $n \times n$ (3 × 3) convolutions were broken down into two convolutions $1 \times n$, and $n \times 1$. This method proved to be 0.3 times faster and approximately 33% cheaper with the same hardware resources (Fig. 11).

To reduce Representational Bottleneck, the deepness of model was reduced and whole model was made wider. Multiple convolutions were shifted to the same level to make sure image size does not get extremely small, rendering the image useless (Fig. 12).

Inception Net v3 assimilated all features of Inception v2, and in addition, it uses asymmetrical factorization, dropout in auxiliary classifiers, and label smoothening.

Fig. 11 Inception module after filter size reduction [17]

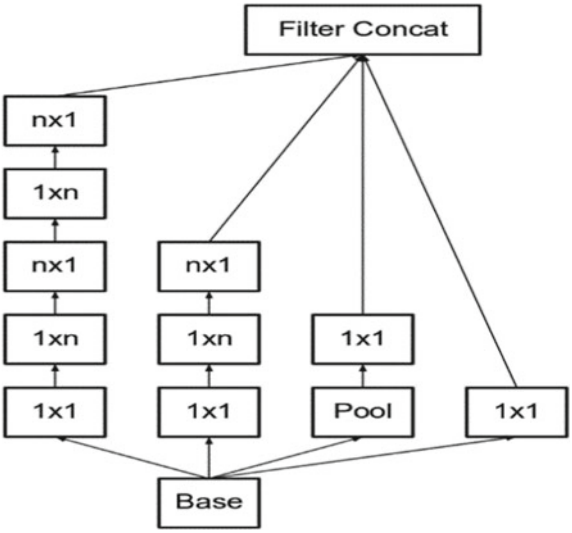
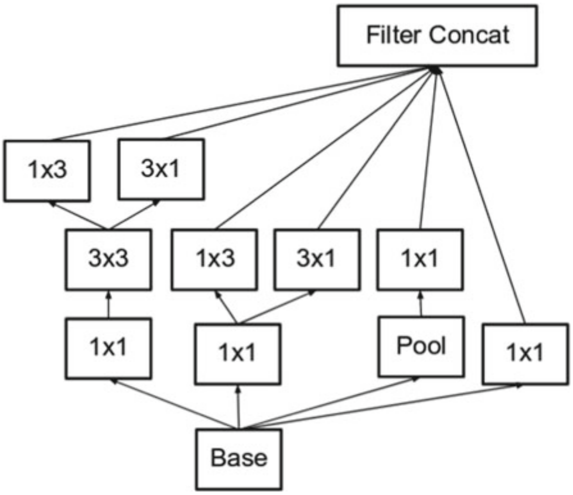


Fig. 12 Inception module using asymmetric factorization [17]



Both auxiliary classifier and label smoothening act as regularizer and help to reduce overfitting. Inception V3 is 42-layer deep architecture but uses fewer parameters and complexity is similar to VGG Net. This version was able to perform 7 × 7 convolutions as well, compared to the last version which was able to only perform till 5 × 5 convolution.

5.4 Inception V4-Inception-Resnet(2016)

Inception V4 and Inception ResNet are explained together. Cost and design structure of "Inception-ResNet-v2" roughly matches with the Inception-v4 network. There are minute difference of step time and training time. Inception-Resnet is faster in terms of practice and training time. As the name suggests, Inception-Resnet is the hybrid combination and stimulated by the performance of the residual network. The idea of Inception network with residual connections is proposed by Microsoft ResNet. It outperforms in a similar way the expensive pure Inception network works. In ILSVRC classification task this hybrid network has achieved 3.08% error [20].

Idea of this collaboration came from the findings that residual connections are innately vital for training the very deep networks. As inception modules are deep-layered networks so by replacing the filter concatenation stage of inception model with residual block, we can allow Inception to enjoy the gains of the Resnet and simultaneously intact to computational efficiency [16].

In inception-ResNet, the initial operations performed before Inception blocks were modified to make this model much more uniform. These operations were known as the stem. The overall scheme for pure inception V4 network and detailed composition of stem configuration of for inception Resnet is shown in Fig. 13.

This model introduced special blocks known as reduction blocks. These provided the user with the ability to change the width and height of the block making the model more tuneable and thus easier to test. The inception blocks are very flexible and modifications can be done on a lot of parameters including filter sizes (Figs. 14 and 15).

