S. No. - 99

OBJECT DETECTION ON INDIAN ROADS AND ROADSIDES CLASSIFICATION

BACHELOR OF TECHNOLOGY PROJECT (PART-I)

IN

MECHANICAL ENGINEERING

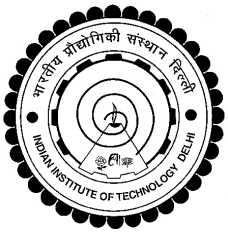
Submitted by

KULDEEP SHARMA Entry No: **2013ME20785**

Under the guidance of

PROF. SUBHASHIS BANERJEE **PROF. SUDIPTO MUKHERJEE**

Signature(s) **Signature(s)**



Department of Mechanical Engineering

Indian Institute of Technology Delhi

New Delhi 110 016

CONTENTS

List of figures

List of tables

Nomenclature

List of Abbreviations

CNN – Convolution Neural Network

SGD – Stochastic Gradient Descent

SVM – Support Vector Machine

AI – Artificial Intelligence

RCNN – Region-based Convolutional Neural Networks

mAP – Mean Average Precision

ABSTRACT

This document reports on development of a state-of-the-art Deep Networks to detect and classify road users on roads and sides in an Indian environment. Deep Networks like VGG16, AlexNet and ResNet (*Residual Networks*) have been implemented and Image classification accuracy on Indian images dataset and standard dataset ImageNet has been used to compare the methods. Using AlexNet, the image classification accuracy on Indian images dataset (59%) which is lower than the accuracy on ImageNet dataset (63.5 %). We collected dataset of roadsides, used it for training AlexNet network and this network achieve an accuracy of 83.54% on this task of roadside classification. For the object detection, we use VGG16 and ResNet and train them on Pascal-VOC 2007 dataset and achieves a mAP of 69.6 % and 71.26 % respectively.

CHAPTER 1

INTRODUCTION

In the last few years deep learning has been a trending topic around the world and it is not just trending on paper, we can see the effect deep learning in various applications like image classification[1], object detection[2], image segmentation[3] and many other. Deep networks have strength in learning complex functions. Deep convolution neural networks(CNN) have a layered structure and each layer consisted of convolutional filters. At earlier layers each filter look for a special and simple features from image and then combination of these features make whole new complex structure in the subsequent layers, which deep network understands and gives most probable output. We used the deep visualisation toolbox [4] to visualise the function of each neuron in AlexNet.

In our daily life object detection is a very important task to make our decision for successful movement. Human's eye act as a camera and provide input image to our brain's neural network which makes the decision about our movement. As time passes it's human innate feeling to move towards new technology so now we are moving towards the making machines which detects object and then makes decision by itself e.g. autonomous vehicles and due to increase of it's application in the filed like autonomous vehicles, robotics object detection has become indispensable part of computer vision. Object detection has been a challenging problem in computer vision. After revival of deep networks in 2012 in the performance of many vision task such as image classification *(Krizhevsky et al*.[1]), object detection (*Girshick et al.*[2]) has almost reached to human accuracy.

Till date, little of the development has been done in Indian context and Indian locality. This becomes especially relevant in the context of vision as the sensory output is affected by both the object and ambient lighting. We implemented state-of-the-art object detection papers[2], [5], [6] with different pre-trained(on ImageNet dataset) deep networks such as VGG16 (*Simonyan et al.*[7]) and ResNet (*He et al*.[8]) and trained these networks on *Pascal-VOC 2007 dataset.*

We have divided this project in two different parts. First part of the project is to find the performance of state-of-the-art deep networks on Indian images. We trained many deep networks such as AlexNet[1], VGG16[7] on Indian images and compared their performance with accuracy obtained by these networks on the ImageNet dataset. The second part of the project is to create a virtual environment of the environment around the vehicles. This consist of detection of various objects present on the roads along with roadsides such as market area, residential area, highway etc. Now after creating this virtual environment we can have better control of the vehicle and hence we can reduce the number of accidents to a great extent. After detecting the number of different objects such as cars, pedestrians in a given area, we can find the density of the these object and then control the speed of vehicle accordingly.

The remainder part contains Literature review, Objectives, Results and Conclusion.

