**Image Classification Using**

**CNN**

**Literature Report**

**Machine Intelligence**

**By**

**Name: Gautam R A SRN: PES1UG20CS151**

**Name: Jayasimha S SRN: PES1UG20CS177**

**Name: Gurukiran V SRN: PES1UG20CS156**

**Paper 1:**

**Title - Image Classification using Convolutional Neural Networks**

By Muthukrishnan Ramprasath Sr. Assistant professor, Department of Computer Science & Engineering, Madanapalle Institute of Technology & Science, Andhra Pradesh M. Vijay Anand Professor, Department of Computer Science and Engineering Saveetha Engineering College, Chennai Shanmugasundaram Hariharan Professor, Department of Computer Science and Engineering, Saveetha Engineering College, Chennai, India

Link: <https://acadpubl.eu/hub/2018-119-17/4/419.pdf>

**Abstract**

In recent year, with the speedy development in the digital content’s identification, automatic classification of the images became most challenging task in the fields of computer vision. Automatic understanding and analysing of images by system is difficult as compared to human visions. Several researches have been done to overcome problem in existing classification system, but the output was narrowed only to low level image primitives. However, those approach lack with accurate classification of images. In this paper, our system uses deep learning algorithm to achieve the expected results in the area like computer visions. Our system present Convolutional Neural Network (CNN), a machine learning algorithm being used for automatic classification the images.

**Introduction:**

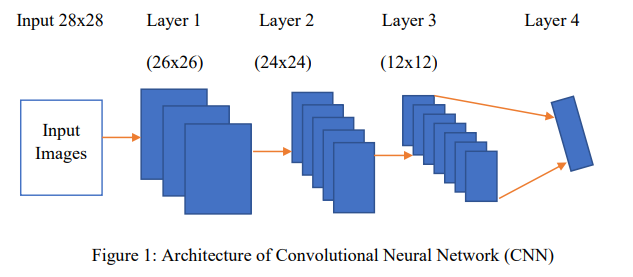
Due to the explosive growth of digital content, automatic classification of images has become one of the most critical challenges in visual information indexing and retrieval systems. Image classification is a big problem in computer vision for the decades. In case of humans the image understanding, and classification is done very easy task, but in case of computers it is very tricky and exhaustive task. In general, each image is composed of set of pixels and each pixel is represented with different values. Henceforth to store an image the computer must need more spaces for store data. To classify images, it must perform higher number of calculations. For this it requires systems with higher configuration and more computing power. For this purpose, CNN (Convolutional Neural Network), a Deep Learning algorithm is being used. The CNN basically extracts the features of the respective images and based on the extracted features classifies the images into their respective classes.

**Proposed Architecture for image classification using CNN:**

The dataset that the authors have used here is mentioned below.

1. **Dataset used: MNIST dataset**
2. **Working of the model:**

Initially, the human will train classifier to obtain the desired pattern from the images. Then the images classified with help of the pattern precisely obtained from the previous stages. The obtained results will vary with respect to the patterns observed and it is completely dependent on the knowledge of the person who classifies. The model created here has used deep learning architecture (i.e. CNN) for classification of images also it uses different layers in Convolutional Neural Network (CNN) to extract new feature form the images datasets. The figure explains the components of CNN networks.



Their system uses grayscale images as input image having 28x28 sizes. The first layer in CCN applied 32 filters on input images, each image size is 3x3 producing 32 feature maps of size 26x26. The second layer is applying 64 filters, each of size 3x3 producing 64 feature maps of size 24x24. Max pooling layer is act as third layer which is used to down sampling the images to 12x12 by using subsampling window of size 2x2. The layer 4 is fully connected layer having 128 neurons and uses sigmoid activation function for classification of images and produce the output image.

1. **Building of the Conv Nets:**

Convnet is a sequence of layers and every layer of convnet transforms one volume of activations to another using a differentiable function.

Most important types of layers are used to build ConvNet architecture which will be called as Convolutional layer, Pooling layer and Fully connected layer.

These layers will be stacked to form a full ConvNet architecture. Input image will give as input to the layer it holds the raw pixels values of the input images.

Conv layer compute the output of neurons which are connected to local regions in the input and each neuron performs the computation of dot product between their weights and a small region they are connected to in the input volume.

RELU layer leaves the input volume unchanged as such if 28x28x1 is given as input volume, the output volume would be 28x28x1. It will apply an elementwise activation function like max (0, x) thresholding at zero.

