

A REPORT OF SIX WEEKS INDUSTRIAL TRAINING (14 PT.)
at
NIELIT LUCKNOW (MINISTRY OF ELECTRONICS AND
INFORMATION TECHNOLOGY, GOVERNMENT OF INDIA) ON
ARTIFICIAL INTELLIGENCE USING PYTHON
PROJECT: Object Detection

Submitted in partial fulfillment of the requirements for the award of degree of

**BACHELOR OF ENGINEERING
IN
COMPUTER SCIENCE & ENGINEERING**




Submitted By:
Shashank Kumar - 17BCS2291

**DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING
Chandigarh University, Gharuan
June-July 2019**

Certificate of Training Completion

Certificate No.: NC/LKO/15254 Roll No.: 2

National Institute of Electronics & Information Technology (NIELIT)
An Autonomous Scientific Body of Meity, Ministry of Electronics & Information Technology, Govt. of India

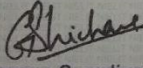
 रा.इ.सू.प्रौ.सं.
NIELIT

A-1/9, Sumit Complex, Vibhuti Khand, Gomti Nagar, Lucknow.

CERTIFICATE

This is to Certify that SHASHANK KUMAR S/o RAKESH KUMAR
has successfully completed a course of 90 ~~week(s)~~ / hour(s) duration from 10/06/2019 to
11/07/2019 on Artificial Intelligence using Python
conducted at NIELIT Lucknow
securing 85 marks out of 100 marks and has been awarded grade * S.

Curriculum for the Course
Artificial Intelligence & Machine Learning using Python (Module covered: NumPy, Panda,
Matplotlib) and Completed AI/ML based Project using Python


Programme Coordinator
Date : 11/07/2019


Centre-In-charge

***GRADING SCHEME : S - 80% and above, A - 70% to <80%, B - 60% to <70%, C - 50% to <60%, D - 40% to <50%**

CHANDIGARH UNIVERSITY,GHARUAN,MOHALI

CANDIDATE'S DECLARATION

I Shashank Kumar hereby declare that I have undertaken six weeks industrial training at NIELIT LUCKNOW during a period from __10 June_2019__ to __11 July 2019__ in partial fulfillment of requirements for the award of degree of B.E (COMPUTER SCIENCE & ENGINEERING) at CHANDIGARH UNIVERSITY GHARUAN, MOHALI. The work which is being presented in the training report submitted to Department of Computer Science & Engineering at CHANDIGARH UNIVERSITY GHARUAN, MOHALI is an authentic record of training work.

Signature of the Student

The six weeks industrial training Viva–Voce Examination of _____ has been held on _____ and accepted.

Signature of Internal Examiner

Signature of External Examiner

ACKNOWLEDGEMENT

Motivation and guidance are the key towards the success. I would like to thank all the source of motivation and who guided me with due respect and gratitude.

I express my deep sense of gratitude to my training instructor **Mr Ghanshyam** , for his timely help and advice.

Finally, i would like to extend my thanks to all teaching and non-teaching staff-members of Computer Science and Engineering for their co-operation in completing this project.

Name-

Shashank Kumar

ABSTRACT

Creating accurate Machine Learning Models which are capable of identifying and localizing multiple objects in a single image remained a core challenge in computer vision. But, with recent advancements in **Deep Learning, Object Detection** applications are easier to develop than ever before. TensorFlow's Object Detection API is an open source framework built on top of TensorFlow that makes it easy to construct, train and deploy object detection models.

Object Detection is the process of finding real-world object instances like car, bike, TV, flowers, and humans in still images or Videos. It allows for the recognition, localization, and detection of multiple objects within an image which provides us with a much better understanding of an image as a whole. It is commonly used in applications such as image retrieval, security, surveillance, and advanced driver assistance systems (ADAS).

