CS308: Machine Learning
Project Report: Vehicle Detection in Aerial Scenes

Gaurav Raj 201951064
Yogeshwar 201951180
Course Instructor Dr Jignesh Bhatt

I. Introduction

The goal is to detect/localize vehicles by drawing bounding boxes around them in the images of the type aerial scenes (e.g. taken using drones or satelites). Currently, the program addresses a simpler problem with only one type of vehicle – airplanes. The dataset mostly consists of images containing parked airplanes from the top view. The technique used for object detection here is R-CNN (Region based Convolutional Neural Network). The annotations contain labels (0 or 1) and the two opposing corners' coordinates of the bounding rectangle.

II. Libraries

The python libraries used in this project are OpenCV (Image processing), Pandas (Dataset usage), Matplotlib (Plotting), Numpy (Calculations), Tensorflow Keras (Load model data augmentation).

III. Dataset

The dataset is a registered dataset with most of the images containing airplanes. The dataset has around 700 image files and corresponding equal number of files containing annotations. The image files are colored images of size 256 x 256 pixels of parked airplanes taken from a height showing their top view. The corresponding annotation files contain the information about the exact location of those airplanes in the whole image. The first row in each file contains the count of airplanes (say, n number of airplanes) present in the corresponding image file. This row is followed by that count (n) many number of rows, where each row contains four values, say x1, y1, x2, y2, which corresponds to the top-left (x1, y1) and bottom-right (x2, y2) corners of the bounding box capturing just the airplane in the whole image.

3
173 15 232 70
49 76 139 153
4 164 93 244

Fig. 1.   : Annotation

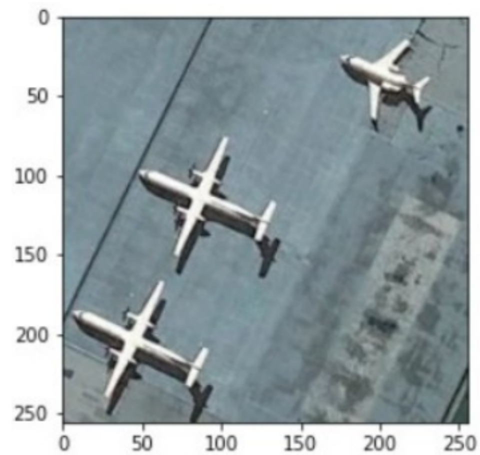

Fig. 2.   : Image in Dataset

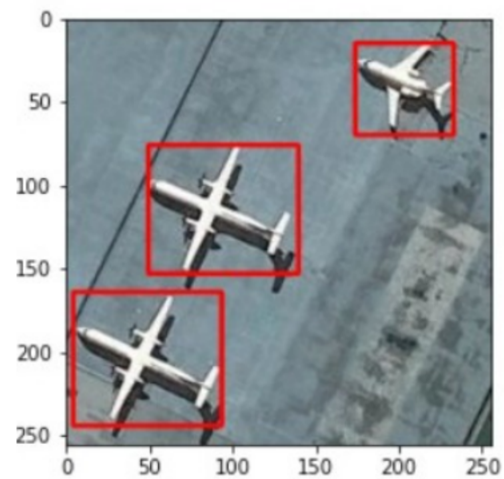

Fig. 3.   : Image with bounding box

IV. Neural Network

The artificial neural network used for this task of locating objects in the image is the RCNN (Region based Convolutional Neural Network). Traditionally, CNNs are used to simply classifly an image, i.e., it outputs whether the image contains an instance of a class, but it doesn't tell where that instance appears in the whole image. The object localization task needed here is accomplished by using CNN on various regions in an image (by using RCNN). This techinique, in a broad sense, consists of generating various proposed regions (candidate bounding boxes) from an image, and selecting the one having the most probablity of containing an airplane, given by the trained probabilistic classifier for this case. This task is preceded by generation of such labeled regions from the training portion of the registered dataset and training the classifier using them.

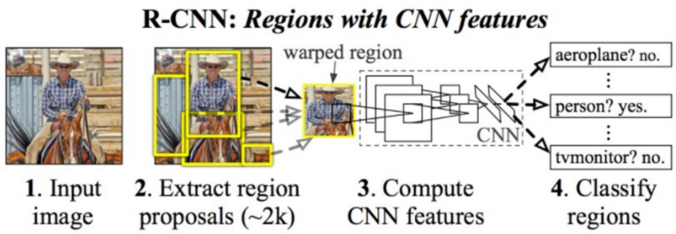

Fig. 4. : R-CNN

V. Training

A. Generation of Proposed Regions

When using RCNN, an image is processed to get many proposed regions which can be considered as various bounding boxes in that image, which might contain some object. For this purpose, image processing technique called Selective Search using OpenCV library has been used to generate around 2000 such proposed regions.

B. Resize and Labelize the Regions

For the training part, we have the registered dataset, i.e. images along with the location of the bounding boxes. The area confined by these bounding boxes in an image (of airplanes) is considered as the ground truth regions. The proposed regions generated through Selective Search are then compared with these ground truth regions to measure their similarity. The measure here is called Intersection Over Union (IOU) (next section). Through this, the regions having enough similarity with any ground truth region are considered as positive samples (containing an airplane) and are labeled 1. On the contrary, the regions having enough disimilarity with all ground truth regions are the negative samples (absence of airplane) and are labeled 0. At most, 30 samples from both, positive and negative category, are picked from an image and labeled either 0 or 1. These regions are further resized to the dimensions of 224 x 224 pixels for uniformity and to comply with the neural network used later in the project.

C. Intersection Over Union

IOU is used as a measure of similarity/overlap between two regions in this project. Here, both the regions are rectangular – first one being the bounding box of the ground truth, and the second one being that of the proposed region, both of which need not be identical in size. IOU value is calculated as the ratio of the area of intersection and the area of union of both the regions. IOU value increases when the size of the proposed region approaches that of the ground truth region and also requires maximum overlap between both the regions. Following this, the IOU value is the highest (= 1) when both the boxes are identical and completely overlap, and lowest (= 0) when there is no overlap between the two.

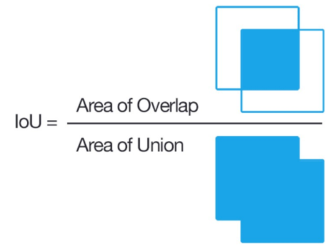

Fig. 5. : Intersection over Union

Fig. 6. : Comparison between IOU value

VI. References

1.)http://www.escience.cn/people/JunweiHan/NWPU-RESISC45.html
2.)https://towardsdatascience.com/step-by-step-r-cnn-implementation-from-scratch-in-python-e97101ccde55
3.)https://pyimagesearch.com/2016/11/07/intersection-over-union-iou-for-object-detection/
4.)https://www.cs.toronto.edu/~frossard/post/vgg16/