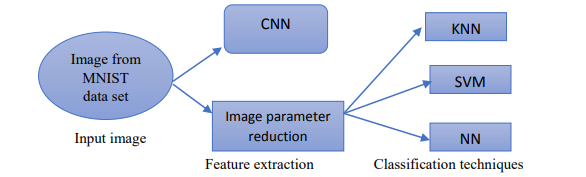
In pooling layer dimensions of the image will be reduced but the information of the image is retained. The down sampling operation is performed along the spatial dimensions (width, height) that is if the input is 24x24x64 then the output volume would be 12x12x64.

Finally, fully connected layer computes the class scores resulting in volume of size 1x1x10 where each of the 10 numbers correspond to a class score such as among the 10 categories given as input. convolutional neural networks will transform the original image layer by layer from the original pixel values to the final class scores.

1. **Implementation of the proposed system (Algorithm Used):**

The proposed system here uses CNN for implementation purpose. Convolutional Neural Networks are very similar to ordinary Neural Networks, that are made up of neurons that have learnable weights and biases. Every neuron performs dot product by receiving some input and using bias it follows non-linearity. The whole convent still expresses a distinct score function, from the raw pixels on one end to class scores at the other end.

They have a loss function like SoftMax on the last layer which is fully connected layer. As the inputs are images to convent, it allows to encode certain properties in architecture. These properties make the forward function more efficient to implement and vastly reduce the number of parameters in the network. The mail goal of the image classification able to extract the feature from raw images



Algorithm

1. Batch size =128, no of classes 10, number of epochs = 5,
2. Dimension of input image 28 ×28
3. Loading the input images from MNIST data set
4. Variable exploration: X=test data set (10000,28,28,1), Train data set (60000,28,28,1)
5. Creating and compiling the models
6. Training the network.
7. Testing the trained network on the test case inputs.

**Inference:**

In this paper, we used Convolutional Neural Networks (CNN) for image classification using images form hand written MNIST data sets. This data sets used both and training and testing purpose using CNN. It provides the accuracy rate 98%. Images used in the training purpose are small and Grayscale images. The computational time for processing these images is very high as compare to other normal JPEG images. Stacking the model with more layers and training the network with more image data using clusters of GPUs will provide more accurate results of classification of images. The future enhancement will focus on classifying the colored images of large size and its very useful for image segmentation process.

**Paper 2:**

**Title: Image Classification Using Convolutional Neural Network**

By P. Lakshmi Prasanna, D. Raghava Lavanya, T.Sasidhar , B.Sekhar Babu ,Department of Computer Science and Engineering, KoneruLakshmaiah Education Foundation, Vaddeswaram, AP, India

Link: http://www.warse.org/IJETER/static/pdf/file/ijeter308102020.pdf

**Abstract:**

In recent years, research related to images is a challenging task as there are very few techniques which can be used for classification of images. Several researches took place to overcome the drawbacks in image classification, but the output was limited to the basic low-level picture. There are many deep neural network techniques used for image classification like Convolutional Neural Network, Deep Belief Network, and Machine Learning Algorithms like SVM, Random Forest and many. In this paper we want to implement image classification using CNN. CNN is a type of the deep neural networks, most frequently used for visual imaging analysis. CNN is implemented through multilayer perceptron that follows a hierarchical model that works on network building and finally delivers to fully connected layer. In this layer all the neurons are connected, and the output is processed.

**Introduction:**

In computer vision, most of the times the information is in the form of non-textual form, such as images. Due to the presence such a huge number of images the image databases also increased. As a result, we face complex task of organizing and accessing vast amounts of images that are available. Image classification helps in solving such tasks. Convolutional Neural Networks (a deep learning algorithm) is being used here for the purpose of classification. In this paper they have used CIFAR-10 dataset. This dataset contains 60000 training images of size 32\*32 in 10 classes likely airplanes, cats, cars, deer, birds, frogs, dogs, ships, horses and trucks. Out of which 50000 are training images and the remaining 10000 are testing images. Figure 4 is the output of the CNN for Cifar-10 with Epoch size 2

**Proposed Method:**

The dataset that is used in this paper is mentioned below.

1. **Dataset Used:** CIFAR – 10 Dataset

Convolutional neural Network has hidden layers, Known as Convolutional layers which make CNN more effective for image analysis.

CNN layer types mainly include three types

• Convolutional layer

• Pooling layer

• Fully connected layer

When a computer sees image, it converts the image into an array of pixel values depending on the image resolution and size. They have considered only images of type jpg and size be 480 x 480. Then it is converted to 480 x 480 x 3 image where the represents the RBG values. To describe the intensity of the pixel, they are given numbering from 0 to 255. Further the array with numbers are given as input to the image classification.

