International Journal of Engineering Science Invention Research & Development; Vol. IV, Issue VII, JANUARY 2018
www.ijesird.com, E-ISSN: 2349-6185

CONVOLUTIONAL NEURAL NETWORK: AN OVERVIEW

[1]Priyanka R. Gulhane, [2]Manisha Thakkar
[1,2]*Department of Information Technology., MITCOE, Pune, India*
[1]priyanka.gulhane@mitcoe.edu.in, [2]manisha.thakkar@mitcoe.edu.in

***Abstract*—In recent years, deep artificial neural networks is swiftly emerging in pattern recognition and machine learning. Convolutional Neural Networks (CNNs) is one of the most remarkable forms of Artificial Neural Network (ANN) architecture. This paper focuses on Convolutional neural networks and its visualization technique. Compared to other techniques Convolutional neural networks (CNNs) are mostly used in image and pattern recognition problems since they have a number of advantages compared to other techniques. This paper emphasizes on the fundamental of CNNs, functioning and description of the different layers used in Convolutional neural networks.**

***Keywords*—deep learning, artificial neural network, pattern recognition.**

I. INTRODUCTION

Artificial Neural Networks (ANNs) are computer simulations, which are inspired by the operation of biological nervous systems (i.e the human brain). ANNs consist of neurons which are a large number of interconnected computational nodes. It is a three layered feed forward neural network (FNN).It consisting of input layer, a hidden layer and an output layer. This structure is the basis of a number of common ANN architectures.

Convolutional Neural Networks (CNNs) are similar to traditional ANNs as they consist of neurons. These neurons self-optimize through learning. Each neuron will receive an input and perform operation such as a scalar product which is followed by a non-linear function. The entire network will express a single perceptive score function i.e the weight from the input raw image data to the final output of the class score. The last layer holds the loss functions associated with the classes.

Convolutional Neural Networks is a type of Neural Networks which is very effective in area of image processing. Convolutional Networks have been found successful in classification, identifying objects, faces and text processing etc. Convolutional Networks are growing as an important tool and becoming popular amongst machine learning practitioners today. The primary objective of this paper is to cultivate an understanding of application of Convolutional Neural Networks on images.

Using the algorithm of backpropagation deep learning detects detailed structure in the large datasets and specifies how a particular machine should change its internal parameters/specifications which are used to compute the representation of each layer from the the previous layer. Deep convolutional neural networks have brought the dramatic improvement in processing various images, video, speech and audio.

Typical deep neural network architectures include deep belief networks (DBNs) (2), deep Boltzmann machines (DBMs) (3), SAEs (4), and stacked denoising AEs (SDAEs) (5). The layer-wise training models have a bunch of alternatives such as restricted Boltzmann machines (RBMs) (6), pooling units (7), convolutional neural networks (CNNs) (8), AEs, and denoising AEs (DAE) (4). In this paper, we adopt one of the above deep learning models, CNNs.

II. BACKWARD PROPOGATION

Artificial neural network can be trained by Backward Prorogation of Errors. It is a supervised training scheme. In Supervised training scheme, it learns from the labeled training data.

In the network, initially all the edge weights are assigned randomly. Then in the training dataset for every input, the artificial neural network is activated and its output is observed. This actual output is then compared with the desired output which we already know, and then the error is calculated. The calculated error is then "propagated" back to the previous layer. Then the weights are "adjusted" accordingly to minimize the error. These steps are repeated until the output error goes below a predetermined threshold. After termination of above algorithm, we can say that, we have a "learned" network. This network can work with "new" inputs. This artificial neural network is said to have learned from several labeled data and from its error propagation.

III. CONVOLUTIONAL NEURAL NETWORKS(CNNS)

A neural network is a system of interconnected processing elements, called as “neurons” that exchange the information between each other. Layers are made up of a number of interconnected 'nodes' which contain an 'activation function'. Patterns are sent to the network through the 'input layer', which then communicates to one or more 'hidden layers'. Actual processing is done in the hidden layer through system of weighted connections. The hidden layers are connected to the 'output layer' where the result is output. As shown in Figure 1, the first layer is the input layer which accepts the input, the second layer is the hidden layer where actual processing is done and third layer is output layer which generates output.

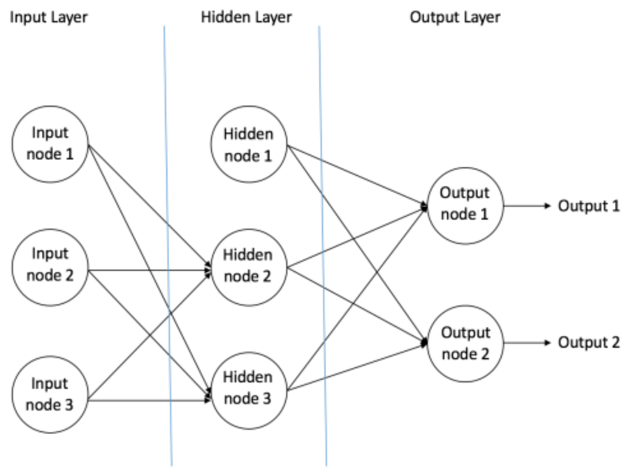


Figure 1: Feedforward neural network

There are four main operations in the Convolutional Neural Network:

1. Convolution
2. Rectified Linear Unit
3. Down sampling or Pooling
4. Classification

These four operations are the basic building blocks of every Convolutional Neural Network. So understanding how these operations work is an important for understanding of Convolutional Neural Network.