6 ResNet

Deep residual networks were implemented shortly after Google's Inception V3. The Inception was based on increasing width by developing shallower network to reduce overall error. In DCNN, going deep means ability to solve more complex task and significant improvement in object recognition capability. But, as we go further deeper down the network, the neural network experiences more difficulty in training the network (slow and tedious) and degradation and saturation in accuracy. Residual learning developers or technicians to take care of these issues. Developers of residual network were trying to reduce data degradation while maintain the depth of the network. To maintain the depth of network Residual blocks were introduced in the network.

Resnet(Residual network) developed by kaiming He et al., introduces new technology called residual learning. ResNet Won ILSVRC challenge 2015 and COCO 2015 competition in ImageNet Detection localization, Coco detection and segmentation. ResNet makes use of special*skip connections* and batch normalization [21].

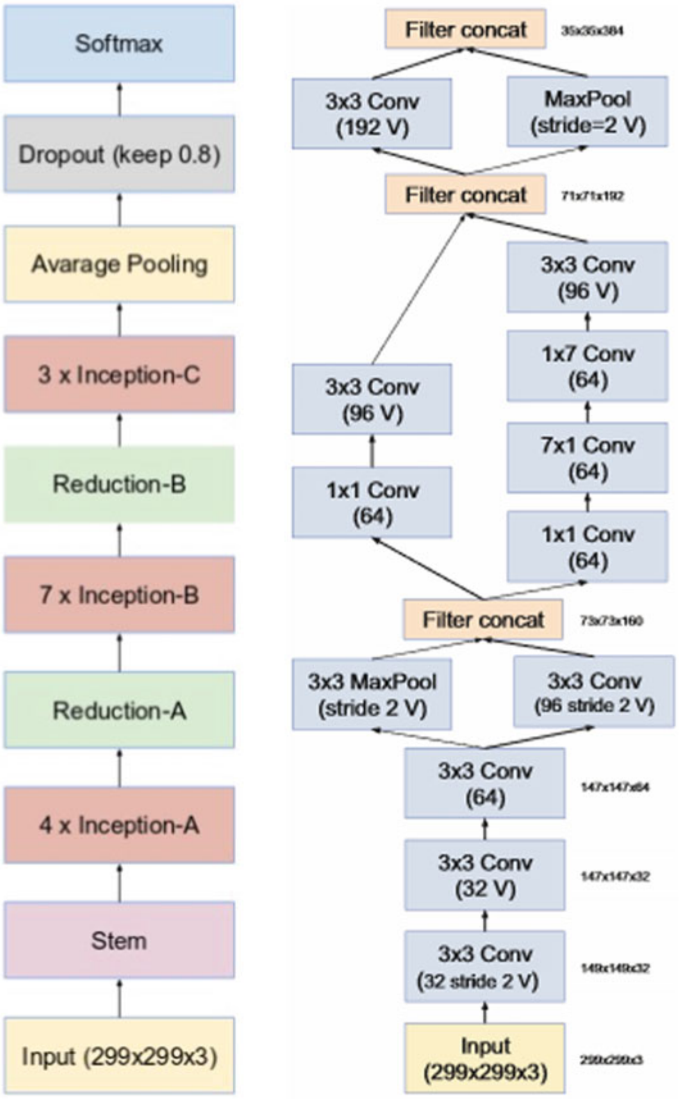


Fig. 13 Overall scheme for pure inception V4 network and detailed composition of stemconfiguration of for inception Resnet [20]

The main idea is: learning the differences or changes of the transformation is simpler than learning the transformation directly. So, in residual learning, instead of learning features at the end of the layer, the network learns some residual. Residual is the subtraction of feature gained from input of former layer. ResNet does this by utilizing identity shortcut connections (directly connecting input of nth layer to some $(n + x)$th layer), i.e., skipping one or few layers. Due to short and skip structure