Object Detection can be done via multiple ways:

- Feature-Based Object Detection
- Viola Jones Object Detection
- SVM Classifications with HOG Features
- Deep Learning Object Detection



TABLE OF CONTENTS

Sr no	Topics	Page no.s
1	[Chapter-1] INTRODUCTION	8-9
2	[Chapter-2] TRAINING WORK UNDERTAKEN	10-11
3	About NumPy, Pandas etc.	12-13
4	Project: https://github.com/shashank077/Object-Detection	
4a	Source Code	14-15
	[Chapter-3] RESULTS	
4b	Sample inputs	17-18
4c	Module(s) used	19
	[Chapter-4] CONCLUSION	20

CHAPTER-1

INTRODUCTION

What is Object Detection?

Object detection is a computer technology related to computer vision and image processing that deals with detecting instances of semantic objects of a certain class (such as humans, buildings, or cars) in digital images and videos. Well-researched domains of object detection include face detection and pedestrian detection. Object detection has applications in many areas of computer vision, including image retrieval and video surveillance.

One of the important fields of Artificial Intelligence is Computer Vision. Computer Vision is the science of computers and software systems that can recognize and understand images and scenes. Computer Vision is also composed of various aspects such as image recognition, object detection, image generation, image super-resolution and more. Object detection is probably the most profound aspect of computer vision due the number practical use cases. In this tutorial, I will briefly introduce the concept of modern object detection, challenges faced by software developers, the solution my team has provided as well as code tutorials to perform high performance object detection.

Object detection refers to the capability of computer and software systems to locate objects in an image/scene and identify each object. Object detection has been widely used for face detection, vehicle detection, pedestrian counting, web images, security systems and driverless cars. There are many ways object detection can be used as well in many fields of practice. Like every other computer technology, a wide range of creative and amazing uses of object detection will definitely come from the efforts of computer programmers and software developers.

Getting to use modern object detection methods in applications and systems, as well as building new applications based on these methods is not a straight forward task. Early implementations of object detection involved the use of classical algorithms, like the ones supported in OpenCV, the popular computer vision library. However, these classical algorithms could not achieve enough performance to work under different conditions.

The breakthrough and rapid adoption of deep learning in 2012 brought into existence modern and highly accurate object detection algorithms and methods such as R-CNN, Fast-RCNN,

Faster-RCNN, RetinaNet and fast yet highly accurate ones like SSD and YOLO. Using these methods and algorithms, based on deep learning which is also based on machine learning require lots of mathematical and deep learning frameworks understanding. There are millions of expert computer programmers and software developers that want to integrate and create new products that uses object detection. But this technology is kept out of their reach due to the extra and complicated path to understanding and making practical use of it.

CHAPTER-2

TRAINING WORK UNDERTAKEN

Software Used in Projects and during training:

- Programming Language: -At the core of the project is Python 3.5.2, since the whole project is coded in Python.

- Operating System: - Windows 10 version 1903
- Anaconda
- Spyder
- Jupyter Notebook
- Pycharm
- IDLE py3.5.2

However, some of the most important libraries and tools used for this software are listed below: -

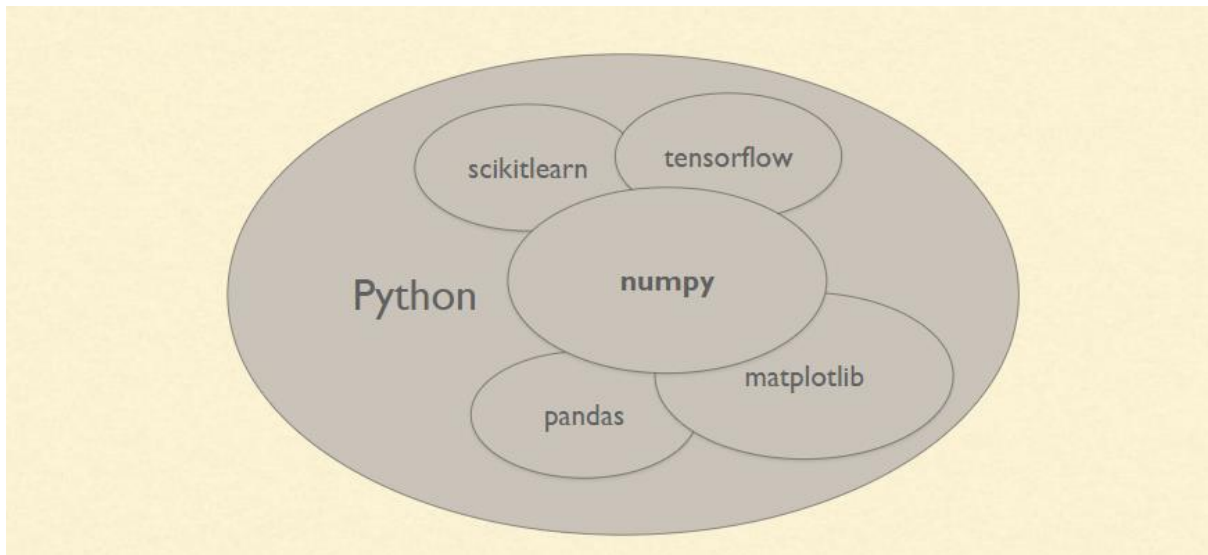
1. The **OS module in python** provides functions for interacting with the operating system. **OS**, comes under **Python's** standard utility **modules**. This **module** provides a portable way of using operating system dependent functionality. The ***os*** and ***os.path*** **modules** include many functions to interact with the file system.
2. **ImageAI** is a **python** library built to empower developers, reseachers and students to build applications and systems with self-contained Deep Learning and Computer Vision capabilities using simple and few lines of code.