1. Convolution

The convolution operation is used to extracts features of an input image. In convolution operation, the first layer of convolution extracts low-level features. Low level features are generally gradients, edges, lines, and corners of an input image whereas higher-level layers of this operation extract higher-level features.

Convolution of an input image with one filter/kernel produces one output feature, and with K kernels/filters independently produces K features. Convolution operation works by moving the each filter/kernel left to right beginning from top-left corner of the input image, one element at a time. The filter is moved in a downward direction by an element when the top-right corner of the image is reached, and again the filter is moved in left to right manner, again one element at a time. This process is continual till the filter/kernel reaches the bottom-right corner of an image. For example, when the image is of 30 x 30 i.e. n=30 and filter/kernel is 5 x 5 i.e. K=5, then for these positions, the resultant matrix of features in the output will contain 26 x 26 elements (i.e., (n-K+1) x (n-K+1)).

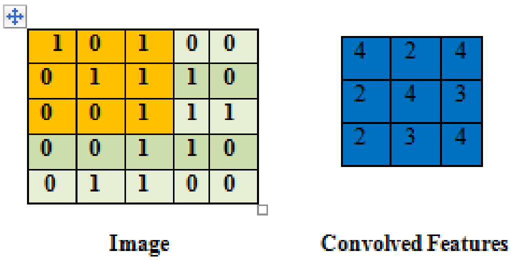
Consider for example, a 5 x 5 image with pixel values only 0 and 1

1	0	1	0	0
0	**1**	**1**	**1**	**0**
0	**0**	**1**	**1**	**1**
0	**0**	**1**	**1**	**0**
0	**1**	**1**	**0**	**0**

Considering another 3 x 3 filter matrix as shown below:

1	0	1
0	1	0
1	0	1

Then, the Convolution of 5 x 5 image matrix and 3 x 3 filter matrix can be computed as shown below in **Figure 2**:



1	0	1	0	0
0	1	1	1	0
0	0	1	1	1
0	0	1	1	0
0	1	1	0	0

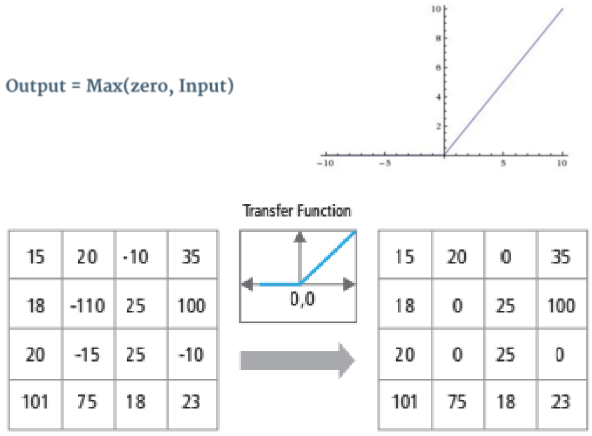
4	2	4
2	4	3
2	3	4

Figure 2: Convolution operation. The output matrix is Convolved Feature or Feature Map.

Here, the 3×3 matrix is called a '**filter**' or 'kernel' or 'feature detector' and the matrix formed by sliding the filter over the image and computing the dot product is called the 'Convolved Feature' or 'Activation Map' or the '**Feature Map**'. It is important to note that filters acts as feature detectors from the original input image. For the same image different values of the filter matrix will produce different Feature Maps.

2.RECTIFIED LINEAR UNIT

ReLU is Rectified Linear Unit and it is a non-linear operation. and its output is given as shown below:



15	20	-10	35
18	-110	25	100
20	-15	25	-10
101	75	18	23

15	20	0	35
18	0	25	100
20	0	25	0
101	75	18	23

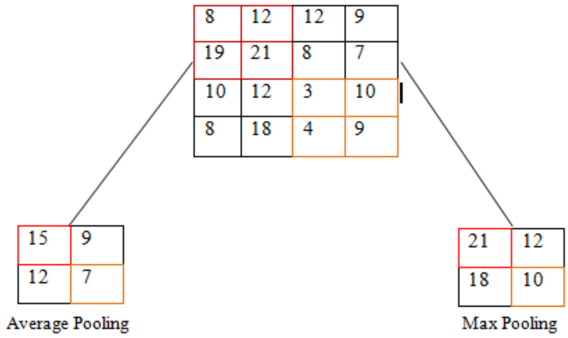
Figure 3: The ReLU (Rectified Linear Unit) operation

In ReLU operation the input and output size of the image is same. ReLU is an activation function. It can be computed very efficiently as compared to other activation functions like sigmoid and hyperbolic tangent. It is used to add non linearity to the network. ReLU increases the nonlinear properties of the overall network without disturbing the receptive fields of the convolution layer. ReLU trains the network many times faster when compared to the other non-linear functions used in CNNs,.

