Indian Institute of Technology Kanpur
CS783: Visual Recognition, 19–20

ASSIGNMENT 3

Student Name: Gurpreet Singh, Kamlesh Kumar Biloniya
Roll Number: 150259, 160317
Date: February 24, 2019

1. Problem Statement

Our task is to detect bounding boxes using the standard sliding window approach with classification using ResNet18. The steps for the same include correct data loading and pre-processing. We are given a dataset of images wherein each image has a corresponding annotation file containing location of the bounding boxes in that image. We have to extract and save them with corresponding label. We are asked to use two approaches for object detection (one layer and two layer detection)

2. Data and Classification

2.1. Data Loading and Pre processing

The dataset consists of images from 20 classes including detection annotations, however for the purpose of this assignment we needed only three classes. We extracted all the bounding boxes from each image at the given locations and stored them in corresponding class folder for our training data. As locations of background patches was not given, we extracted one patch from each image (but maximum 2500) if all intersections over unions with the 3 classes was less than 0.1. Although we were suggested to threshold on the intersection over unions with all 20 classes, we found this approach to work better overall.

Moreover, we observed a high bias in our model for classifying images as chairs. In order to avoid this, we reduced the number of chair images and limited them to 500.

2.2. Classification

For the purpose of classification using a single layer, we initially fine tuned a pretrained ResNet18 model for classification replacing the last fc layer to fit our classification model. However, this gave inaccurate classification results due to the high variability in our images. In order to fix this underfitting, we allowed for full training of the resnet model post fine-tuning. The fine-tuning was run for 25 epochs and the complete training was run for 50 epochs.

The changes for the two-layer were prominent only in the forward propagation code wherein the output from an intermediate layer (we used the output from the 2nd convolution block from the ResNet model) is passed through a MaxPool layer before concatenating with the final output which is fed to a fc layer for final classification. This approach marginally improved the mAP values as noted later in the report.

3. Detection

3.1. Sliding Window

For detection, we use the sliding-window method with windows of different aspect ratios and scales. Due to limitation of time and resources, we were not able to test many different window sizes. The window sizes we used are as listed -> (100, 100), (100, 250), (250, 100), (200, 200), (400, 400)

For each window, the trained resnet model is used to classify the image (or patch) among the three classes and an additional class "background". A bounding box is assumed if the predicted class is the "background" class. In order to remove noise, we threshold on two things, the probability of the path being a background patch and the probability of it being an object (among the 3 classes). If the probability of it being a background is more than 0.3, it is discarded and if the probability of it being an object is less than 0.8, it is again discarded.

The stride for the same is set to be half the minimum dimension of the window size.

3.2. Non-Maximum Suppression

We used **Non-maximum suppression(NMS)**method to remove multiple detection of same object. We gave priority to box with maximum score and kept boxes with intersection over union value above a threshold which is 0.1

4. Testing and Accuracy Calculation

We used AP(Average Precision) and mAP (mean Average Precision) as our metric to measure the accuracy of the object detectors. These are recorded in table 1. The mAP is computed using the following formula.

$$AP = \frac{1}{K_c} \sum_{k=1}^{K} \frac{\mathcal{I}\left(pred_k = class_k\right)}{k} \sum_{i=1}^{k} \mathcal{I}\left(pred_i = class_i\right)$$

where K_c is the total number of correct predictions.

Essentially, we compute AP as the average of the precisions calculated at each recall point, r. The mAP for object detection is the average of the AP calculated for all the images. The order of predictions is decided by score given to the box (probability of that object).

5. Results

Bounding box color scheme: We are using following color scheme to show the bounding boxes

1. aeroplane: black
2. bottle: blue
3. chair: green

Results of our models on test data can be seen in table 1.

class	**mAP score (single layer)**	**mAP score(2 layer)**
aeroplane	0.049769	0.056391
bottle	0.057549	0.050636
chair	0.219666	0.223850

Table 1: mAP scores on Different Methods