Hardware Requirements for the Project

- Processor: - Intel Core iS-7200U up to 3.1GHz
- RAM: - 8 GB for developer evaluation
- Hard Disk: -1 GB
- Operating System: - Linux/Windows 10 (1903,1807)

Topics Covered in training:

- Programming Fundamentals
- Python Basics
- Numpy
- Matplotlib
- Pandas
- Tkinter



Why NumPy and Pandas over regular Python arrays?

In python, a vector can be represented in many ways, the simplest being a regular python list of numbers. Since Machine Learning requires lots of scientific calculations, it is much better to use NumPy's ndarray, which provides a lot of convenient and optimized implementations of essential mathematical operations on vectors.

Vectorized operations perform faster than matrix manipulation operations performed using loops in python. For example, to carry out a $100 * 100$ matrix multiplication, vector operations using NumPy are two orders of magnitude faster than performing it using loops.

What is Pandas?

Similar to NumPy, Pandas is one of the most widely used python libraries in data science. It provides high-performance, easy to use structures and data analysis tools. Unlike NumPy library which provides objects for multi-dimensional arrays, Pandas provides in-memory 2d table object called Dataframe. It is like a spreadsheet with column names and row labels.

Hence, with 2d tables, pandas is capable of providing many additional functionalities like creating pivot tables, computing columns based on other columns and plotting graphs. Pandas can be imported into Python using:

```
1 >>> import pandas as pd
```

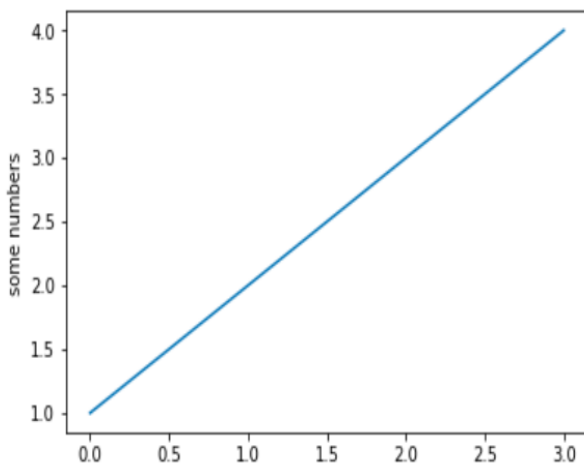
What is matplotlib?

