# OBJECT DETECTION USING CNN

Ms.Gunasundari B[1], Lokesh R[2], Gopirengaraj C[3.]
[1]Assisstant Professor, [2,3]Final year student,
Department of CSE, Prathyusha Engineering College,

**ABSTRACT:**

Object Detection is the process of finding what object is present the image. This can be done using Deep learning algorithms. To find an object, we can use their shapes or edges. The edges of the objects are memorized by the trained model and that is used to find the objects which have been trained to it. The identification of particular object will help in various problem. There are more ways to find an object in an image, but convolutional neural networks will give more accuracy and minimum loss.

**KEYWORDS:** Object Detection, CNN, Deep Learning.

## 1. INTRODUCTION:

The Convolutional neural networks are the popular technique to find objects in the image. This study is shows the implementation process of Object detection. This process comes under supervised learning process. In this experiment, the images are collected from various websites. For 10 objects, 1000 images were downloaded for each classes. They are resized to same size and the label for each image is created. 700 images were taken for training purpose, 300 images for testing purpose. All the images were preprocessed and feed into the model. The model will understand the structure of each class and it will be able to predict the trained objects in the real time. For training purpose, GPU's has been used for faster training. After training, the model has to be tested, wether it is capable of predicting the trained objects. This is work is done for the study purpose, to know about the convolutional neural networks.

## 2. CNN

The convolutional neural networks has many layers including input layer, in which the input images are given, output layer, which will predict the object in the image and in between many hidden layers will be present, which help the model to predict the object in the image. The hidden layer consists of convolution layer, pooling layer and fully connected layer. The convolution layer is used to detect the pattern in the image. In this layer, the image is multiplied with a filter matrix to detect the pattern of the object. This will feed forward to the next layer. Sometimes in the same convolution layer more number of filters are used to detect the pattern. The filters are the weights, the values will be changing for each iteration in the training process. If one filter is multiplied, the result image matrix has depth one, if n numbers of matrices are used then the depth will be n. The next one is the pooling layer, which will combine the output of the previous layer. This layer will reduce the size of the image in to half. Two types of pooling are there, one is max pooling and another is average pooling. Again another convolution and pooling will be there. Finally the image will be flattened in to single dimension. This output is feed in to our model with the label and the training will get started.

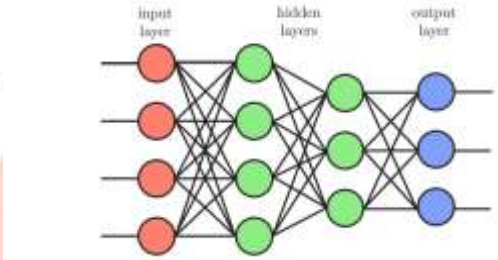


Figure 1: A simple convolutional neural network with hidden layers.

## 3. DATASETS:

The images are collected from various sites for each classes**.** For each classes 1000 images were downloaded. Each images are resized into 32x32 dimension. Each image has the object in the center. 70% of the images were used for the training purpose. 30% percent were used for the testing purpose. Training images were stored in the separate directory. Each class of image is stored in separate directory. Testing images are also saved like that. All the images were converted into an array. There were two arrays, one for training purpose, whose dimension is 7000x32x32 and other one is for testing purpose, whose dimension is 3000x32x32. They were converted into pickle file using python cpickle library. So that we need not read the whole images for all the time.


Figure2: Some images from the training dataset

## 4. CONVOLUTION:

Each image is multiplied with a filter matrix, which is used to extract the feature from it. Initially the filer contains random values and it is a 3x3 matrix. The result matrix will be 32x32 and if we use more than one filter, the result will change. For example if we use 4 filters, the output will be 32x32x4.

## 5. POOLING:

The pooling process is to reduce the dimension of the image as well as the depth of image. This process is used to flattened the image, finally the array will be 1x1x1024. This array and the label is feed in to the training process with set of some parameters.

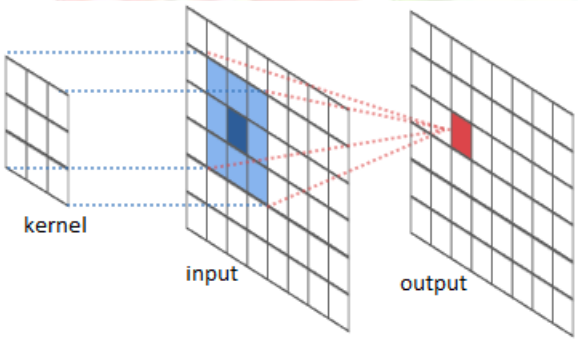


Figure 3: The convolution and pooling process of the image

## 6. TRAINING:

The training process will start with set of hyperparameters like no of epochs, number of images per epoch and the most important parameter, the training rate. The training rate must be very low so that the model can get the best fit. For each epochs, the loss value is calculated. Initially, the loss value will be larger, and in time it will comes to nearly zero. After the training is stopped, some value will be generated for the filters. We need to store those values. And we are using one-hot encoding, so if the detect object is correct, then there will be one in the object position in the label and other nine values will be zero.

## 7. TESTING:

The testing array is again convoluted with filter values, which we get from the training.
We need to do the convolution and pooling process again and finally we will get an one dimensional array of 0's and 1's. With that array we need to calculate the accuracy using the original labeled array.

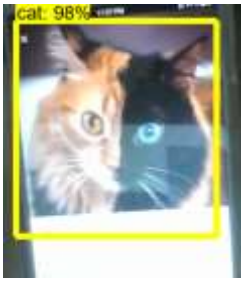


Figure 4: Object identified as cat by 98%

If the accuracy is more than 95%, we can say that our model predicts well. Otherwise we need to change the hyperparameters in the training process and do the training process once again. Some time we need to add extra filters in the convolution process.

## 8. REFERENCE:

[1] Bengio, Y., et al. Deep learning of representations for unsupervised and transfer learning. ICML Unsupervised and Transfer Learning 27 (2012), 17–36.
[2] Imagenet large scale visual recognition challenge 2016.http://image-net.org/challenges/LSVRC/2016. Accessed: 2017-03-18.
[3] Infotrends - how long does it take to shoot 1 trillion photos? http://blog.infotrends.com/?p=21573. Accessed: 2017-06-20.
[4] Bodla, N., Singh, B., Chellappa, R., and Davis, L. S. Improving object detection with one line of code. arXiv preprint arXiv:1704.04503 (2017).