1. **Different Layers of the CNN and its role in classifying the images:**

**Convolutional Layer:** Convolutional Layer is most important part of image classification. The main task in this layer is extracting features from the input image. Conv layer consists of many feature maps. The neuron of same feature map is used in extracting regional characteristics of various positions in the former surface. But for single neuron, its extraction is regional feature of the same positions in the former separate feature map. The results in the Conv layers are passed to nonlinear Activation function like sigmoid, tanh, ReLu.

**Pooling Layer:** A problem with the output of the Conv layer is that they are sensitive to the location of the features in the input. One idea to reduce the sensitivity is that we can decrease its dimensionality i.e., down sampling. Pooling layer is used to decrease the dimensions of the feature map. There are two types of common pooling techniques that can be used to decrease the dimensionality. They are max pooling and the average pooling. In max pooling, calculating the max value of each patch in the feature map. Whereas average pooling, finding the average of each patch in the feature map.

**Fully Connected layer:** The task of the fully connected layer is to connect the output of the previous layer. There is no spatial arrangement in this layer. There can be many fully connected layers where the last layer is connected to the output layer. One of the most commonly used method is soft regression because of its performance.

**Inference:**

For image classification we need a system that itself can extract features efficiently and classify them. We used Convolutional Neural Network (CNN) for image classification which contains Convlayers to extract features and max pooling to decrease the size of image thus classifies the image accurately. Whereas for other techniques like SVM, K-Means Feature Extraction needed explicitly to be done. We implemented this using CIFAR-10 dataset in python.

**Paper 3:**

# **Title – Hyperspectral Image Classification Using Convolutional Neural Networks and Multiple Feature Learning**

By Qishuo Gao, Samsung Lim, Xiuping Jia

Link: https://www.mdpi.com/2072-4292/10/2/299/htm

**Abstract:**

Convolutional neural networks (CNNs) have been extended to hyperspectral imagery (HSI) classification due to its better feature representation and high performance, whereas multiple feature learning has shown its effectiveness in computer vision areas. This paper proposes a novel framework that takes advantage of both CNNs and multiple feature learning to better predict the class labels for HSI pixels. The authors have built a novel CNN architecture with various features extracted from the raw imagery as input. The network generates the corresponding relevant feature maps for the input, and the generated feature maps are fed into a concatenating layer to form a joint feature map. The obtained joint feature map is then input to the subsequent layers to predict the final labels for each hyperspectral pixel. The proposed method not only takes advantage of enhanced feature extraction from CNNs, but also fully exploits the spectral and spatial information jointly. The effectiveness of the proposed method is tested with three benchmark data sets, and the results show that the CNN-based multi-feature learning framework improves the classification accuracy significantly.

**Introduction:**

Hyperspectral imagery (HSI) has been widely used in the remote sensing community in order to take advantage of the composition of hundreds of spectral channels over a single scene. However, HSI demands robust and accurate classification techniques to extract the features from the image. The classification of HSI has been considered as a particularly challenging problem due to the complicated nature of the image scene (i.e., a large amount of data, mixed pixels and limited training samples), and therefore many attempts have been made to address this issue in the last few decades. In the early stage of HSI classification, spectral domain classifiers, such as support vector machines (SVMs), random forest (RF), and multinomial logistic regression (MLR), have made great improvements in understanding the image scenes.

Very recently, deep learning is of interest to researchers in the field of computer vision. In particular, convolutional neural networks (CNNs) have attracted a lot of attention due to their superior performance in many domains, such as face recognition, object detection and video classification. In terms of feature extraction, CNNs can learn feature representations through several convolutional blocks. In contrast to the traditional rules-based feature extraction methods, CNNs can learn features automatically from the original images. Moreover, CNNs can be designed as an end-to-end framework that can produce classification maps directly. Therefore, many CNN models have been applied to HSI classification.

**Proposed Methodology/Implementation:**

The dataset used by the authors to train the convolutional neural network is mentioned below

1. **Dataset used for training the CNN: Hyperspectral image Dataset**

The first step of this framework is the extraction of multiple HSI features followed by several CNN blocks. Given T sets of features, each individual CNN block will learn the corresponding representative feature map, and all the feature maps will be jointed by a concatenating layer. The weight and bias for each block are fine-tuned in this network through back propagation. The output of the network for each pixel is a vector of class membership probability with C units, corresponding to C classes defined in the hyperspectral data set.

