Image Classification

Nade Tade
Department of Computer Science
University of Texas at Dallas
Richardson,Texas
nxt180027@utdallas.edu

Rakesh Kumar Mahato
Department of Computer Science
University of Texas at Dallas
Richardson,Texas
rkm190000@utdallas.edu

Nimrat Bedi
Department of Computer Science
University of Texas at Dallas
Richardson,Texas
nxb200004@utdallas.edu

Abstract—***Deep Learning has emerged as a new area in Machine Learning and is applied to several signal and image applications. Image processing is one of the modern world's most researched topics. Convolutional Neural Network (CNN) has the capacity of training huge datasets with millions of parameters and convolve it with filters to produce the desired output. The main purpose of the work presented in this project is to apply the concept of a Deep Learning algorithm namely, Convolutional neural networks (CNN) in image classification. We experimented with CNN models built with Convolution layer, Relu Activation layer, Maxpooling layer, Softmax layer, and connected them to evaluate its performance. This algorithm is tested on the X-ray dataset of covid and normal images inspired by current situation of Covid happening in the world. The performance of the algorithm is evaluated based on the classification accuracy. The results of our trials are presented in the later section of this document.***

Keywords—Machine learning, Convolutional Neural Network, Image Classification, Deep learning

I. INTRODUCTION

Processing of images involves some basic operations namely image restoration/rectification, image enhancement, image classification, image fusion, etc. Image classification forms an important part of image processing. The objective of image classification is the automatic allocation of the image to thematic classes. Two types of classification are supervised classification and unsupervised classification.

The process of image classification involves two steps, training of the system followed by testing. The training process means, to take the characteristic properties of the images (form a class) and form a unique description for a particular class. The process is done for all classes depending on the type of classification problem, binary classification or multi-class classification. The testing step means categorizing the test images under various classes for which the system was trained. This assigning of the class is done based on the partitioning between classes based on the training features.

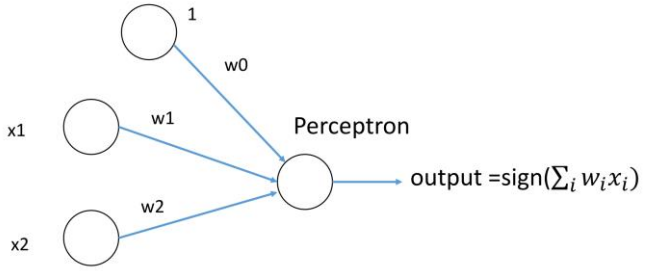
As the traditional neural network algorithms succumb to faster scene recognition and feature selection, the Convolutional Neural Network tends to give the better results in the performance of pattern recognition; from image processing to voice recognition. The latest image classification techniques are being applied to facial identification, self-driving cars, facial emotion detection, robotics, advertising and other fields.

CNNs use relatively little pre-processing compared to other image classification algorithms. This means that the network learns the filters that in traditional algorithms were hand-engineered. This independence from prior knowledge and human effort in feature design is a major advantage.

II. NEURAL NETWORK FUNDAMENTALS

A. Perceptron

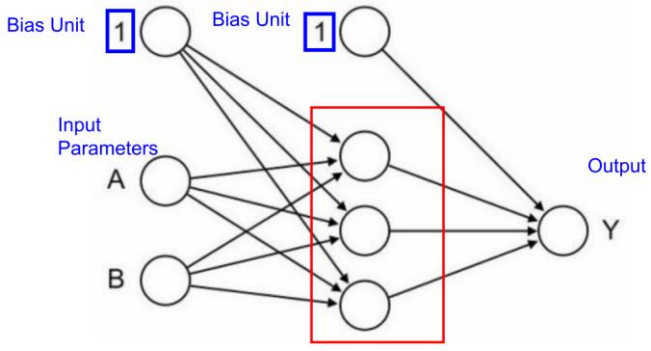
The fundamental of Convolution Neural Network starts with a single Perceptron. Fig 1 below shows a schematic of a simple perceptron:



The input parameters are passed through a layer of different weights. There might also be a bias factor depending on the requirement. All the collected data is then passed through an activation function which gives a weighted output for the input parameters.

