Interview Question:

Paul Estano
Data management and analysis

May 12, 2019

code repository :

`https://github.com/paulesta55/TrafficSign2`

1 Question 1

We filter the predictions to find frames where the network detected a 'pn' sign. Then we use *LabelImg* (`https://github.com/tzutalin/labelImg`) to add annotations to related frames.

Since LabelImg generates one XML file per frame (under /ex2/xml in the repository), we parse each XML file and add the corresponding labels to an "updated ground truth" file (ex2/gtUpdated.txt in the repository).

2 Question 2

To analyze the results we first compute the number of true positives, false positives and false negatives.

To match a prediction with a specific ground truth we compute the intersection over union (IoU) value of the predicted bounding box and every potential ground truth bounding box of the related image. The selected ground truth box is then the one with the highest IoU. If this IoU is greater than a specific threshold (here 0.5) this match is a true positive. If a sign does not match any ground truth box then it is considered as a false positive. Then the ground truth bounding boxes which are not matched with any prediction are considered as false negatives.

To compute the IoU we only have to compute the following formula :

$IoU = \frac{AreaofOverlap}{AreaofUnion}$

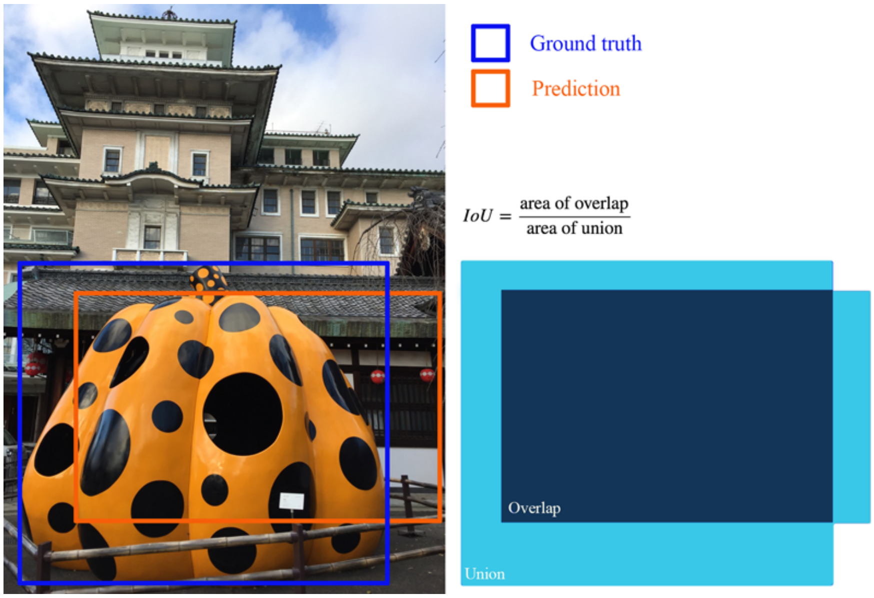


Figure 1: IoU Definition

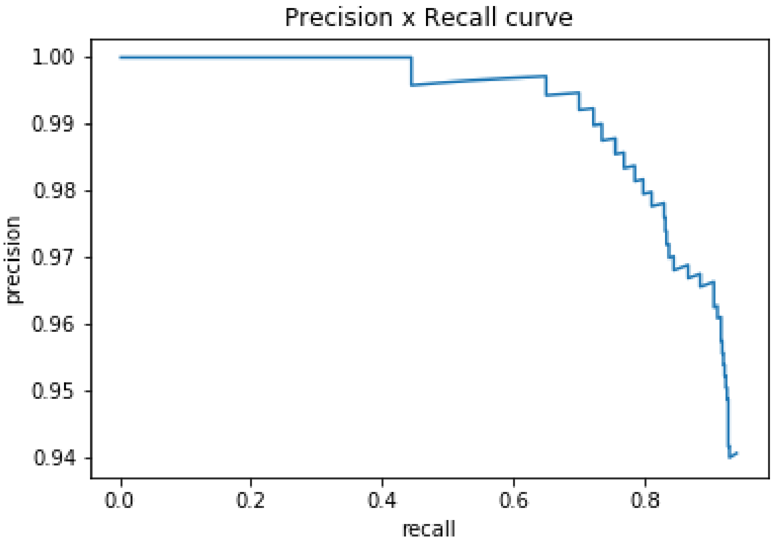
We find a total of 508 true positives, 32 false positives and 63 false negatives out of 540 predictions with an intersection over union threshold of 0.5.

For every prediction we use the current *detection_confidence* value as a detection threshold to compute the recall and the precision values using the following formulas :

$$Precision = \frac{TP}{TP+FP}$$

$$Recall = \frac{TP}{TP+FN}$$

With the recall and precision values we computed we can display the precision x recall curve.



Then we use 2 methods to compute the average precision of the system: Area under Curve-AUC (PASCAL VOC20102012) and 11-point interpolated method (PASCAL VOC2008).

2.1 11-point interpolation

To compute the average precision using the 11-point interpolation method we average the precision for 11 recall values. We interpolate each precision value by taking the maximum precision value among the points which have a greater recall value than the current recall value.