CHAPTER 2

LITERATURE REVIEW

One of the earliest work in the field of neural network was done by Hubel & Wiesel where they performed experiments on a cat and learned when does a neuron get activated. They found that some neuron fired rapidly when they saw a line projected at some angle whereas other neuron did it for other lines projected at some different angle. After few years in 1980 Fukushima's "neocognitron"[9] come up with a biological inspired deep neural network model but his model lacked a supervised learning. Then in 1989 *LeCun et al.* [10] showed that SGD and back propagation can be used to train deep neural networks. In early 90's deep neural network saw heavy use but being hard to train and lack of the data present at that time made it hard to get expected results. And also with the rise of SVM, a simpler model to train and understand, deep neural networks started to lose their charm. In 1997, *LeCun et al.*[11] come up with a 5 layer CNN, 2 fully-connected layers on top of it and in the and one last layer to classify, called *LeNet-5.* But due to the limitation in the computation power and scarcity of the dataset we were not able to use these networks on large scale. In 2012, the year of revival of deep learning, due to increase in the computation power and the presence of massive dataset like ImageNet, *Krizhevsky et al*.[1] showed the importance of deep learning. He used the same network presented in *LeCun et al.*[11] and trained it on the ImageNet dataset and improved the accuracy of image classification task significantly.

After this, large part of AI research society started working in the field of deep learning. A worldwide famous challenge called ImageNet Large Scale Visual Recognition Challenge (ILSVRC) happens every year in December. People from the whole world come with their deep network and whichever networks perform the best wins the competition. In 2012, AlexNet *Krizhevsky et al*.[1] won this competition. In 2013, ZF *Zeiler et al.*[12] they made slight changes in the parameters of AlexNet and improved the performance. In 2014, GoogLeNet and VGG16 (*Simonyan et al.*[7]), both modified their network a lot and also increased the number of layers as compared to the earlier networks with number of layers 22 and 16 respectively. GoogLeNet won the competition but VGG16 gained more fame due to it's simple network architecture compared to GoogLeNet and not much difference in the accuracy. One of the biggest problem in deep network was of vanishing gradient and hence we couldn't go very deep while creating new networks. So we always had the curiosity that what would happen if we go more deep. In 2015, MSRA(Microsoft Research Asia) team come up with the residual networks called ResNet (*He et al.*[8]) and provided a solution to the problem of vanishing gradient by adding a residual branch in parallel to the stacked convolution layers. They come up with the network of 152 layers and won the competition and achieved the accuracy better than human.

Now with the significant improvement in image classification task due to these new deep networks and dataset present next questions comes how to improve the task of object detection using deep learning. In 2013, *Girshick et al.*[2] came up with an idea of using this CNN network on multiple region of a single image and then classify each region and hence the name Region-based Convolution Neural Network(RCNN). Beside this they also made a bounding box around the objects which get detected. They used the pretrained networks(on ImageNet dataset) and added region proposals on top of the networks and trained this whole new networks on Pascal-VOC dataset to adapt these pretrained network for new task of object detection. This was a momentous idea for the object detection task. But this method was very slow and hence it can only be used as an offline process. In 2015, *Girshick et al.*[5] made changes in the RCNN(*Girshick et al.*[2]), instead of using CNN on each region from image they first used CNN on original image and mapped the original image to convolution feature map and then applied regions proposals on convolution feature map. So this change reduced the time taken in processing one image by a significant amount and hence the name Fast-RCNN. After few months, *Shaoqing et al.*[6] made some changes in *Girshik et al.*[5], they used the regions-proposal network(RPN) instead of regions proposals methods (selective search) on top of pretrained networks and trained it in single step. So these RPN performed faster and added a slight improvement in performance and hence the name Faster-RCNN.

For the tracking of detected objects I read about many famous tracking algorithms. One of the famous and most commonly used is Kalman Filter, *Kalman et.al.[12].* This paper helped me in understanding the basic structure about the prediction of location which helped me in predicting the pedestrian spatial location in the missing frames.

My objective is to detect objects on Indian roads and the classification of Indian roadsides. For this We are using these state-of-the-art deep networks and faster-RCNN method for object detection and training them on Indian dataset. By doing so we can create a virtual environment for the vehicles and then we can control the vehicle accordingly hence we can reduce the number of accident. And this is very important in the field of autonomous vehicles to make sure that before giving the control of the vehicle to a machine that we have considered every aspects otherwise it can lead to disaster. Especially in India we have to be very careful because people still don't follow the traffic rules and this can be really difficult it it really important that we consider more parameters than only traffic rules.