B. Neural Network

The next level of Machine Learning is a Neural Network. It is a network of perceptrons. There is network of hidden layers in between the input parameters and the output data. Fig. 2 below shows a structure of Neural Network:



The output from the input layer is passed through all the hidden neurons (marked in a red box). There is an added weight parameter in each path. We try to give high weightage to the parameter with higher impact on the output. The output is usually a highly complex and non-linear function. This network is a combination of simple perceptrons; but together they form a very powerful classifier. The network is formed by keeping an emphasis on learning internal representations automatically.

III. CNN ARCHITECTURE

Convolution neural network includes an input layer and an output layer as well as multiple hidden layers. The IMAGE DATA will be passed through series of layers namely:

- Convolution Layer
- Activation function (We use Relu here)
- Max-Pooling Layer
- Flattening Layer
- Fully Connected Layer
- SoftMax Layer

Each of the above layers are explained in detail later in the document. There can be multiples of a set of layers of Convolution, Activation function and Pooling. The 3D data is then flattened to a 1D data. This is then passed through a Fully Connected layer (or sometimes called as densely connected layer). The final layer (SoftMax) will have many tuning parameters to classify Image with the probabilistic values ranging from 0 and 1.

Fig 1 shows a model of a Convolution Neural Network.

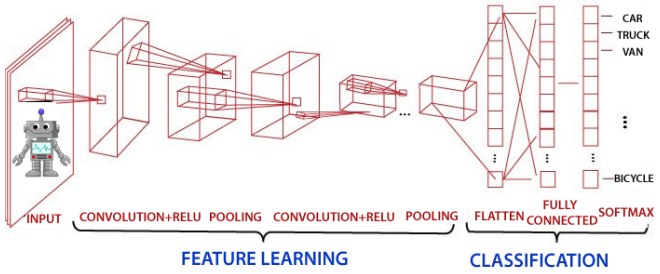


Fig1 Neural network with many layers to classify

A. *Convolution Layer*

Convolution is the first layer to extract features from an input image. Convolution preserves the relationship between pixels by learning image features using small squares of input data. It is a mathematical operation that takes two inputs such as image matrix and a filter or kernel.(As shown in Figure below)

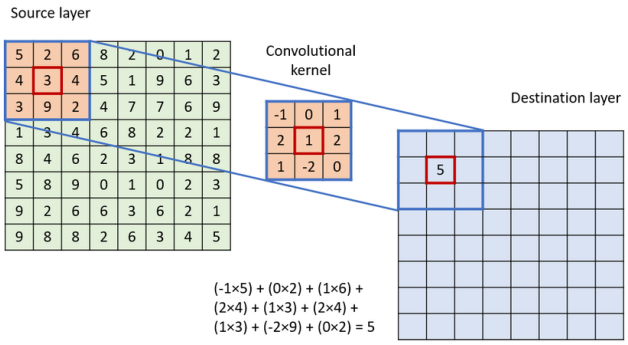


Fig 2 Convolution layer

A convolutional layer within a neural network should have the following attributes:

- Convolutional kernels defined by a width and height (hyper-parameters).
- The number of input channels and output channels (hyper-parameter).
- The depth of the Convolution filter (the input channels) must be equal to the number channels (depth) of the input feature map.

In the convolution layer, there are two other factors that are important, namely ***Stride*** and ***Padding***.

Stride is the unit of steps taken by the filter from one block of cells to another. Example: If an Image of size (NxN) is passed through a filter of size (FxF), and the stride is kept 1 and padding =0; the output matrix will be of size ((N-F+1)x(N-F+1)).

The output matrix after filter is bit smaller than the original matrix. Here the padding is utilized. Padding adds cells on the outer of the previous matrix.

B. *ReLu Activation Function*

ReLU stands for Rectified Linear Unit. It is used for a non-linear operation to convolved feature map. The ReLu will be computed after convolutional layer to remove negative values from the activation map without affecting the fields of hidden layer.
The output is $f(x) = max(0,x)$.

