International Journal of Advanced Technology in Engineering and Science www.ijates.com
Volume No 03, Special Issue No. 01, May 2015 ISSN (online): 2348 – 7550

A REVIEW ON CONVOLUTIONAL NEURAL NETWORKS

B. Gnana Priya
Assistant Professor
Department of Computer Science and Engineering
Annamalai University

ABSTRACT

The traditional methods for solving computer vision problems depend on the feature extraction process. The introduction of Convolutional Neural Networks (CNN) has provided a way for automatically learning the domain specific features. Convolutional Neural Network is a special type of Neural Networks, which provides a better performance on several Computer Vision and Image Processing tasks compared to the traditional networks. Some of the areas of CNN applications are Image Classification and Segmentation, Object Detection, Video Processing, Natural Language Processing, and Speech Recognition. Due to the use of multiple feature extraction stages of CNN it can automatically learn representations from the data. The CNN needs a large amount of data and hardware resources for better performance.

Keywords: Convolutional neural networks, Convolution operation, Pooling

1. INTRODUCTION

Using CNNs for learning is becoming much popular due the ability of CNNs to learn feature extraction directly, they can be retrained for new recognition tasks, enabling us to build on pre-existing networks and produce state-of-the-art recognition results. A convolutional neural network can have hundreds of layers and each of them learn to detect different features of an image. Filters are applied to each training image and the output of each convolved image is used as the input to the next layer. The filters can start as very simple features, such as brightness and edges, and increase in complexity to features that uniquely define the object.

CNN is composed of an input layer, an output layer, and many hidden layers like other neural networks. These layers perform operations that alter the data with the intent of learning features specific to the data. A convolutional neural network is trained on hundreds or time to train a model. Three types of layers are used here: Convolutional Layer**,** Pooling Layer and Fully-Connected Layer. Convolutional layer forms the basic building block and uses kernels to detect features all over the image. The Kernels carries out a convolution operation which is an element-wise product and sum between two matrices.

Pooling layers are inserted between convolutional layers to reduce the parameters and computation in the network. It resizes the input and prevent overfitting of network. These operations are repeated over hundreds of layers with each layer learning to identify different features. Filters are applied to each training image at different resolutions, and the output of each convolved image is used as the input to the next layer. After learning features in many layers, a fully connected layer that outputs a vector that contains probabilities of each class that the network will be able to predict is used. The final layer of the CNN architecture uses a classification layer such as Softmax to provide the classification output.

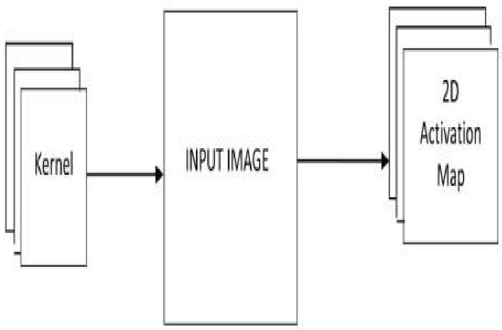
2. CNN ARCHITECTURE

CNNs are comprised of three types of layers. These are convolutional layers, pooling layers and fully-connected layers. When these layers are stacked, a CNN architecture has been formed. The basic functionality of the CNN can be broken down into four key areas. One the input layer, which receives the pixel values of the image fed as input. Second, the convolutional layer will determine the output of neurons of which are connected to local regions of the input through the calculation of the scalar product between their weights and the region connected to the input volume. The rectified linear unit --ReLu aims to apply an elementwise activation function such as sigmoid to the output of the activation produced by the previous layer.

Third, the pooling layer will then simply perform downsampling along the spatial dimensionality of the given input, further reducing the number of parameters within that activation. Fourth, the fully-connected layers will then perform the same duties found in standard ANNs and attempt to produce class scores from the activations, to be used for classification. It is also suggested that ReLu may be used between these layers, as to improve performance. Finally, output layer determines the output depending on the problem.

2.1 Convolutional Layer

Convolutional layers are the key elements of the network and play a vital role in the network operation. The layers in the network has learnable kernels. These kernels are normally very small in size and spreads along the entire image given as input. In a convolutional layer, the layer convolves each filter across the spatial dimensionality of the input to produce a 2D activation map. Each filter applied to the input image produces a separate activation function. This can be visualised, as in Fig 1. As we glide through the input, the scalar product is calculated for each value in that kernel. From this kernels, the network will learn specific feature at a given spatial position of the input. These are known as activations.



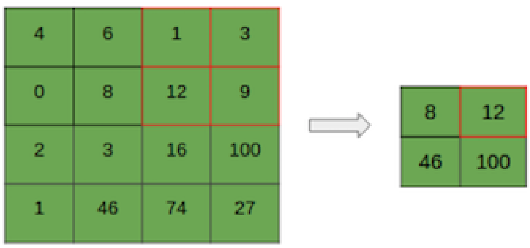
Fig(1) : Filters applied to Input image

2.2 Pooling Layer

Pooling layer reduces the dimensionality of the input given to it. This layer reduces the number of parameters and the computational

Volume No 03, Special Issue No. 01, May 2015 ISSN (online): 2348 – 7550

complexity of the model. The pooling layer operates over each activation map in the input, and scales its dimensionality using the Max function. In most CNNs, these come in the form of max-pooling layers with kernels of a dimensionality of 2 × 2 applied with a stride of 2 along the spatial dimensions of the input. The size of the kernel is dependent on the nature of the application. Due to the destructive nature of pooling, having a kernel size above 3 will usually greatly decrease the performance of the model. It is also important to understand that beyond max-pooling, CNN architectures may contain general-pooling. General pooling layers are comprised of pooling neurons that are able to perform operations like normalisation and average pooling. Fig(2) gives the results of maxpooling operation.