Matplotlib is a 2d plotting library which produces publication quality figures in a variety of hardcopy formats and interactive environments. Matplotlib can be used in Python scripts, Python and IPython shell, Jupyter Notebook, web application servers and GUI toolkits.

matplotlib.pyplot is a collection of functions that make matplotlib work like MATLAB. Majority of plotting commands in pyplot have MATLAB analogs with similar arguments. Let us take a couple of examples:

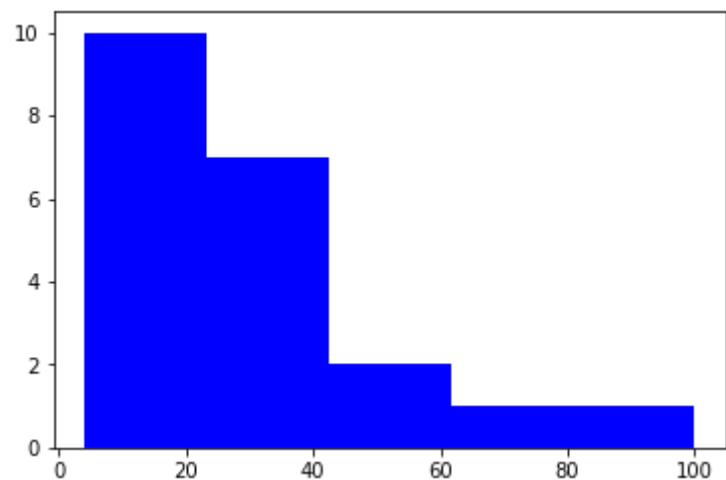
Example 1: Plotting a line graph

```
1 >>> import matplotlib.pyplot as plt
2 >>> plt.plot([1,2,3,4])
3 >>> plt.ylabel('some numbers')
4 >>> plt.show()
```



Example 2: Plotting a histogram

```
1 >>> import matplotlib.pyplot as plt
2 >>> x = [21,22,23,4,5,6,77,8,9,10,31,32,33,34,35,36,37,18,49,50,100]
3 >>> num_bins = 5
4 >>> plt.hist(x, num_bins, facecolor='blue')
5 >>> plt.show()
```



Project: Object detection using TensorFlow and imageai module in python.