CHAPTER 3

OBJECTIVES

The main objectives of the present work are 1) to compare the performance of the deep networks on Indian images with standard dataset, 2) to develop a model which detect objects on Indian roads and classify the roadsides. By comparison of performance we can

analyze the difference between images and then we can use the appropriate filters or changes in network to get better performance. And object detection is a very crucial part of autonomous vehicle and even a slightest mistake can lead to a disaster. So my objective is to create a virtual environment of the surroundings of the vehicle and by considering object on roads and information about the roadsides we can have better control.

To achieve the objective, we have implemented the state-of-the-art object detection and classification models. To improve their performance in Indian context we are collecting more data and once we have enough data, we will train these networks and analyze their performance. To further improve their performance, we can change the hyper parameters values of the network to obtain the best network.

CHAPTER 4

RESULTS AND DISCUSSION

In the first part of the project we worked on understanding deep learning and implementing the state-of-the-art deep networks. We used caffe's python wrapper to implement networks. And then compared these networks on the basis of their accuracy of image classification task on the standard dataset ImageNet. In fig.1 we can see the revolution of depth on classification task and the strength of each network. From fig.1

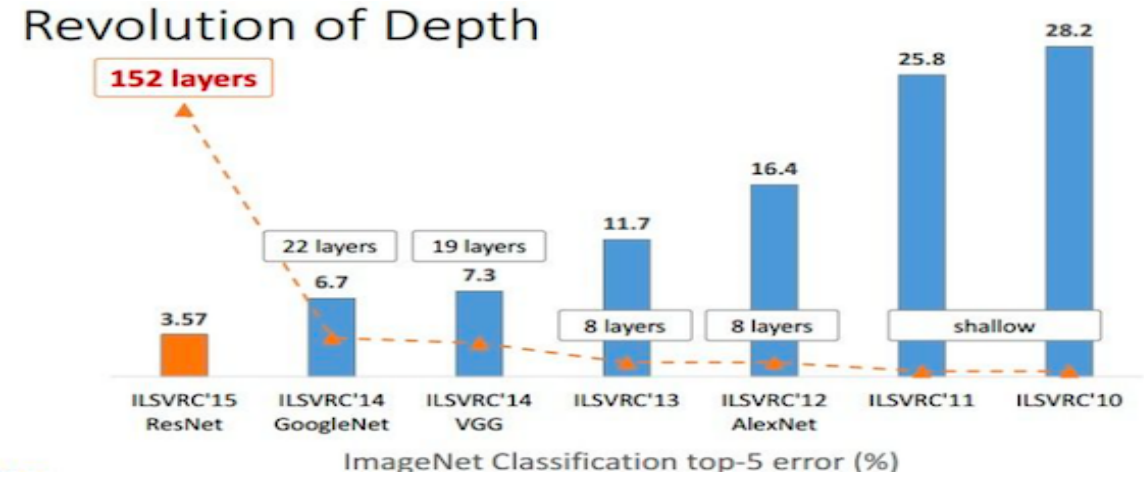


Fig. 1: This is showing the top-5 error % on ImageNet dataset vs the network with their number of layers. (Ref. Kaiming He's presentation in ILSVCRS'15)

we can clearly see a sharp decrease in the top 5% error value and clearly showing the strength of deep learning. And then every subsequent year with the increase in the number of layers we achieved better accuracy. So after having a basic understanding about each network we finetuned AlexNet and VGG16 network on Indian images. We collected a dataset of around 7,000 images of 7 different classes like cars, cycle, auto etc. and each class has roughly 1,000 images. We used around 800 images to train and remaining images we used for the testing of the network. And then for each class we have created a

confusion matrix. Confusion matrix is a matrix shows the actual label and the predicted labels for each data point so that we can see that which class is getting labelled to which class. This makes easy to see the right classification and misclassification as well. We worked on AlexNet networks, trained it from scratch and also finetuned it on Indian images dataset.