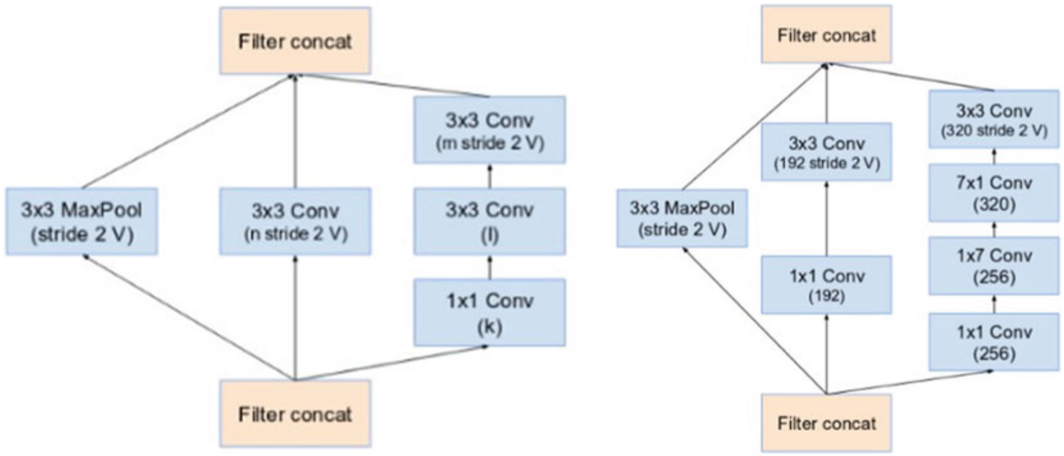


Fig. 14 Reduction block A (35 × 35–17 × 17 size reduction) and reduction block B (17 × 17–8 × 8 size reduction) [20]

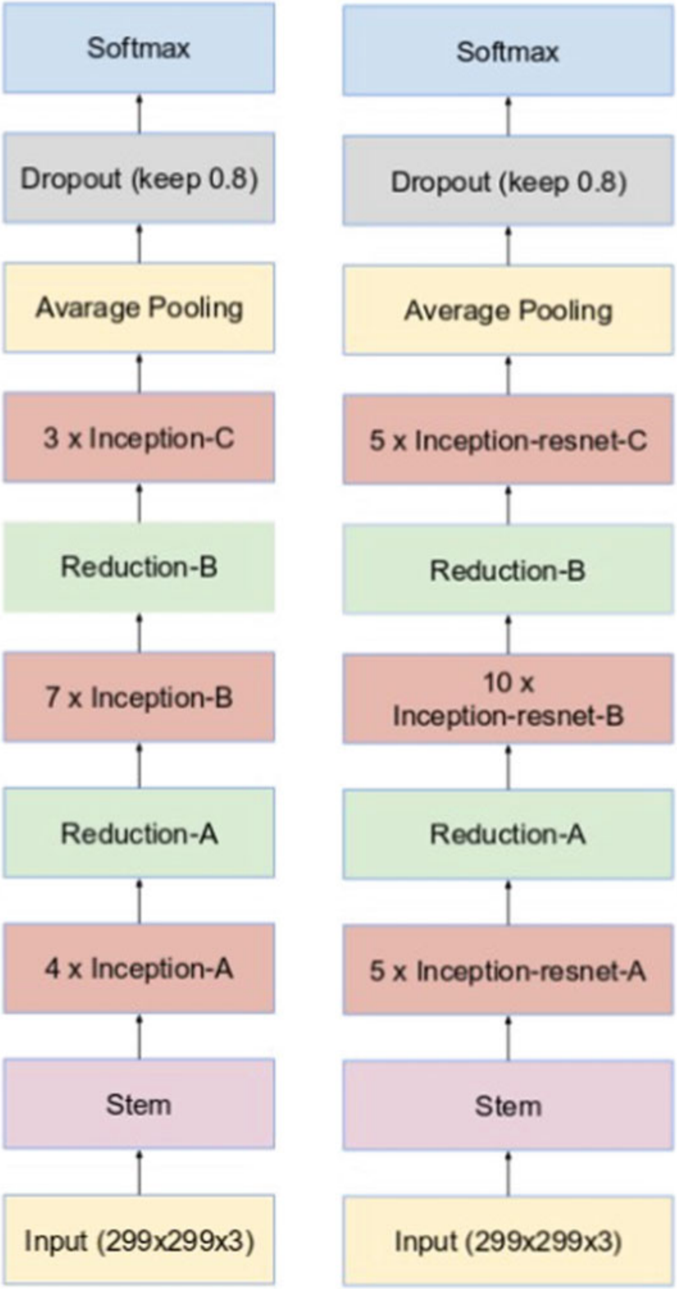


Fig. 15 Layout for unception V4 and layout for inception Resnet [20]

