Internship Report

**Intern Details:**

|  |  |
| --- | --- |
| Student Name | LOKESH KOSHALE |
| Roll Number | CS15B049 |
| Company of Internship | eClerx |
| Profile | SOFTWARE SERVICES |
| Internship start date | 17 MaY 2018 |
| Internship duration | 2 Months |
| Location | Airoli, Mumbai |
| Designation | Intern |
| Internship Manager | Mr. NItin thakkar |
| Internship PM Coach | XXXXXXX |
| Project Name | **SW\_DMS\_Dev** |
| Team | XXXXXXX |

**Project Overview:**

**Project - I**

Name of the project: Object Detection in Image

Languages used: Python

Frameworks used: Tensorflow, Keras.

Environments used: Python.

Tools/Services used: PyCharm

Libraries used:

* Numpy, Scipy, Sci-kit learn, Scikit-Image, Matplotlib

**Project Progress:**

**Aim:** To detect (draw boxes) and recognize (label) all the desired object in a given image.

**Attempt 1:** Use Deep-Convolutional neural network to extract features of the image and localizing and identifying multiple objects in a single image.

**- approach (i):** Build an image pyramid of various aspect ratio from image and convolve the image pyramid using pre-defined boxes (anchor boxes) and for each box use state of the art neural network architecture (i.e VGGNet, RCNN, Fast-RCNN) to recognizing images (i.e. find the probabilities of the box belonging to classes of object ).

**- approach (ii):** Use single neural network architecture for both generating anchor boxes and classifying them (i.e. Single shot detector) or divide the image in grid and for each grid find the probability of it lying in a anchor box (i.e. YOLO: You only look once)

Keras with tensorflow as backend was used to implement the neural network architecture and trained it using an image database to recognize the objects. The algorithm in approach (i) are harder to train and takes long time than algorithms in approach (ii) but have better accuracy.

**Attempt 2:** Used tensorflow’s object-detection api open sourced by Google, to detect multiple objects in an image. In this approach we get better running time as the network was highly optimized for tensorflow framework.

**Attempt 3:** Instead of recognizing and labeling image we have to only find the possible bounding boxes for the objects in the image. The approach was first to segment the image based on color, texture and edges. After this we have to join similar segments of the image and keep repeating it till threshold and then draw bounding boxes based on the image segments. It was able to detect objects most of the time but its accuracy was lower than neural networks but the running time was considerably very high, and the amount of boxes generated was also very high.

**Learning of the student:**

Technical:

* XXXXXXX
* XXXXXXX
* XXXXXXX
* XXXXXXX

**AESTHETIC**:

* XXXXXXX
* XXXXXXX
* XXXXXXX
* XXXXXXX

**RATIONALE:**

Software Development Life Cycle (SDLC) learnings:

* XXXXXXX
* XXXXXXX
* XXXXXXX
* XXXXXXX

Personal Learnings:

* XXXXXXX
* XXXXXXX
* XXXXXXX

**Non-revelation of confidential Information**

The above report does not contain any confidential information from eClerx .

**Mr. Nitin Thakkar (Project Manager)**