Label	Class	Number of Test Images
0	Two Wheeler	219
1	Auto	181
2	Cars	296
3	Cycle	273
4	Lamp + Street Light	125
5	Pedestrian	146
6	Street Signs	225

Table 1: This contains information of the test dataset of Indian Images

r,p	count
0,0	112
0,1	41
0,2	39
0,3	18
0,4	3
0,5	4
0,6	2
1,0	76
1,1	70
1,2	2
1,3	0
1,4	10
1,5	11
1,6	12
2,0	71
2,1	21
2,2	144
2,3	15
2,4	35
2,5	7
2,6	3
3,0	78
3,1	7
3,2	56
3,3	125
3,4	2
3,5	1
3,6	4
4,0	12
4,1	2
4,2	1

r,p	count
0,0	143
0,1	19
0,2	51
0,3	1
0,4	0
0,5	0
0,6	5
1,0	46
1,1	104
1,2	1
1,3	0
1,4	13
1,5	13
1,6	4
2,0	26
2,1	16
2,2	225
2,3	1
2,4	3
2,5	23
2,6	2
3,0	50
3,1	0
3,2	14
3,3	195
3,4	0
3,5	2
3,6	12
4,0	1
4,1	1
4,2	1

r,p	count
0,0	141
0,1	18
0,2	54
0,3	2
0,4	0
0,5	2
0,6	2
1,0	49
1,1	82
1,2	2
1,3	0
1,4	28
1,5	15
1,6	5
2,0	39
2,1	14
2,2	207
2,3	1
2,4	7
2,5	27
2,6	1
3,0	66
3,1	1
3,2	9
3,3	186
3,4	0
3,5	1
3,6	10
4,0	1
4,1	0
4,2	1

4,3	2
4,4	99
4,5	8
4,6	1
5,0	21
5,1	0
5,2	4
5,3	4
5,4	22
5,5	93
5,6	2
6,0	3
6,1	4
6,2	0
6,3	0
6,4	3
6,5	0
6,6	215

4,3	0
4,4	120
4,5	2
4,6	0
5,0	4
5,1	1
5,2	1
5,3	0
5,4	5
5,5	135
5,6	0
6,0	0
6,1	0
6,2	0
6,3	0
6,4	0
6,5	0
6,6	225

4,3	0
4,4	120
4,5	3
4,6	0
5,0	3
5,1	0
5,2	0
5,3	1
5,4	8
5,5	134
5,6	0
6,0	0
6,1	0
6,2	0
6,3	0
6,4	0
6,5	0
6,6	225

Table 2: This is the confusion matrix when AlexNet trained from scratch on Indian data.
Table 3: This is the confusion matrix when pretrained AlexNet finetuned on Indian data.
Table 4: This is the confusion matrix when pretrained AlexNet with one fully-connected less finetuned on Indian data. (r=actual label & p=predicted label)

As we can see from the above three tables that the main confusion is between two-wheeler and cycle, two-wheeler and auto, two-wheeler and cars and there was almost no confusion in classifying the street sign. Accuracy obtained on test dataset of total 1466 images in Table 2, Table 3 and Table 4 is 58.5%, 78.2% and 74.7% respectively. From Table 2 it is cleared that when we used Indian Images to train network accuracy(58.5%) was slightly less as compared to standard dataset ImageNet(62.5%).

After image classifcation task on indian dataset we moved to the object detection task on indian roads. We used the faster-rcnn with VGG16 network to detect 20 objects of pascal-VOC dataset. We used caffe library to implement faster-rcnn with VGG16 and ResNet-50 networks. We compared their performance on VOC'07 dataset and ResNet-50 performed slightly better than VGG16 with mAP estimation of 71.26 % and 69.46 % respectively and for pedestrian it is 77.76% and 75.78% respectively. But being a very large network of 50 layers, ResNet(0.41 sec. per image), take more time to process single image as compared to VGG16 (0.2 sec. per image), a 16 layer network. Initially, we worked only on pedestrian detection before moving for to the detection of any object and now pedestrian detection program is running, it takes a video as input and generetes a video with detected pedestrian and a csv file which contain the spatial and temporal information about each detected pedestrian. We have run this program on many short clips of *2-3 min.* and long videos of around *10 min*. making total length of video more than *60 min.* and it is working fine. We couldn't calculate the effiecieny due to the limitation of annotated datasets but we could see it through the detected videos that it was detecting almost every pedestrian.