3.THE POOLING STEP

Pooling makes the features robust against scaling and noise. Sub-sampling/down-sampling i.e. the pooling operation layer reduces the resolution of the feature matrix. There are two types of pooling namely max pooling and average pooling. The input is divided into non-overlapping two-dimensional spaces in both mentioned pooling types.

Considering an example as shown in Figure 4 for average pooling, the average value of the four values in the region are calculated and for max pooling, the highest value of the four values is selected.



8	12	12	9
19	21	8	7
10	12	3	10
8	18	4	9

15	9
12	7

21	12
18	10

Figure 4: Representation of max pooling and average pooling

Pooling reduces the size of the input representation (9). To be specific, pooling

- Reduces the size of input image and makes it even more manageable.
- Controls overfitting by reducing the number of parameters and computations in the network, [9]
- Makes the network more robust against small transformations and distortions like scaling, translation therefore distortion in input image will not affect the output of pooling layer because we take the maximum / average value.

4.FULLY CONNECTED LAYER

The fourth layer of CNN is fully connected layers of neurons. Every neuron from the previous layer is connected to each neuron on the next layer. High-level features of the input image can be obtained as an output from convolutional and pooling layers. These high-level features are used for classifying the input image into various classes based on the training dataset. It is also an inexpensive way of learning non-linear combinations of these features. Combination of the features from convolutional and pooling layers are better for classification instead using individual features from those layers.

IV. WHY CNN?

This section discusses the need and benefits of using CNN for image recognition.

1.INVARIANT TO DISTORTIOSNS AND SMALL TRANSFORMATIONS IN IMAGE

CNN is robust as it is invariant to small transformations and distortions. These transformations or distortions may be due to change in shape, camera lens position, different poses, and different conditions of lighting, partial occlusions. As same weight configuration is used across space CNNs are shift invariant also.

2. LESS MEMORY REQUIREMENT

Memory requirement is reduced in CNN as the same coefficients are used across different locations in the space. Considering the hypothetical case where we use a fully connected layer to extract the features, the input image matrix of size 32x32 and considering a hidden layer having 1000 features. It will require 106 coefficients which is a huge memory requirement. CNN is memory efficient.

3. EASIER AND BETTER TRAINING

In CNN training time is reduce by reducing the number of parameters as compared to standard neural network. In standard CNN number of parameters would be much higher which will result in increased training time. In a CNN, as the number of parameters is reduced drastically and in same proportion training time is also reduced. We can design a standard neural network whose performance would be same as a CNN by taking into account training time. Standard neural network are equivalent to CNN due to increased number of parameters, as it brings noise during the training process performance of a standard neural network will be poor.

V. CONCLUSION

This paper describes the basic concepts and operations of Convolutional Neural Networks. This paper also explains the layers required to build the Convolutional Neural Networks and how to apply it for image analysis tasks.

Convolutional Neural Networks differ than other forms of Artifical Neural Network as it exploits the particular type of input instead of focusing on whole problem domain which makes CNN network architecture simpler.

The research in the area of image analysis using neural networks has somewhat reduced due to false belief surrounding the level of complexity and knowledge which is required to initiate forming of these powerful machine learning algorithms. Convolutional Neural Networks gives the good performance in problems such as pattern/image recognition.

REFERENCES

[1] O'Shea, Keiron, and Ryan Nash. "*An introduction to convolutional neural networks*." arXiv preprint arXiv:1511.08458 (2015).

[2] G. E. Hinton, S. Osindero, and Y. Teh, "*A fast learning algorithm for deep belief nets*," Neural Comput., vol. 18, no. 7, pp. 1527–1554, Jul. 2006.

[3] R. Salakhutdinov and G. E. Hinton, "*Deep Boltzmann machines*," in Proc. Int. Conf. Artif. Intell. Statist., Clearwater Beach, FL, USA, 2009, pp. 448–455.

[4] Y. Bengio, P. Lamblin, D. Popovici, and H. Larochelle, "*Greedy layer-wise training of deep networks*," in Proc. Neural Inf. Process. Syst., Cambridge, MA, USA, 2007, pp. 153–160.

[5] P. Vincent, H. Larochelle, I. Lajoie, Y. Bengio, and P. Manzagol, "*Stacked denoising autoencoders*," J. Mach. Learn. Res., vol. 11, no. 12, pp. 3371–3408, Dec. 2010.

[6] G. E. Hinton, "*A practical guide to training restricted Boltzmann machines*," Dept. Comput. Sci., Univ. Toronto, Toronto, ON, Canada, Tech. Rep. UTML TR2010-003, 2010.

[7] Y. LeCun et al., "*Backpropagation applied to handwritten zip code recognition," Neural Comput*., vol. 1, no. 4, pp. 541–551, Apr. 1989.

[8] K. Fukushima, "*Neocognitron: A self-organizing neural network model for a mechanism of pattern recognition unaffected by shift in position*," Biol. Cybern., vol. 36, no. 4, pp. 193–202, Apr. 1980.

[9] CS231n Convolutional Neural *Networks for Visual Recognition*, Stanford.