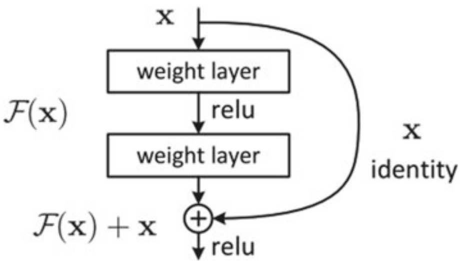

x
weight layer
$\mathcal{F}(\mathbf{x})$
relu
weight layer
x
identity
$\mathcal{F}(\mathbf{x}) + \mathbf{x}$
relu

Fig. 16 Residual learning: a building block [22]

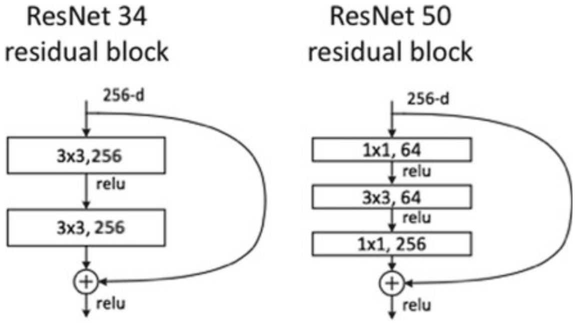


Fig. 17 A deeper residual function. Left: ResNet 34 block). Right: a bottleneck building block for ResNet 50 [23]

training this residual networks is easy. It can efficiently train networks with 100 layers and 1000 layers with lower complexity. The short cut connection is shown in Fig. 16.

ResNet block is either two-layer or three-layer deep. (two-layer network used for ResNet 18, 34 and three layer used for ResNet 50, 101, 152 Fig. 17).

In ResNet 50, the two-layer residual block is replaced with a three-layer bottleneck block. This block used 1 × 1 convolution to reduce computation required for a 3 × 3 convolution. This model consisted of 25 M parameters.

7 ResNext

ResNext, the winner of the ImageNet Large Scale Visual Recognition (ILSVRC) 2017, is the extension of ResNet and inception model. It is also inspired from the VGG. ResNext uses a homogeneous, multi-branch architecture for image classification. As ResNet and ResNext both follow the split-transform-merge paradigm. The main difference is in Inception module is that, the output from preceding layers is depth concatenated, but in ResNext, the outputs of different paths are merged by adding them together. ResNext performs point-wise grouped convolutional(1 × 1), which divides its input into groups of feature maps and performs normal convolution, respectively; their outputs are depth-concatenated and then fed to a 1 × 1 convolutional layer (In ResNext all paths share same topology).So in ResNext, convolution

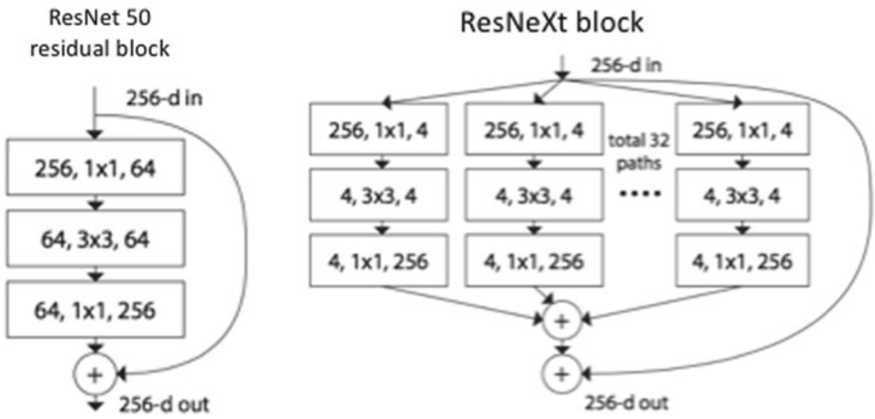


Fig. 18 ResNet-50 block vs. ResNeXt block [25]

is performed on lower-dimensional representations and later merged to provide the result, instead of performing convolution on complete feature map [24] (Fig. 18).

According to research conducted by *SainingXie,* it was found that residual networks provide better optimization, while aggregated transformations (in Inception Networks) provide stronger representations. This made ResNeXt one of the most powerful convolutional networks of its time, as it combined best of both the networks providing us with a much deeper and wider network.