Fig. 2: This figure shows the pedestrian and car detection on Delhi's road. Each detection with the bounding box with the probability of being that object on top of the box.

Our program of pedestrian detection is working fine, it detects every pedestrian at least in one frame out of total frames in which this particular pedestrian is present. For e.g. if a person is present in frame number 10 to 25, then this person is being detected at least in one frame between 10-25. Now, this is a problem because we want to detect this person in every frame he is present. To solve this we used Kalman Filter, a very famous technique used for tracking objects. By using the information of the pedestrian, present in some of frames in which it is being detected, Kalman Filter predicts its motion. We use this predicted motion and fill the blank spaces i.e. we predict the spatial location of the pedestrian for the frames in which the pedestrian was not being detected. This technique helped us in finding the spatial location of the pedestrian in every frame and hence improving the overall performance of our system.

Now while finding the efficiency of a network in the task of object detection we match detected bounding box with the annotated bounding box manually, which is a tedious job to do. We write the python program to solve this problem. This python program takes 2 csv file as input and return the number of false positive, false negative etc. For a given threshold (number of pixels) it matches the predicted bounding box from one csv file with the annotated bounding box of the other csv file. Then on the basis of the distance between the centres of the bounding boxes and the given threshold, it makes the decision about this bounding if it is false positive or false negative. This python program will help in future when we will have lots of annotated dataset then matching the predicted bounding box and the given annotated bounding box will not be a tedious task.

After this we started working on our second part of project i.e. Roadsides classification. First we collected the dataset form different parts Delhi including Markets, Residential Areas, Parks, Flyovers etc. We mounted a GoPro camera on a vehicle and drove the vehicle through various parts of South Delhi such as AIIMS, RK Puram, South ex, Yusuf Sarai etc. We recorded a video of around 2 hours and then we converted this into many images by breaking it into frames with 30fps(frames per second). Then we searched through the all images and collected the images which can be used for training and testing purpose. So we collected around 16,000 images of four different classes (Flyover, Market, Residence and Parks). Out of these 16,000 images we selected 10,000 images and we created our final dataset which we used for training and testing.

Labels	Classes	Number of Training images	Number of Test images
0	Residence	2000	500
1	Flyover	2000	500
2	Parks	2000	500
3	Market	2000	500

Table 5: This contains information of the dataset of roadsides images

Fig. 3: These are the images of different classes of the dataset.
(Flyover - Top Left, Park - Top Right, Residence - Bottom Left, Market -Bottom Right)

After the collection of the data next step is to train a deep neural network on this dataset. I have already explained about the state-of-the-art deep networks in *Literature Review*, for this part we chose the AlexNet because of its simple and small architecture. As we have a small number of dataset and classes so there is no point in taking more strong networks as compared to the AlexNet, although we could try this also and compare the results.

We trained AlexNet on this dataset, it took around 2 hours for 2,000 iteration on our supercomputer in IIT Delhi. We achieved an accuracy of 83.54% on the test dataset of 2,000 images(500 per class).

```
Iteration 2000, loss = 0.377562
Iteration 2000, Testing net (#0)
    Test net output #0: accuracy = 0.8354
    Test net output #1: loss = 0.407675 (* 1 = 0.407675 loss)
```

Fig. 4: This is the screenshot of the supercomputer showing test accuracy → 0.8354

After training we created confusion matrix on the test dataset to analyse the performance of this network. Confusion matrix as I have explained above helps us in finding the correct classification and misclassification.

r,p	0,0	0,1	0,2	0,3
count	398	19	8	75

r,p	1,0	1,1	1,2	1,3
count	2	482	0	16

r,p	2,0	2,1	2,2	2,3
count	57	0	443	0

r,p	3,0	3,1	3,2	3,3
count	149	5	0	346

Table 6: This is confusion matrix showing actual label and predicted label with number of data points as count.(r – Actual Label p – Predicted Label)

Accuracy: 83.5%, 1669 out of 2000 were classified correctly

From the above confusion matrix it is very clear that there is very less misclassification for the class label 1(Flyover) and 2(Parks) and this very easy to understand because the features which represent a Flyover is very different from the Parks, so chances of confusion is almost zero. But the number of misclassification is more in between label 0(Residence) and 3(Market) and this can be justified because both market and residence have almost same features i.e. same building structure. The only major different feature is hoardings present on the market shops and which are not there on most of the houses. And also there is not any standard definition of house and market on the basis of their structure, so a human can also be wrong while making a decision. But the whole point of research in this field is to create deep networks which will outperform humans, so in future we might have stronger deep networks than AlexNet which will be able to defeat humans.