Fig(2): MaxPooling operation

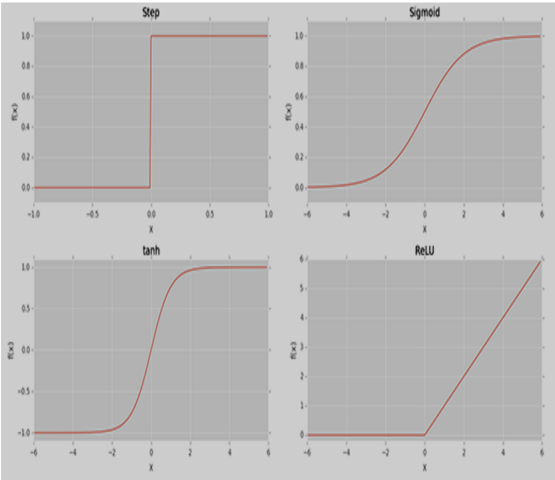
2.3 Fully Connected Layer

The fully-connected layer contains neurons of which are directly connected to the neurons in the two adjacent layers, without being connected to any layers within them. This is analogous to way that neurons are arranged in traditional forms of ANN.

3. ACTIVATION FUNCTION

Activation functions are important for our Network to learn and to give Non-linear complex functional mappings between the inputs and response variable. They introduce non-linear properties to the Network. If we do not apply a Activation function then the output signal would simply be a simple linear function and are limited in their complexity and have less power to learn complex functional mappings from data. A Neural Network without Activation function would simply be a Linear regression Model. High dimensional and non linear large datasets comprising of images, videos and audios requires complex architecture to extract the required knowledge from them. So it is essential for us to pick up the right activation function depending on the problem.

Most popular types of Activation functions are Sigmoid or Logistic, Tanh (Hyperbolic tangent), ReLu -Rectified linear units. Sigmoid Activation function is of form $f(x) = 1 / 1 + \exp(-x)$ **.** Its Range is between 0 and 1. It is a S shaped curve. It is easy to understand and apply. The major disadvantages being Vanishing gradient problem, output isn't zero centered. Hyperbolic Tangent function output is zero centered because its range in between -1 to 1. *ReL*u- Rectified Linear units is very popular because it was proved that it had 6 times improvement in convergence from Tanh function. It is very simple and efficient . Almost all deep learning Models use ReLu nowadays. But the limitation here is that it should be only used within Hidden layers of a Neural Network Model**.** Therefore for output layers we should use a Softmax function for a Classification problem to compute the probabilities for the classes , and for a regression problem it should simply use a linear function. Fig(3) depicts the various activation functions used in artificial neural networks.



Fig(3) :Activation function for Neural Networks

4. CONCLUSION

Today Convolutional neural networks are the driving force behind machine learning and computer vision applications like robotics, medical diagnostics, self driving cars, etc. This is because of their ability to work with few parameters and their simplicity. This paper briefly describes about the operation and the various parameters involved in training a convolutional neural network.

REFERENCES

1. Ciresan, D., Meier, U., Schmidhuber, J.: Multi-column deep neural networks for image classification. In: Computer Vision and Pattern Recognition (CVPR), 2012.
2. Cires¸an, D.C., Giusti, A., Gambardella, L.M., Schmidhuber, J.: Mitosis detection in breast cancer histology images with deep neural networks. In: Medical Image Computing and Computer-Assisted Intervention–MICCAI 2013.
3. Ciresan, D.C., Meier, U., Masci, J., Maria Gambardella, L., Schmidhuber, J.: Flexible, high performance convolutional neural networks for image classification. In: IJCAI Proceedings-International Joint Conference on Artificial Intelligence. vol. 22, 2011.
4. Cires¸an, D.C., Meier, U., Gambardella, L.M., Schmidhuber, J.: Convolutional neural network committees for handwritten character classification. In: Document Analysis and Recognition (ICDAR), 2011.
5. Egmont-Petersen, M., de Ridder, D., Handels, H.: Image processing with neural networks a review. Pattern recognition, 2002.
6. Hinton, G.: A practical guide to training restricted boltzmann machines. Momentum , 2010.
7. Ji, S., Xu, W., Yang, M., Yu, K.: 3d convolutional neural networks for human action recognition. Pattern Analysis and Machine Intelligence, IEEE Transactions, 2013.
8. Karpathy, A., Toderici, G., Shetty, S., Leung, T., Sukthankar, R., Fei-Fei, L.: Largescale video classification with convolutional neural networks. In: Computer Vision and Pattern Recognition (CVPR), 2014.
9. Krizhevsky, A., Sutskever, I., Hinton, G.E.: Imagenet classification with deep convolutional neural networks. In: Advances in neural information processing systems, 2012.
10. Nebauer, C.: Evaluation of convolutional neural networks for visual recognition. Neural Networks, IEEE Transactions, 1998.
11. Simard, P.Y., Steinkraus, D., Platt, J.C.: Best practices for convolutional neural networks applied to visual document analysis, IEEE, 2003.
12. Srivastava, N.: Improving neural networks with dropout. Ph.D. thesis, University of Toronto , 2013.
13. Szarvas, M., Yoshizawa, A., Yamamoto, M., Ogata, J.: Pedestrian detection with convolutional neural networks. In: Intelligent Vehicles Symposium, 2005.
14. Szegedy, C., Toshev, A., Erhan, D.: Deep neural networks for object detection. In: Advances in Neural Information Processing Systems, 2013.
15. Tivive, F.H.C., Bouzerdoum, A.: A new class of convolutional neural networks (siconnets) and their application of face detection. In: Neural Networks, 2003.