Object detection (autorickshaw) in images

Devendra Pratap Yadav
2014CSB1010

Assignment 4 - CSL462
IIT Ropar

Object detection is one of the classic Computer Vision problems. Many algorithms focusing on accuracy and speed have been proposed using traditional Computer Vision techniques as well as Machine Learning [2]. A direct application of object detection is autonomous vehicles. However, most of the datasets available are specific to western countries. We attempt to learn a object detection for autorickshaws, which are commonly found in India, but not western countries. We will use sliding window based approach to identify a bounding box for objects. We calculate the accuracy using Intersection over Union.

1 Dataset

We use the "Autorickshaw Detection Challenge" dataset for developing and evaluating our approach. The dataset consists of 800 images taken on streets consisting of one or more autorickshaws. We use 600 images for training our classifier and remaining 200 for testing and evaluating our method. A set of rectangular bounding boxes is given for each image. We need to predict the location of the bounding boxes using our classifier.

2 Object detection methodology

We implement a sliding window based approach to find square windows containing the object. We first extract square blocks from image, compute HOG [1] features and train a classifier to predict if it contains 'autorickshaw' or not. Then, we take sliding windows at various scales and find candidate bounding boxes. These boxes are thresholded by classification accuracy and non-maximal supression is applied to remove duplicates. We also calculate a heuristic based score for each box and remove overlapping boxes with low score or accuracy. Finally, predicted boxes are returned.

2.1 Generating training samples

We use the first 600 images to generate patches for training our classifier to detect if 'autorickshaw' is present in current image window or not. Since there exists great variation in shape of bounding rectangles, we would need to consider large number of shapes for our sliding window, which will be very slow. We consider a simpler problem of predicting a square bounding box.

Positive samples (autorickshaw present and bounded in region) : We add some random translation to bounding box and extract image portion.

Negative samples (autorickshaw partially/not present) : We randomly consider square regions in image in varying scales and extract image portion. However, if such region has high overlap with an autorickshaw's bounding box, we remove it.

2.2 Training Neural Network

We consider a binary classification problem i.e. positive and negative classes. Positive means presence of bounded autorickshaw. We collect around 9000 positive images and 14000 negative images.

The algorithm steps are as follows:

- Resize all images to 64x64 size.
- Compute 1764 length HOG feature vector for each image.
- Create a Neural Network with two hidden layers of size 100 and 50.

Input : 1764 HOG features,

Output : 2 values denoting probability of each class

- Train NN with training samples and store it.
- Provide HOG features of any 64x64 image to classify it.

We test this NN on 6000 unseen patches and obtain accuracy of **81.2%**.

2.3 Sliding window algorithm

We use a sliding window based algorithm to detect object in new image. The algorithm steps are as follows:

- Resize image to be smaller than 512 pixels in any dimension.
- Compute image at various scales. Eg. 10%, 20%, 30% ... of original size.
- At each image scale, take a 64x64 sliding window and move it throughout the image.
- Extract HOG features for current window and classify it using trained Neural Network.

Figure 1: Autorickshaw Detection Challenge" dataset with ground truth shown in green boxes.

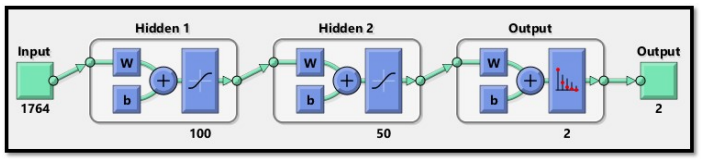

Figure 2: Neural Network architecture

- Store bounding box corresponding to current window along with probability of 'autorickshaw' being present.
- Remove all bounding boxes having probability lower than a threshold probability (Eg. 0.8).
- Perform non-maximal suppression and heuristics to remove duplicate and highly overlapping boxes.
- Return the list of bounding boxes for detected objects.

3 Results

We compare the predicted bounding box with the ground truth using **Intersection Over Union (IOU)** measure. It measures the amount of overlap between the bounding boxes. Since corresponding boxes are not known, for each true box, we find the predicted box with maximum IOU value. This is average for all true boxes to obtain IOU score for one image. The average IOU for 200 test images in dataset ig given below:

$$\textbf{IOU = 0.3183} \tag{1}$$

Some sample object detection results are shown in Figure 3,4,5. We observe that large size images with single autorickshaw and no occlusion provide the best results. Despite predicting a square bounding box, we obtain good overlap for these images. However, images with large number of autorickshaws, and other vehicles pose a problem for detection. Spurious and missed detections are common in such images. This is often due to occlusion of part of autorickshaw. Overall, we achieve good detection at various scales for isolated autorickshaw, whereas improvement is needed for detecting multiple objects. State-of-the-art methods employ R-CNN which directly predicts coordinates of bounding boxes. This reduces chances of error due to sliding window classification and non-maximal suppression.

[1] Navneet Dalal and Bill Triggs. Histograms of oriented gradients for human detection. In *CVPR*, pages 147–151, 2005.

[2] Joseph Redmon, Santosh Kumar Divvala, Ross B. Girshick, and Ali Farhadi. You only look once: Unified, real-time object detection. *arXiv*, 2015. URL `http://arxiv.org/abs/1506.02640`.

Figure 3: Correctly predicted bounding boxes(red) vs ground truth(green).

Figure 4: Partially correct predicted bounding boxes(red) vs ground truth(green). Some boxes are missing or spurious boxes are predicted.

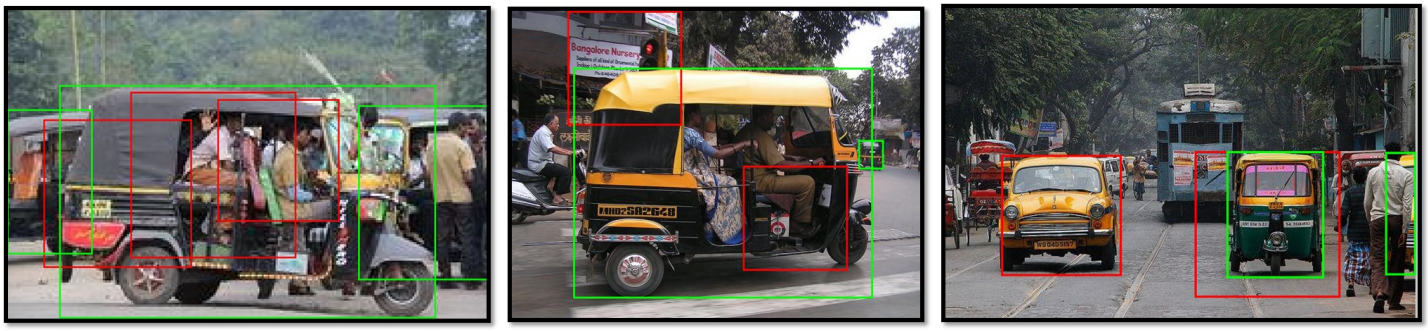

Figure 5: Incorrectly predicted bounding boxes. Part of actual object is detected or incorrect object is detected.