1. **Working of the model:**

CNN blocks for different features were designed to have the same architecture. There are three convolutional layers, pooling layers, ReLU layers and concatenating layers. The input images are initially normalized into [−1 1]. The number of kernels in each convolutional layer is set as 200 empirically. The input neighbourhood of each feature is set as 5 × 5, 7 × 7 and 9 × 9 for the Indian Pines data set, the University of Pavia data set and the Salinas data set, respectively. The learning rate for CNN models is set as 0.01; the number of epochs is set as 100 for the Indian Pines and the University of Pavia data sets, and 150 for the Salinas data set. The batch size is set as 10. To quantitatively validate the results of the proposed framework, overall accuracy (OA), average accuracy (AA) and the Kappa coefficient (k) are adopted as the performance metrics. Each result is shown as an average of ten times repeated experiments with the randomly chosen training samples.

Once the CNN has been trained and tested, verification of its effectiveness is done using three benchmark datasets namely

1. Airborne Visible/Infrared Imaging Spectrometer (AVIRIS) data set: Indian Pines
2. Reflective Optics System Imaging Spectrometer (ROSIS) data set: University of Pavia, Italy.
3. AVIRIS data set: Salinas

**Inference:**

In order to prove the potential of CNNs for HSI classification, the authors have presented a framework consisting of a novel CNN model. The framework was designed to have several individual CNN blocks with comprehensive features as input. To enhance the learning efficiency as well as to leverage both the spatial contextual and spectral information of the HSI, the output feature maps of each block are then concatenated and fed into subsequent convolutional layers to derive the pixel label vectors. By using the proper architecture, the built network is a shallow but efficient one, and it can concurrently exploit the interactions of different spectral and spatial contextual information by using the concatenating layer. In comparison with the CNN-based single feature learning method, the classification results are improved significantly with multiple features involved. Moreover, in contrast to the traditional rule-based classifiers, the CNN-based framework can extract the deep features automatically and in a more efficient way.

Moreover, the experiments suggest that a three-layer CNN is optimal for HSI classification, and the neighbourhood size between 2 × 2 to 6 × 6 can balance the efficiency and complexity of the network. The pooling layer with a size of 2 × 2 and 200 kernels in each layer can provide an enough capacity for the network. Since the training samples are very limited in HSI classification, the multiple input feature maps and ReLU in the proposed network can help alleviate the overfitting phenomenon and accelerate convergence. The tests with three benchmark data sets showed superior performances of the proposed framework. As CNNs are gaining attention due to the strong ability in extracting the relevant features for image classification, the proposed method is expected to provide various improvements for the better feature representation purpose.

**Paper 4:**

**Image Classification Using Convolutional Neural Networks**

**Deepika Jaswal, Sowmya.V, K.P.Soman**

**Year:2014**

Reference: [Image Classification Using Convolutional Neural Networks (ijser.org)](https://www.ijser.org/researchpaper/image-classification-using-convolutional-neural-networks.pdf)

**Introduction**

The main purpose of the work presented in this paper, is to apply the concept of a Deep Learning algorithm namely, Convolutional neural networks (CNN) in image classification. The algorithm is tested on various standard datasets, like remote sensing data of aerial images (UC Merced Land Use Dataset) and scene images from SUN database.

**Abstract**

The input to the network is a 2D image. The network has input layer which takes the image as the input, output layer from where we get the trained output and the intermediate layers called as the hidden layers, together the layers produce an approximation of input image data’s.The performance of the algorithm is evaluated based on the quality metric known as Mean Squared Error (MSE) and classification accuracy. The graphical representation of the experimental results is given on the basis of MSE against the number of training epochs. The experimental result analysis based on the quality metrics and the graphical representation proves that the algorithm (CNN) gives fairly good classification accuracy for all the tested datasets.

**Working of algorithm**  
CNNs exploit spatially local correlation by enforcing a local connectivity pattern between neurons of adjacent layers. In the CNN algorithm, each sparse filter is replicated across the entire visual field. These units then form a feature maps, these share weight vector and bias. CNN also make use of the concept of max-pooling, which is a form of non-linear down-sampling. In this method, the input image is partitioned into non-overlapping rectangles. The output for each sub-region is the maximum value.

**Inference**

Face vs Non – Face Image Classification , the data used in this experiment are taken from a face and skin detection database, 1000 face images and 1000 nonface images are used for training and testing the network. The images are of 32 × 32 pixels. From the variation of MSE for 200, 300 and 500 epochs , we infer that the MSE reduces considerably as the epoch value changes from 200 to 300. However, the MSE value undergoes minimum variation, as we change the number of epochs from 300 to 500. A better trained image will result in lower MSE with the original image. As the value for Mean Squared Error (MSE) tends to decrease, the variation in the final reconstructed output and the original image is very less. MSE indicates the close proximity between underlying true image and the final reconstructed output. The idea here is to use enough number of epochs that would result in low MSE, high classification accuracy and with least duration for training the network.