8 DENSENET:(Dense Convolution Network)

Convolution networks perform better (in terms of accuracy, efficacy and training time) when they are deeper and contain shorter connections between layers closer to input as well as output. In DenseNet, the layers are connected in feed-forward manner rather than a traditional direct connection. For each layer, the feature maps of the respective layer and all forgoing layers are used as input to all succeeding layers which led a colossal increase in the number of connection. Hence, dense net has $L(L + 1)/2$ direct connections. The feature maps are aggregated with depth-concatenation which preserve the features, increase the variance of the outputs, and encouraging feature reuse. Figure below illustrates this layout schematically [26] (Fig. 19)

For each layer, all the feature maps for its previous layers are used as input, and its own output is used as input for all the upcoming layers. This significantly improved the performance of the network as fewer numbers of parameters were required and the direct connection to each layer solved the vanishing gradient dispute. Because of $L(L + 1)/2$ connections in an L-layer network, this architecture shows dense connectivity hence named as dense convolutional network (DenseNet).

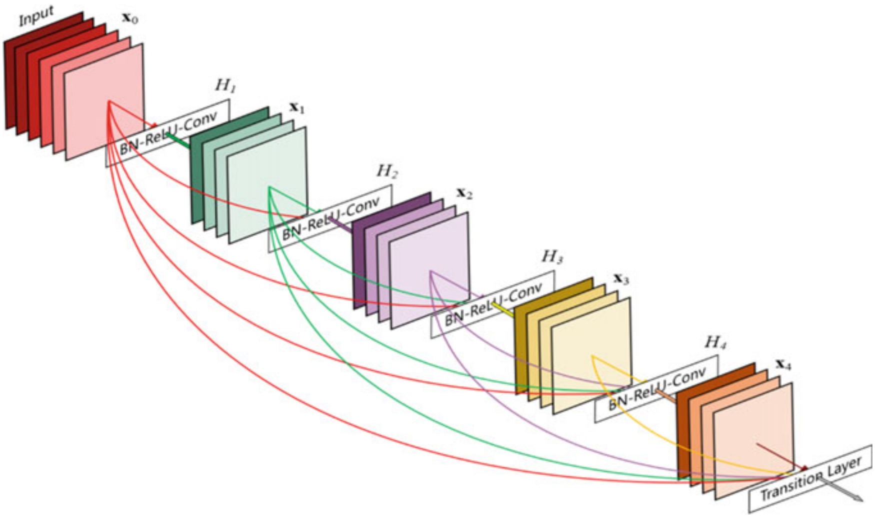


Fig. 19 5-layered densely connected network, every layer takes input as feature map from the former layer [26]

9 Conclusion

Convolutional neural networks (back bone of numerous deep learning algorithms) have shown state-of-the-art performance in high-level computer vision tasks. To the best of our knowledge, in this paper, we try to include and cover the literature which focuses on most advanced deep learning architectures. A brief introduction of classical neural net models has been included to give the necessary background knowledge about the subject in interest. Literature of most recent and complex architectures like inception, denseNet, googLe Net, etc., have been included to give the reader a detailed explanation about deep learning techniques. All views and findings of experts have been mentioned year wise to intact the interest and curiosity of the readers. The general discoveries show that CNN establishes a promising procedure with high grade performances in terms of accuracy, precision, and classification. However, the accomplishment of each convolutional neural network model is profoundly reliant on the nature of the informational collection used.

References

1. Simonyan K, Zisserman A (2015, April 10) Very deep convolutional networks for large-scale image recognition. Visual Geometry Group, Published as a conference paper at ICLR 2015, University of Oxford
2. Maind S, Wankar (2014, January) Research paper on basic of Artificial Neural Network. Int J Recent Innovation Trends in Comput Commun 2(1): 96–100. ISSN: 2321-8169