CHAPTER 5

CONCLUSIONS & FUTURE SCOPE

In first part we showed that accuracy of image classification task is less when trained on Indian images as compared to standard dataset such as ImageNet dataset.

We used the state-of-the-art network and trained them on Pascal-VOC dataset and used it for detection on Indian roads and we found that detection is not as good as we were expecting. There could be multiple reason for this and one of main reason we think is that

we trained these networks on a dataset which does not belong Indian context. So there are many source of variation between test images(i.e. Indian images) and training images (i.e. VOC'07 dataset) e.g. image quality, Indian attire etc. So to improve the accuracy we need a large dataset of annotated images of Indian context and train the network on those images and also the other reason could be that these network are very big and like we had a big dataset e.g. ImageNet for image classification if we can have big dataset for detection as well then we think the accuracy can be increased to a great extent.

In second part of this project we worked on the problem of roadsides classification. We collected dataset, trained the AlexNet network and achieved an accuracy of 83.54%.

In future, Machine learning and Deep learning is going to be a big thing. Especially in the field of computer vision, applications will increase day by day as we will move forward. For e.g. in current time one of trending research in the field of Deep learning is going on Autonomous vehicles. Every big automobiles companies such as Tesla, BMW, Volvo etc. are spending a significant amount of time and money on the research in this direction.

To improve the performance of deep networks we need to have in-depth knowledge about deep learning, resources and huge amount of experimentation. Size and quality of dataset plays an important roles in the final performance of the deep networks.

GANTT CHART

Work Elements	Aug 1-15	Aug 15-31	Sept 1-15	Sept 15-30	Oct 1-15	Oct 15-31	Nov 1-15	Nov 15-30
Literature review								
Image Classification								
Pedestrian Detection								
Tracking Pedestrian								
Data Collection								
Roadside Classification								

REFERENCES

[1] Alex Krizhevsky, Ilya Sutskever, and Geoff Hinton. ImageNet classification with deep convolutional neural networks. In Advances in Neural Information Processing Systems 25, 2012

[2] R. Girshick, J. Donahue, T. Darrell, and J. Malik, "Rich feature hierarchies for accurate object detection and semantic segmentation" in IEEE Conference on

Computer Vision and Pattern Recognition (CVPR), 2014.

[3] Jonathan Long, Evan Shelhamer, Trevor Darrell, Fully Convolutional Networks for Semantic Segmentation, CVPR, 2015.

[4] Jason Yosinski, Jeff Clune, Anh Nguyen, Thomas Fuchs, and Hod Lipson. Understanding neural networks through deep visualization. Presented at the Deep Learning Workshop, International Conference on Machine Learning (ICML), 2015.

[5] R. Girshick, "Fast R-CNN," in IEEE International Conference on Computer Vision (ICCV), 2015.

[6] S. Ren, K. He, R. Girshick, and J. Sun, "Faster R-CNN: Towards real-time object detection with region proposal networks," in Neural Information Processing Systems (NIPS), 2015.

[7] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," in International Conference on Learning Representations (ICLR), 2015.

[8] K. He, X. Zhang, S. Ren, and J. Sun. "Deep residual learning for image recognition" In *CVPR*, 2016.

[9] K. Fukushima. Neocognitron: A self-organizing neural network model for a mechanism of pattern recognition unaffected by shift in position. Biological cybernetics, 36(4):193–202, 1980.

[10] Y. LeCun, B. Boser, J. Denker, D. Henderson, R. Howard, W. Hubbard, and L. Jackel. Backpropagation applied to handwritten zip code recognition. Neural Comp., 1989.

[11] Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner. Gradient based learning applied to document recognition. Proc. of the IEEE, 1998.

[12] R.E. Kalman: A new approach to Linear fitting and Prediction problems, 1960.