**Paper 5:**

**Classification Using K Nearest Neighbour for Brain Image Retrieval P.A.Charde, S.D.Lokhande**

**YEAR:2013**

Reference: <https://www.ijser.org/researchpaper/Classification-Using-K-Nearest-Neighbor-for-Brain-Image-Retrieval.pdf>

**Introduction**

The proposed algorithm is for the retrieval of the most visually similar images to a given query image from a database of medical images by content. In this algorithm we take shape feature extraction by canny Edge detection and texture feature extraction by using Gabor filter. Gabor filter is best feature extraction method for texture. And on the basis of these feature, medical images are classified using KNN method. The retrieval performance of the proposed system is tested using large medical image database of about 500 computed tomography images of brain. The retrieval performance and retrieval complexity are measured and evaluated.

**Abstract**

K-Nearest Neighbour classification technique is the simplest technique conceptually and computationally that provides good classification accuracy. The K-NN algorithm is based on a distance function and a voting function in KNN , the metric employed is the Euclidean distance. The K-NN has higher accuracy and stability for MRI data than other common statistical classifiers, but has a slow running time. The k-nearest neighbour classifier is a conventional nonparametric supervised classifier that is said to yield good performance for optimal values of k. Like most guided learning algorithms, K-NN algorithm consists of a training phase and a testing phase.

**Working of Algorithm**

In the training phase, data points are given in a n-dimensional space. These training data points have labels associated with them that designate their class. K-NN algorithm comprises of following stages:

1. Determine a suitable distance metric.

2. In the training phase: Stores all the training data set P in pairs (according to the selected features) 3. During the test phase: Computes the Distances between the new feature vector and all the stored features (training data).

4. The k nearest neighbours are chosen and asked to vote for the class of the new example.

The correct classification given in the test phase is used to assess the correctness of the algorithm. If this is not satisfactory, the k value can be tuned until a reasonable level of correctness is achieved.

**Inference**

The proposed method has training and classification phases. In training phase, from a given set of training images the shape and texture features are extracted and used to train the ­­ system using the KNN classifier. In classification phase a given test ct of brain image pre-processed and then texture features are extracted for classification. These features are queried to KNN classifier to label an unknown image. Pre-processing includes conversion of Input image into gray scale image and noise removal. In this work median filter used for de-noising. The purpose of feature extraction is to reduce the original data set by measuring certain properties of features that distinguish on input pattern from another.

**Paper 6:**

**Classification Of Medical Image Data Using K Nearest Neighbour And Finding The Optimal K Value**

**Preeti Nair, Indu Kashyap**

**Year:2020**

Reference: 1. <https://www.ijstr.org/final-print/apr2020/Classification-Of-Medical-Image-Data-Using-K-Nearest-Neighbor-And-Finding-The-Optimal-K-Value.pdf>

**Introduction**

One of the most widely employed algorithm is, KNN which serves better for any classification purpose. This paper proposes a finding of optimal k value using k NN on medical data. For a possible set of data folds, the achieved optimal k value dictates the effectiveness of the research objectives.

**Abstract**

Initially, a set of medical images are collected from a public repository. The collected images composed of irrelevant noises, low contrast with degraded quality. Gaussian filtering is employed to reduce the noise with enhanced image quality. The contrast enhanced image is processed under morphological operations in segmentation process. Feature extraction process play a vital role for selecting the optimal k values. Using Gray Level Co-Occurrence Matrix (GLCM), the relevant features are estimated and selected. These selected features are given as input to k NN algorithm and obtains the optimal k value based on accuracy.

**Working of Algorithm.**

Foremost step that helps to meet the requirements of the research objectives. A set of medical images related to brain tumour and breast cancer are been collected from a public dataset repository. Data Pre-processing This process assist to remove the irrelevant details of an image. Here, resizing sampling and Gaussian filtering are used for achieving better pre-processed image. Gaussian filters are the type of smoothing filters that operates on estimated weights with respect to Gaussian function. The contrast-enhanced image is then examined using morphological operations which is a kind of processing the images based on their shapes. Dilation and Erosion are the two fundamental morphological operations which operates in similar basis. Dilation inserts pixels to boundaries of an object whereas erosion eliminates the pixels on object boundaries. Rom the region detected image, Gray Level Cooccurrence Matrix (GLCM) is a matrix that depicts spatial distribution of grey levels in binary image. The spatial relationship between two neighbouring pixels can be specified in many ways with different offsets and angles, the default one being between a pixel and its immediate neighbour to its right.