Source code:

```
from imageai.Detection import ObjectDetection

import os

import numpy as np

import matplotlib.pyplot

execution_path = os.getcwd()

detector = ObjectDetection()

detector.setModelTypeAsRetinaNet()

detector.setModelPath( os.path.join(execution_path , "resnet50_coco_best_v2.0.1.h5"))

detector.loadModel()

detections = detector.detectObjectsFromImage(input_image=os.path.join(execution_path ,
"image.jpeg"), output_image_path=os.path.join(execution_path , "imagenew.jpg"))

#print(detector.detectObjectsFromImage(input_image=os.path.join(execution_path ,
"image.jpeg"), output_image_path=os.path.join(execution_path , "imagenew.jpg")))

ct=0

temp=0

list001=[]

for eachObject in detections:

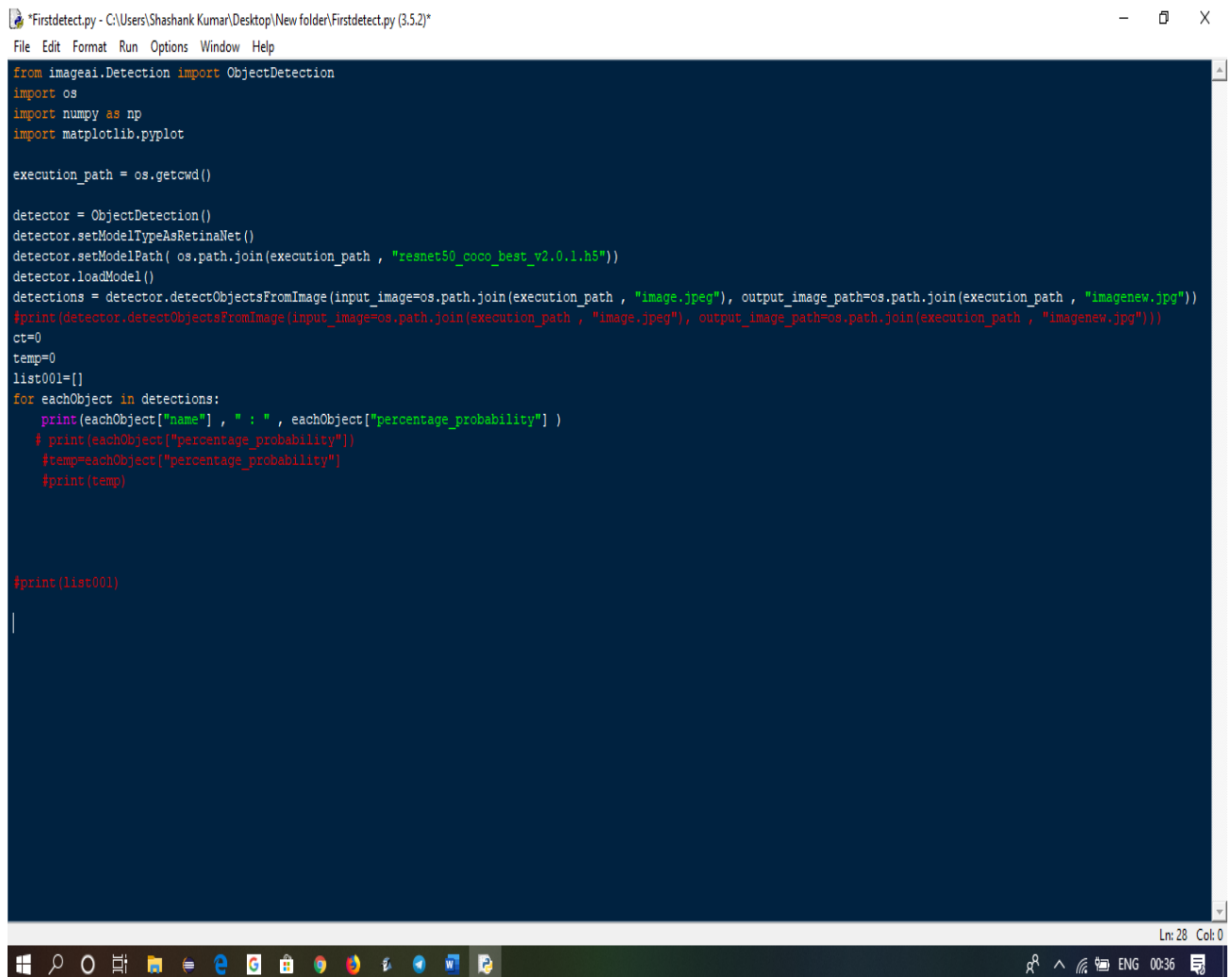
    print(eachObject["name"] , " : " , eachObject["percentage_probability"] )
```

```
# print(eachObject["percentage_probability"])
```

```
#temp=eachObject["percentage_probability"]
```

```
#print(temp)
```

```
#print(list001)
```



The screenshot shows a Windows desktop with a text editor window titled "Firstdetect.py - C:\Users\Shashank Kumar\Desktop\New folder\Firstdetect.py (3.5.2)". The editor contains a Python script for object detection using the imageai library. The script imports necessary modules, sets the execution path, loads a ResNet50 model, and processes an image to detect objects. It then iterates through the detected objects, printing their names and percentage probabilities. The script concludes by printing a list of the detected objects. The Windows taskbar at the bottom shows various application icons and the system clock indicating 00:36 on 08/08/2024.

```
from imageai.Detection import ObjectDetection
import os
import numpy as np
import matplotlib.pyplot

execution_path = os.getcwd()

detector = ObjectDetection()
detector.setModelTypeAsRetinaNet()
detector.setModelPath( os.path.join(execution_path , "resnet50_coco_best_v2.0.1.h5"))
detector.loadModel()
detections = detector.detectObjectsFromImage(input_image=os.path.join(execution_path , "image.jpeg"), output_image_path=os.path.join(execution_path , "imagenew.jpg"))
#print(detector.detectObjectsFromImage(input_image=os.path.join(execution_path , "image.jpeg"), output_image_path=os.path.join(execution_path , "imagenew.jpg")))
ct=0
temp=0
list001=[]
for eachObject in detections:
    print(eachObject["name"] , " : " , eachObject["percentage_probability"] )
    # print(eachObject["percentage_probability"])
    #temp=eachObject["percentage_probability"]
    #print(temp)

#print(list001)
```

Test inputs

Input:



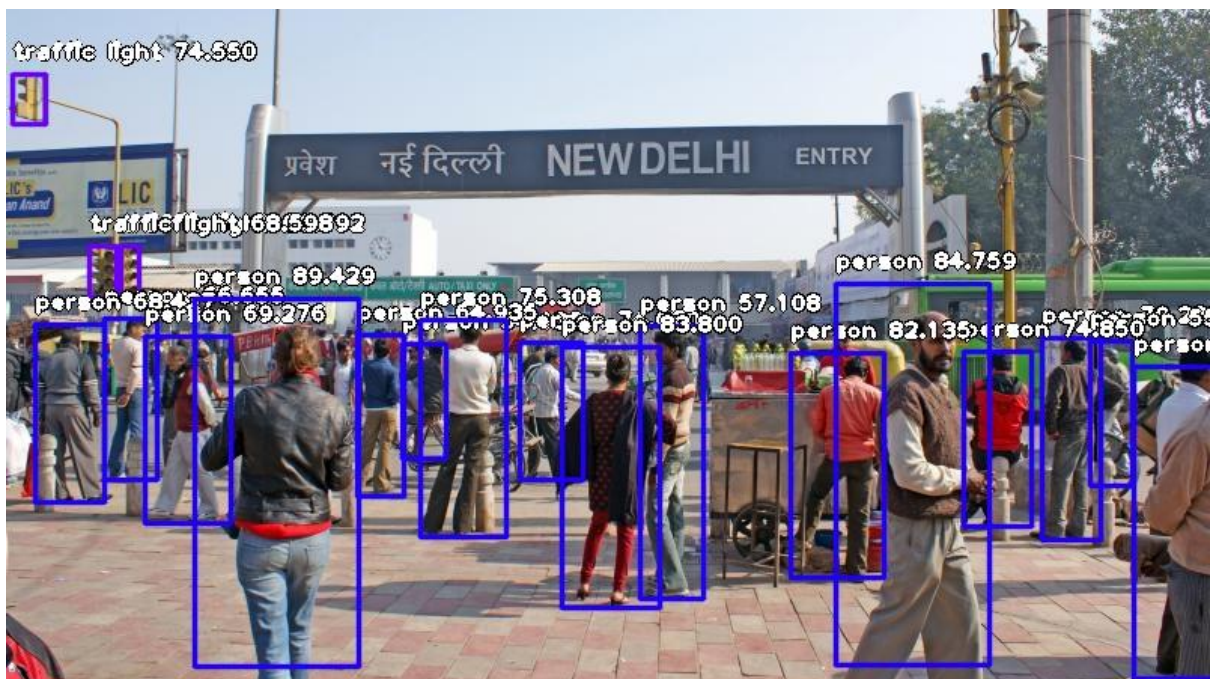
Output:



Input:



Output:



Module Used: ImageAI

ImageAI is a **python** library built to empower developers, reseachers and students to build applications and systems with self-contained Deep Learning and Computer Vision capabilities using simple and few lines of code.

Future Scope:

With Some more effort in modification of code the project can be elevated to be in use for user friendly input by designing a UI and and also the input can be from live cameras or ongoing mp4 clips where it can detect the objects and produce the results

Instead of taking input only from static images as of how it is working now.



Probability Results Generated for an image:

traffic light : 74.55009818077087

traffic light : 60.792481899261475

traffic light : 68.59811544418335

person : 66.65471196174622

person : 54.43474054336548

person : 74.07011389732361

person : 64.9347186088562

person : 79.23179268836975

person : 55.39653301239014

person : 74.85049366950989

person : 68.42652559280396

person : 69.27569508552551

person : 75.30776262283325

person : 82.13503360748291

person : 57.108426094055176

person : 83.79958868026733

person : 50.35324692726135

person : 89.42949771881104

person : 84.7585916519165

>>>

CONCLUSION

Overall it is really fair to say that it was a really good experience to undergo training at such an esteemed and respected government institute endorsed strongly by Government of India and Ministry of Electronics and Information technology to empower and aware the youth and upcoming generations about the latest technologies and their application in real world and these can be used to solve the real world problems.

Thankyou