C. *Max Pooling*

This layer section would reduce the number of parameters when the images are too large. Spatial pooling also called subsampling or down sampling is the one that reduces the dimensionality of each map but retains important information. This layer also reduces the computational complexity of neural network thereby controlling overfitting.

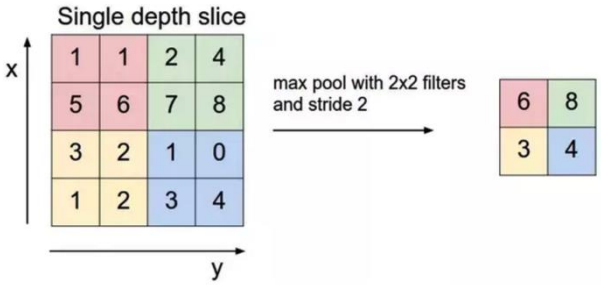


Fig 3 Max Pooling

D. *Flatten Layer*

Flatten layer is present between Convolution Layer and Fully Connect Layer. It converts the 3D output matrix to one dimensional matrix that can be fed into fully connected neural network.
Example: if the output after the final Max-pooling layer is 4x4x128; the flatten layer will convert it to a 1D matrix of size 2048.

E. *Fully Connected Layer*

In this layer, we flattened our matrix into vector and feed it into a fully connected layer like a neural network. It takes the output of previous layer and finds which feature correlates to a particular class.

In the below diagram, the feature map matrix will be converted as vector (x1,x2,x3…..). With fully connected layers, we combined these features together to create a model.

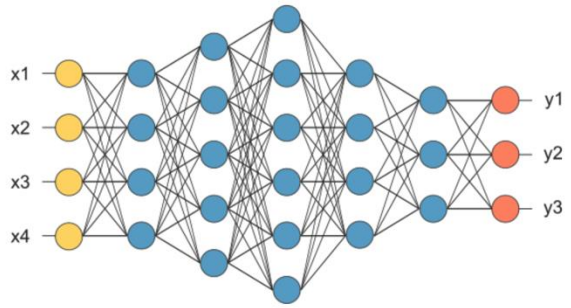


Fig 3 After pooling layer, flattened as Fully Connected Layer

E. *Softmax Layer*

Softmax layer is typically the final output layer in a neural network that performs multiclass classification. The name comes from the Softmax function that takes an input a number of scores values(), and squashes them into values in the range between 0 and 1 whose sum is 1. Softmax gives us the tendency or likelihood of the object to belong to a class. It is used for the mapping of non-normalized output to probability distribution on predicted classes. Softmax should be used when the classes are independent that is they are mutually exclusive.

IV. IMPLEMENTATION

CNN is supervised deep learning approach which requires large labelled data for training on the network. After training the model will learn the weights and the accuracy of the classifier is improved. Weights are updated during backward passes. We have also used learning rate and bias during the implementation.

A. *Image Dataset Preprocessing*

We have used Covid dataset which contains chest x-rays of normal and covid positive cases. For Covid positive dataset we have taken data from GitHub (https://github.com/ieee8023/covid-chestxray-dataset); for the Normal dataset we have taken dataset from Kaggle (https://www.kaggle.com/paultimothymooney/chest-xray-pneumonia). The data from Kaggle has normal and pneumonia dataset. We extracted the normal dataset and used it for our coding. We have hosted the images in www.imgur.com. We are fetching the image urls from imgur and converting it into json file. The json file is then hosted to https://jsonbin.io. We are finally getting the images by fetching the json file using the json bin api.

We took images from each class in training dataset. The training dataset has 180 images. The images were of different sizes so we resized all images to 94x94x3. The 3 is for RGB colors.