**Inference.**

Experimental analysis of the proposed workflow. Initially, a possible set of medical images, related to, brain tumour and breast cancer are collected from a public dataset repository. The acquired images of 256\*256 are subjected to irrelevant noises which is improved by using Gaussian Filtering. Here, noise level of 0.01 is applied over the images. This process will blur the images and reduce contrast. The low-level contrast image degrades the edges of images for segmentation process, represent the contrast-enhanced image which is used for morphological segmentation model. By doing so, the small objects are easily recognized for segmentation process. With help of segmented image, the relevant features are extracted using GLCM model. The obtained features are contrast, correlation, energy, homogeneity, mean, standard deviation, entropy, Root mean square, variance, smoothness, kurtosis, skewness, inverse difference moment are all estimated. The medical data of 200 samples collected from public repository has been observed by dividing into training and testing data. The training data are modelled using pre-processing, segmentation and feature extraction process. Finally, the trained given for each sample and thus obtain accuracy, sensitivity and specificity. The results state that when k is 7, the data emits maximal accuracy.

**Paper 7:**

CNN-RNN: A Unified Framework for Multi-label Image Classification Jiang Wang1 Yi Yang1 Junhua Mao2 Zhiheng Huang3∗ Chang Huang4∗ Wei Xu1 1Baidu Research 2University of California at Los Angles 3Facebook Speech 4 Horizon Robotics

**Reference:**

**Year: 2016**

<https://openaccess.thecvf.com/content_cvpr_2016/papers/Wang_CNN-RNN_A_Unified_CVPR_2016_paper.pdf>

**Introduction:**

* A Convolutional Neural Network is significantly slower due to an operation such as maxpool. Max pooling is a type of operation that is typically added to CNNs following individual convolutional layers. When added to a model, max pooling reduces the dimensionality of images by reducing the number of pixels in the output from the previous convolutional layer.
* If CNN has several layers, then the training process takes a lot of time if the computer doesn’t consist of a good GPU.
* A convolutional network requires a large dataset to process and train the neural network.

**Working Process of CNN:**

A CNN Algorithm works as follows

* Takes input image
  + CNN takes an image as an input, distinguishes its objects based on three colour planes, and identifies various colour spaces. It also measures the image dimensions.
* Convert it to convolutional layer
  + A CNN converts the images into convolutional neural network by using the formula m x n x 1. Where m is height, n is breadth and 1 is format of image for example jpg,png,etc.
* Pooling layer
  + Pooling layer is essential to decrease the spatial size of the Convolved Feature. So, in short words, it works for decreasing the required computational power for the processing of data by the method of dimensionality reduction.
  + Two types of pooling can be done
    - Max Pooling
    - Average Pooling
  + We used Max pooling in our project as it is easy to implement and no computations are required.
* Classification (Fully connected layer)

The task of the fully connected layer is to connect the output of the previous layer. There is no spatial arrangement in this layer. There can be many fully connected layers where the last layer is connected to the output layer. One of the most commonly used method is soft regression because of its performance

**Paper 8:**

**Introduction to Convolutional Neural Network CNNs**

**Reference:**

# **Year: 2020**

<https://aigents.co/data-science-blog/publication/introduction-to-convolutional-neural-networks-cnns>

**Introduction:**

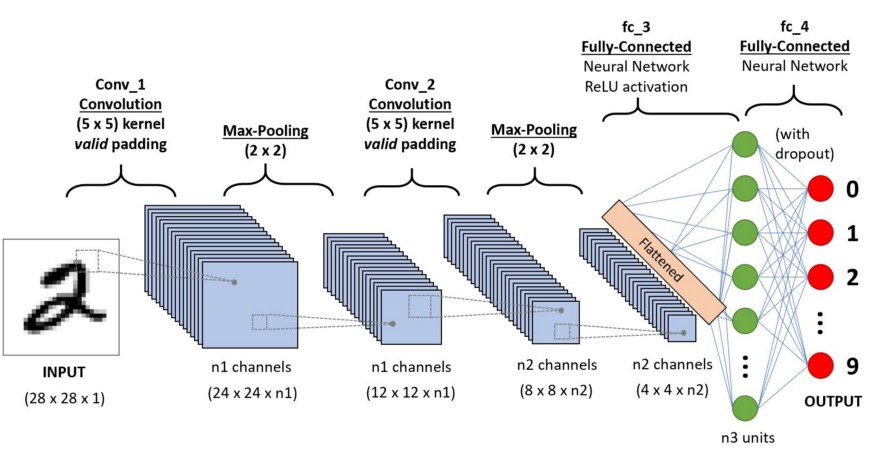
The convolutional Neural Network CNN works by getting an image, designating it some weightage based on the different objects of the image, and then distinguishing them from each other. CNN requires very little pre-process data as compared to other deep learning algorithms. One of the main capabilities of CNN is that it applies primitive methods for training its classifiers, which makes it good enough to learn the characteristics of the target object.