3. Rosenblatt F (1958) The perceptron: a probabilistic model for information storage and organization in the brain. Psychol Rev 65(6):386–408. https://doi.org/10.1037/h0042519
4. Gorach T (2018,July 07) Deep convolutional neural networks-a review. Int Res J Eng Tech (IRJET) 05(07)
5. Dickson B (2018, March 01). The limits and challenges of deep learning. Retrieved from https://bdtechtalks.com/2018/02/27/limits-challenges-deep-learning-gary-marcus
6. Yamashita R, Nishio M, Do RK, Togashi K (2018) Convolutional neural networks: an overview and application in radiology. Insights into Imaging 9(4):611–629. https://doi.org/10.1007/s13244-018-0639-9
7. Joost van Doorn, Van J (2014 June 23) Analysis of deep convolutional neural network architectures, 21th Twente Student Conference on IT Enschede, The Netherlands. Copyright 2014
8. Rawat W, Wang Z (2017) Deep convolutional neural networks for image classification: a comprehensive review. Neural Comput 29(9):2352–2449. https://doi.org/10.1162/neco_a_00990
9. Jeremy Jordan (2018, October 20) Common architectures in convolutional neural networks. Retrieved from https://www.jeremyjordan.me/convnet-architectures/
10. Voulodimos A, Doulamis N, Doulamis A, Protopapadakis E (2018) Deep learning for computer vision: a brief review. Comput Intel Neurosci, Hindawi 2018:1–13. https://doi.org/10.1155/2018/7068349
11. LeCun Y, Bottou L, Bengio Y, Haffner P (1998, November) Gradient-based learning applied to document recognition. Proc IEEE 86(11): 2278–2324
12. Pechyonkin M (2018, October 01) Key deep learning architectures: LeNet-5—max pechyonkin—medium. Retrieved from https://medium.com/@pechyonkin/key-deep-learning-architectures-lenet-5-6fc3c59e6f4
13. Krizhevsky A, Sutskever I, Hinton GE (2017) Image net classification with deep convolutional neural networks. Commun ACM 60(6):84–90
14. Deshpande A (2016, August 24) The 9 deep learning papers you need to know about (Understanding CNNs Part 3). Retrieved from https://adeshpande3.github.io/The-9-Deep-Learning-Papers-You-Need-To-Know-About.html
15. Zeiler MD, Fergus R (2014) Visualizing and understanding convolutional networks. Comput Vision—ECCV 2014 Lecture Notes in Computer Science, 818–833. https://doi.org/10.1007/978-3-319-10590-1_53 Copyright Springer International Publishing Switzerland
16. Raj B (2018, May 29) A simple guide to the versions of the inception network. Retrieved from https://towardsdatascience.com/a-simple-guide-to-the-versions-of-the-inception-network-7fc52b863202
17. Szegedy C, Vanhoucke V, Ioffe S, Shlens J, Wojna Z (2016) Rethinking the inception architecture for computer vision. 2016 IEEE Conf Comput Vision and Pattern Recognit (CVPR). https://doi.org/10.1109/cvpr.2016.308
18. Szegedy C, Liu W, Jia Y, Sermanet P, Reed S, Anguelov D, Rabinovich A (2015 Sep) Going deeper with convolutions. 2015 IEEE Conf Comput Vision Pattern Recognit (CVPR). https://doi.org/10.1109/cvpr.2015.7298594
19. Alemi A (216, August, 31.mproving inception and image classification in tensor-flow. Retrieved from https://ai.googleblog.com/2016/08/improving-inception-and-image.html
20. Sergey L, Szegedy C, Vanhoucke L, Vincent V, Alex A (2016, August 23) Inception-v4, Inception-ResNet and the impact of residual connections on learning. Comput Vision Pattern Recognit (cs.CV) Retrieved from https://arxiv.org/abs/1602.07261
21. He K, Zhang X, Ren S, Sun J (2016) Identity mappings in deep residual networks. Comput Vision—ECCV 2016 Lect Notes Comput Sci 630–645. https://doi.org/10.1007/978-3-319-46493-0_38
22. He K, Zhang X, Ren S, Sun J (2016) Deep residual learning for image recognition. 2016 IEEE Conf Comput Vision Pattern Recognit (CVPR). https://doi.org/10.1109/cvpr.2016.90
23. Xie S, Girshick R, Dollar P, Tu Z, He K (2017) Aggregated residual transformations for deep neural networks. 2017 IEEE Conf Comput Vision Pattern Recognit (CVPR). https://doi.org/10.1109/cvpr.2017.634

24. Hitawala S (2018) Evaluating ResNeXt model architecture for image classification. Comput Vision Pattern Recognit. Retrieved May 09, 2018, from https://arxiv.org/abs/1805.08700
25. Fung V (2017, July 16) An overview of ResNet and its variants—towards data science. Retrieved from https://towardsdatascience.com/an-overview-of-resnet-and-its-variants-5281e2f56035
26. Huang G, Liu Z, Maaten LV, Weinberger KQ (2017) Densely connected convolutional networks. 2017 IEEE Conf Comput Vision Pattern Recognit (CVPR). https://doi.org/10.1109/cvpr.2017.243