B. *Training*

The initial data after preprocessing is passed through a Convolution Layer with 16 filters of size 3x3x3. After convolution, the matrix is of size 92x92x16, which is passed through a Relu activation function. We get a data of the same size. The data is then passed through a Max-Pool Layer of 2x2. After this, the matrix is of size 46x46x16. The same process of Convolution-Relu-Maxpool is repeated 4 times. The training phase is shown in the table below:

Matrix of image size through different layers

				# of filters
Input	94	94	3	
Convolution through Filter	3	3	3	16
After Convolution	92	92	16	
Relu Activation	92	92	16	
Maxpool	2	2		
After Maxpool	46	46	16	
Convolution through Filter	3	3	16	32
After Convolution	44	44	32	
Relu Activation	44	44	32	
Maxpool	2	2		
After Maxpool	22	22	32	
Convolution through Filter	3	3	32	64
After Convolution	20	20	64	
Relu Activation	20	20	64	
Maxpool	2	2		
After Maxpool	10	10	64	
Convolution through Filter	3	3	64	128
After Convolution	8	8	128	
Relu Activation	8	8	128	
Maxpool	2	2		
After Maxpool	4	4	128	
After Flatten Layer	2048			
After Fully Connected Layer	64			
Relu Activation	64			
After Fully Connected Layer	2			
Output (After Softmax)	2			

This gives us the classification of our images

For the convolution layer, we have used a stride of 1 and padding of 0 units. The weights are then learned through backpropagation and are updated into the weight matrix.

C. Testing

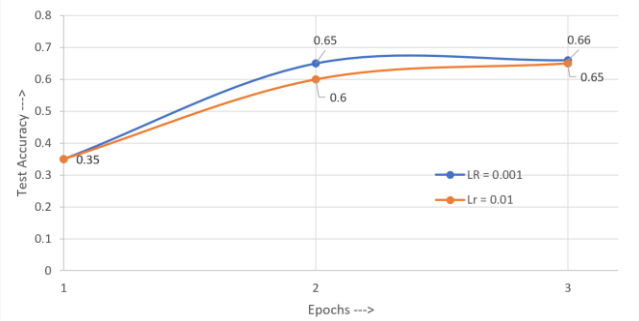
For the testing part we have used 45 images that we run through our algorithm. The testing images are preprocessed first. Then they are each passed through the weights to classify into normal or covid positive cases.

D. Result and Analysis

For the complete dataset, we ran the code on Google collab and took different trials with change in learning rate. The results are shown below:

Learning Rate	0.05	0.025	0.005	0.001
Training Accuracy	0.6	0.6	0.4	0.75
Testing Accuracy	0.67	0.67	0.33	0.73

The complete dataset was too much time consuming, so we took 30% of the data to check for variation with respect to learning rate and epochs. Below graph shows the results:



E. Conclusion

This project was interesting. We could learn about the concepts on CNN and it's practical implementation. We could train our dataset for classification of patients as covid positive or normal based on their chest x-ray images. There is still a lot of scope of improvement. The results can be improved by taking more data and tweaking with learning rate and the number of epochs of training. If the algorithm is made robust, this will be a huge success for the Machine Learning engineers as a contribution towards healthcare sector.

REFERENCES

[1] Yushi Chen, Hanlu Jiang, Chunyang Li, Xiuping Jia, "Deep feature extraction and classification of Hyperspectral images based on Convolutional Neural Network (CNN)" IEEE Transactions on Geoscience and Remote Sensing, vol. 54, no. 10, pp. 6232-6251, 2016.

[2] An introduction to convolutional neural networks [Online]available at:http://white.stanford.edu/teach/index.php/An_Introduction_to _Convolutional_Neural_Networks.

[3] Deepika Jaswal, Sowmya.V, K.P.Soman "Image Classification using convolutional neural networks," in International Journal of Advancements in Research & Technology, vol. III, issue 6, june-2014, 1661-1668

[4] Y. LeCun, B. Boser, J. S. Denker, D. Henderson, R. E. Howard, W. Hubbard, and L. D. Jackel, "Backpropagation applied to handwritten zip code recognition," Neural Computation, vol. 1, no. 4, pp. 541–551, Dec 1989.

[5] Understanding of CNN- Deeplearning [Online]available at: https://medium.com/@RaghavPrabhu/understanding-of-convolutional-neural-network-cnn-deep-learning-99760835f148

[6] Machine Learning by Thomas M. Mitchell; Publisher: McGraw-Hill, Inc. Professional Book Group 11 West 19th Street New York, NY, United States

[7] Lecture Notes by Dr. Anurag Nagar, Department of Computer Science, Erik Jonsson School of Engineering & Computer Science, The University of Texas at Dallas