CNN is based on analogous architecture, as found in the neurons of the human brain, specifically the Visual Cortex. Each of the neurons gives a response to a certain stimulus in a specific region of the visual area identified as the Receptive field. These collections overlap in order to contain the whole visual area.

**Working of CNN:**

Input Image:

CNN takes an image as an input, distinguishes its objects based on three color planes, and identifies various colour spaces. It also measures the image dimensions. In order to explain this process, we will give an example of an RGB image given below.



In this image, we have various colors based on the three-color plane that is Red, Green, and Blue, also known as RGB. The various color spaces are then identified in which images are found, such as RGB, CMYK, Grayscale, and many more. It can become a tedious task while measuring the image dimensions as an example if the image is perse 8k (\*7680x4320\*). Here comes one of the handy capabilities of CNN that it reduces the image’s dimension to the point that it is easier to process, which also maintaining all of its features in one piece. This is done so that a better prediction is obtained. This ability is critical when designing architectures having not only better learning features but also can work on massive datasets of images.

**Convolution Layer (Kernel):**

The Kernel of CNN works on the basis of the following formula.

Image Dimensions = n1 x n2 x 1  
 where n1 = height, n2 = breadth, and 1 = Number of channels such as RGB.

If there are multiple channels such as found in RGB images, then the kernel contains the same depth as found in the input image. The multiplication of the matrix is implemented based on the number of Ks. The procedure is followed as in stack format, for example, {K1, I1}, {K2, I2}, and so on. The results are generated based on the summation of bias. The result is in the form of a squeezed “1-depth channel” of convoluted feature output.

The goal of this convolution operation is to obtain all the high-level features of the image. The high-level features can include edges of the image too. This layer is not just limited to high-level features; it also performs an operation on low-level features, such as color and gradient orientation. This architecture evolves to a new level and thus includes two more types of layers. The two layers are known as Valid padding and the Same padding.

The objective of these layers is to reduce the dimensionality of the image that is found in the original input image and to increase dimensionality or, in some cases, to leave it unchanged, depending on the required output. The same padding is applied to convolute the image to different dimensions of the matrix, while valid padding is applied when there is no need to change the dimension of the matrix.

**Pooling layer:**

As identical to the recognized layer “convolutional,” the foremost aim of the Pooling layer is essential to decrease the spatial size of the Convolved Feature. So, in short words, it works for decreasing the required computational power for the processing of data by the method of dimensionality reduction. Moreover, it is also beneficial for the extraction of the dominant features, which are basically rotational as well as positional invariant, so the maintenance of the process effectively is needed.

**Classification: Fully Connected Layer (FC Layer):**

The addition of the FC layer is mostly the easiest way for the learning purpose of the non-linear combinations of the abstract level structures, as it is also revealed by the output of the convolutional layer. The FC layer provides the space for learning non-linear functions. As now we have achieved our task to convert our image output into a specific form of Multi-layer Perceptron, now we must flatten the output image into a form of a column vector. Over the different eras of epochs, the model is basically succeeded for the distinguishing function between the dominating and low-level features.

**Inference:**

The Convolutional Neural Network based on Deep Learning algorithm is explained. The workflow mechanism of CNN is explained with examples. The most powerful architectures in building CNN are given at the end which can help to make powerful AI algorithms for Computer Vision.

**Paper 9:**

**An Analysis of Convolutional Neural Networks for Image Classification**

**Reference:**

**Year: 2018**

International Conference on Computational Intelligence and Data Science (ICCIDS 2018)

<https://www.sciencedirect.com/science/article/pii/S1877050918309335>

**Abstract:**

The most popular convolution neural networks for object detection and object category classification from images are Alex Nets, GoogleNet and ResNet50. A variety of image data sets are available to test the performance of different types of CNN's. The commonly found benchmark datasets for evaluating the performance of a convolutional neural network are an ImageNet dataset, and CIFAR10, CIFAR100, and MNIST image data sets. This study focuses on analysing the performance of three popular networks: Alex Net, GoogleNet, and ResNet50. We have taken three most popular data sets ImageNet, CIFAR10, and since, testing the performance of a network on a single data set does not reveal its true capability and limitations. It must be noted that videos are not used as a training dataset, they are used as testing datasets.

