EE452 Computer Vision (Spring 2022)

Assignment 3

Object Detection Algorithm

Release date:- 23 April 2022

Due date:- 30 April 2022

Maximum Marks: 100 (Weight in Final grade: 4%)

Instructions:

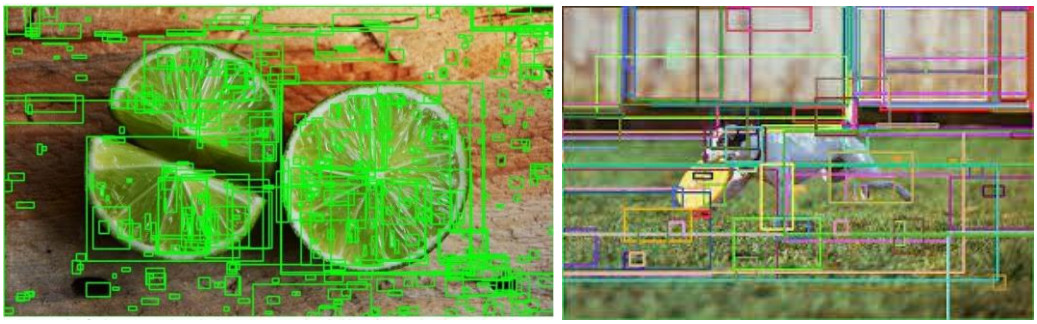
- A report is to be submitted on LMS describing the different phases of the work also including what worked, what did not work, what you learned, what extra things you tried and how to proceed with this etc.
- The assignment can be done individually but the students are recommended to do it in pairs.

Motivation: Let's dive into the world of object detection, it has vast applications covering the domain of video surveillance, autonomous vehicles and monitoring etc. As you have already implemented convolutional neural networks, applied transfer learning and now have a good understanding of basic machine learning classifiers, you are already equipped with all the vital tools. All you need to learn now is how to combine them together for object detection.

Background of RCNN:

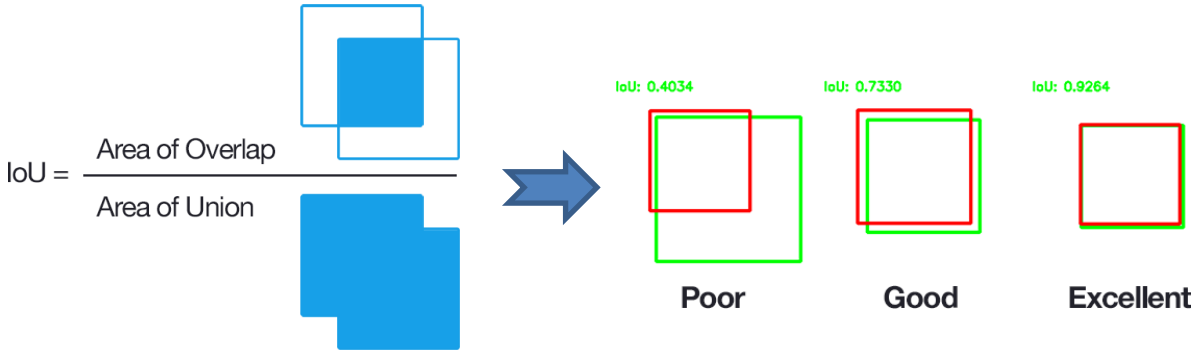
R-CNN stands for Regions with CNN. In R-CNN we pass the image through a selective search process and identify 2000 regions. The resulting regions are then fed into the classification phase. So rather than going through a large number of regions for classifying we take into account only 2000 regions. The very reason that R-CNN is called a 2 stage-detector is that it first proposes regions and then these regions are used for classification phase. There are four fundamental steps in case of an R-CNN model.

1. Pass the input images through the process of selective search in order to generate region proposals. The example of selective search process on an image is provided below. You can see that how well the bounding boxes are placed to cover all possible objects of various size in the image.



2. Once we have all the region proposals in the image, we will then calculate IOU (intersection over union) on proposed region with the ground truth data and then add label to the proposed regions.

You might be wondering what is intersection over union: It is basically a metric used to measure the accuracy of an object detector. So, you have your ground truth region which is the true or actual region highlighting the detected object and then you have the predicted regions. You then compute the ratio of the areas of intersection of the regions with area of the union of the regions. Which can be viewed in the diagram provided below,



3. Once we have the regions highlighted in the images along with the ground truth values then we make the input images go through a pre trained model. For our assignment we recommend VGG16 though you are welcome to use any other you like. Rest of the process is simple stacking of some convolution layers on the pre trained model.

4. Once you have completed your training phase, you will then test your model again the same way. You will take proposed 2000 regions and pass them through the trained network. And predict the class of those regions.

Tasks:

Question 1 [30]: Theoretical Background

Note: These questions are meant to make you familiar with the fundamental idea of object detection. Please do not answer these in more than a few lines.

1. What is object detection?
2. What is input and output in object detection?
3. What is the difference between image recognition and object detection?
4. What is a single stage detector?
5. What is a two-stage detector?

Question 2 [35]: RCNN Implementation

1. In order to complete this question, this tutorial is referred and should be followed.
2. Analyze the dataset that is used in the tutorial and present a short note (3-4 lines) summarizing your observations.
3. Initialize selective search using the following command: cv2.ximgproc.segmentation.createSelectiveSearchSegmentation().
4. Observe the function "get_iou" to calculate IOU (Intersection Over Union) of the ground truth box from the box computed by selective search. Comment on how the algorithm is working.
5. Preprocess the images using the selective search as a base for each image and propose the regions for the RCNN model.
6. To prepare your network, import any existing network with pretrained weights. Fine tune your model further as per the requirements. Note: Here, the tutorial follows with TensorFlow model however, you are not restricted to use PyTorch either. Train the model!
7. Test out the predictions done by the model and visually explain its performance.

Question 3 [35]: Faster RCNN

1. Search on what is a faster RCNN and summarize the findings in 3-4 lines.
2. In order to do its implementation, this tutorial is referred and should be followed. The tutorial has also shared its notebook so that can be used as well.
3. Run the shared notebook and explain the function being performed in each cell. Try to give an in-depth explanation for this part (6-7 lines for each cell).
4. Explain how the model was finetuned and what changes it brought to the overall results in its testing.

Submission Guidelines:

The entire assignment is to be done as a python notebook. Once you are done, you should upload the notebook to LMS. Create sections within the notebook in order to answer each question.

It is preferred that you do this assignment on google colab: https://colab.research.google.com/. Colab allows you to utilize its GPU, which can be accessed as follows: Runtime => Change runtime type => Change Hardware accelerator to GPU. Once you have completed your assignment on colab, you can download it as a notebook as follows: File => Download.ipynb.