**Introduction:**

CNN has been presenting an operative class of models for better understanding of contents present in an image, therefore resulting in better image recognition, segmentation, detection, and retrieval. CNNs are efficiently and effectively used in many pattern and image recognition applications, for example, gesture recognition, face recognition, object classification and generating scene descriptions. The test datasets are videos of different categories and subjects. The contradiction branches out because of the feature extraction capabilities of different CNN. The primary contribution of our work is to present object detection methods using different types of trained neural networks where current up-to-date models show different performance rates for test images or videos when compared to trained images. After training these networks for different object classes presented as input in the form of images, and then testing for the more particular real-time video feed, we can better understand what is being learned and presented by these models. We therefore, can postulate that an image representation on the basis of objects detected in it would be significantly useful for high-level visual recognition tasks for scenes jumbled with numerous objects resulting in difficulty for the network to classify it. These networks also provide supplementary information about the extraction of low-level features. These networks are trained on datasets containing millions of tiny images.

**Working Process:**

1. **Creating training and testing dataset:** The super classes images used for training is resized [224,244] pixels for Alex Net and [227,227] pixels GoogleNet and ResNet50, and the dataset is divided into two categories i.e. training and validation data sets.

2. **Modifying CNNs network:** Replace the last three layers of the network with fully connected layer, a softmax layer, and a classification output layer. Set the final fully connected layer to have the same size as the number of classes in the training data set. Increase the learning rate factors of the fully connected layer to train network faster.

3. **Train the network**: Set the training options, including learning rate, mini-batch size, and validation data according to GPU specification of the system. Train the network using the training data.

4. **Test the accuracy of the network:** Classify the validation images using the fine-tuned network, and calculate the classification accuracy. Similarly testing the fine tune network on real time video feeds for accurate results.

**Inference:**

The work analysed the prediction accuracy of three different convolutional neural networks (CNN) on most popular training and test datasets namely CIFAR10 and CIFAR100. We focused our study on 10 classes of each dataset only. Our main purpose was to find out the accuracy of the different networks on same datasets and evaluating the consistency of prediction by each of these CNN. We have presented a thorough prediction analysis for comparing the networks’ performance for different classes of objects. It is important to note that complex frames often create confusion for the network to detect and recognize the scene. It was also noted that though in real-world beds and couches as well as chair are different and easily recognized objects but the trained networks showed confusion and therefore differ in accuracy rates. The results suggested that trained networks with transfer learning performed better than existing ones and showed higher rates of accuracy.

**Scope and Future of CNN:**

The use of numerous novel concepts in CNN’s architecture has shifted research priorities, particularly in the field of computer vision. To study innovations in CNN’s architecture is an encouraging study area, and has the potential to become one of the utmost utilized AI techniques.

1. Ensemble learning is an upcoming research area in CNN. By extracting distinct semantic representations, the model can improve the generalization and resilience of many categories of images by combining multiple and diverse designs
2. In picture segmentation tasks, although it performs well, a CNN’s ability as a “generative learner” is limited. The use of CNNs’ generative learning capabilities throughout feature extraction phases can improve the model’s representational power. At the intermediate phases of CNN, fresh examples can be incorporated to improve the learning capability by using auxiliary learners.
3. It is observed that the learning capability of a CNN is mainly increased by increasing the network’s size, and this may be achieved by modern advanced hardware technologies, such as the Nvidia DGX-2 supercomputer. Nonetheless, training more deep and high-capacity CNN architectures consumes a substantial amount of memory and computing resources.

**Final Conclusion of what we will be implementing in our Project:**

In our project we will be using the X Ray dataset which contains images of the lungs of people infected with pneumonia, people infected with bacterial pneumonia and normal uninfected people. Firstly, we will be checking our dataset for any dodgy images which have been corrupted and get rid of them. This cleaned dataset is provided as input to the pre-processor which scales and partitions the dataset into training, testing and validating sets. Once this is done, we will begin to build our deep learning convolutional neural network using the tensorflow and keras utility that is provided in python and once it has been built it is trained using the training set of images. After the CNN has been trained its performance is evaluated using the test dataset. Once tested our model is combat ready and can classify the images the user gives providing it belongs to one of the classes our CNN has been trained in.