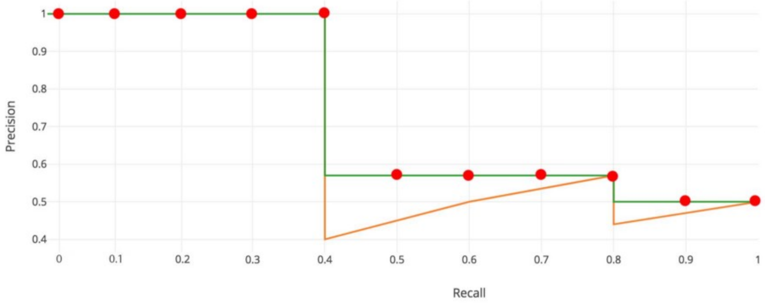


Figure 2: Example of 11-point interpolated method to compute AP

Then we have :

$$AP = \frac{1}{11} \times (AP_r(0) + AP_r(0.1) + ... + AP_r(1.0))$$

2.2 Area under curve AUC or interpolation for all points

Here we only sample the precision value and the recall value when the precision decreases. Hence we approximate the curve by squares to compute the AP.

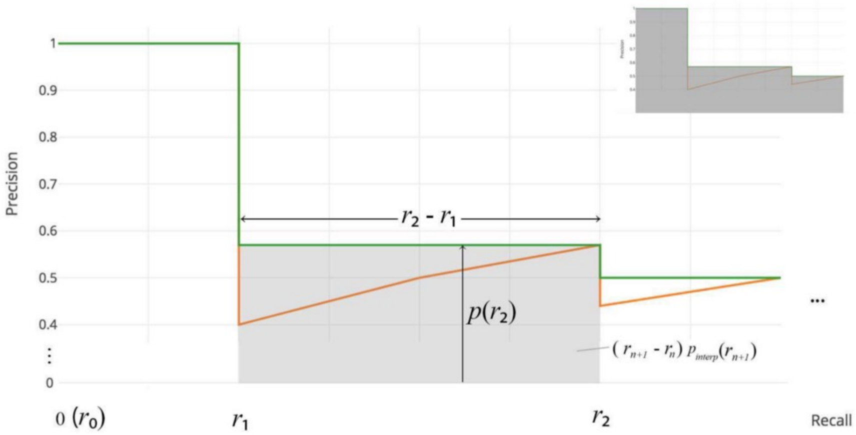


Figure 3: Example of AUC method to compute AP

Then we have :

$$AP = \Sigma(r_{n+1} - r_n)p_{interp}(r_{n+1})$$

$$\text{with } p_{interp}(r_{n+1}) = max[p(r)] \text{ for } r \geq r_{n+1}$$

2.3 Results

With the 11-points interpolation method we find an average precision of **90%**.

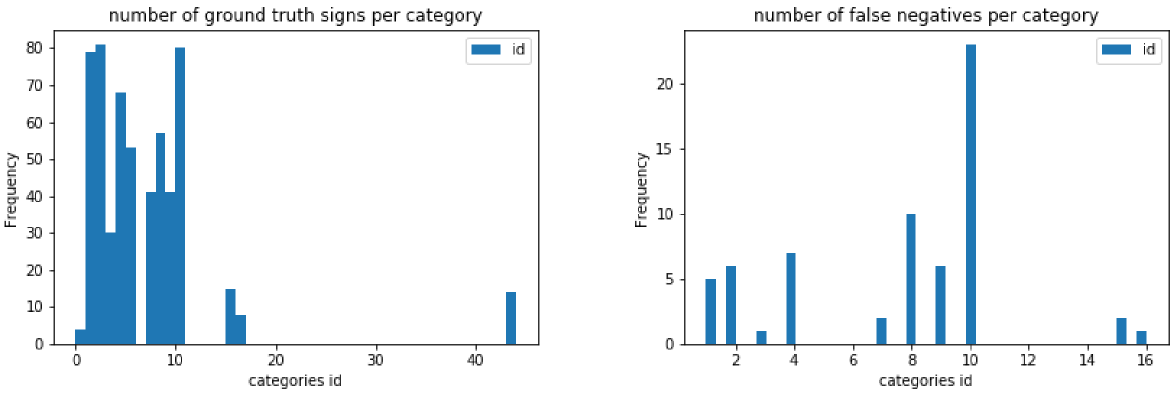
With the AUC method we find an average precision of **93%**.

The 11-points interpolation is usually considered less accurate than the AUC method. Hence, the score is not perfect but is acceptable.

2.4 Problematic Cases

2.4.1 False Negatives

Comparing the red round circle categories distribution in the ground truth signs with the categories distribution in false negatives we can see that the 10th category is over represented among the false negatives. Indeed, other categories are equally or more represented than this one among the ground truth but they are way less represented among the false negatives.



(a) Red round circle categories distribution in the ground truth signs

(b) Red round circle categories distribution in false negatives

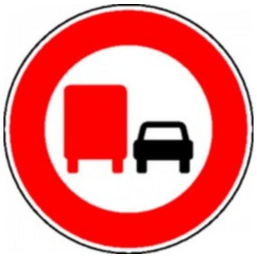
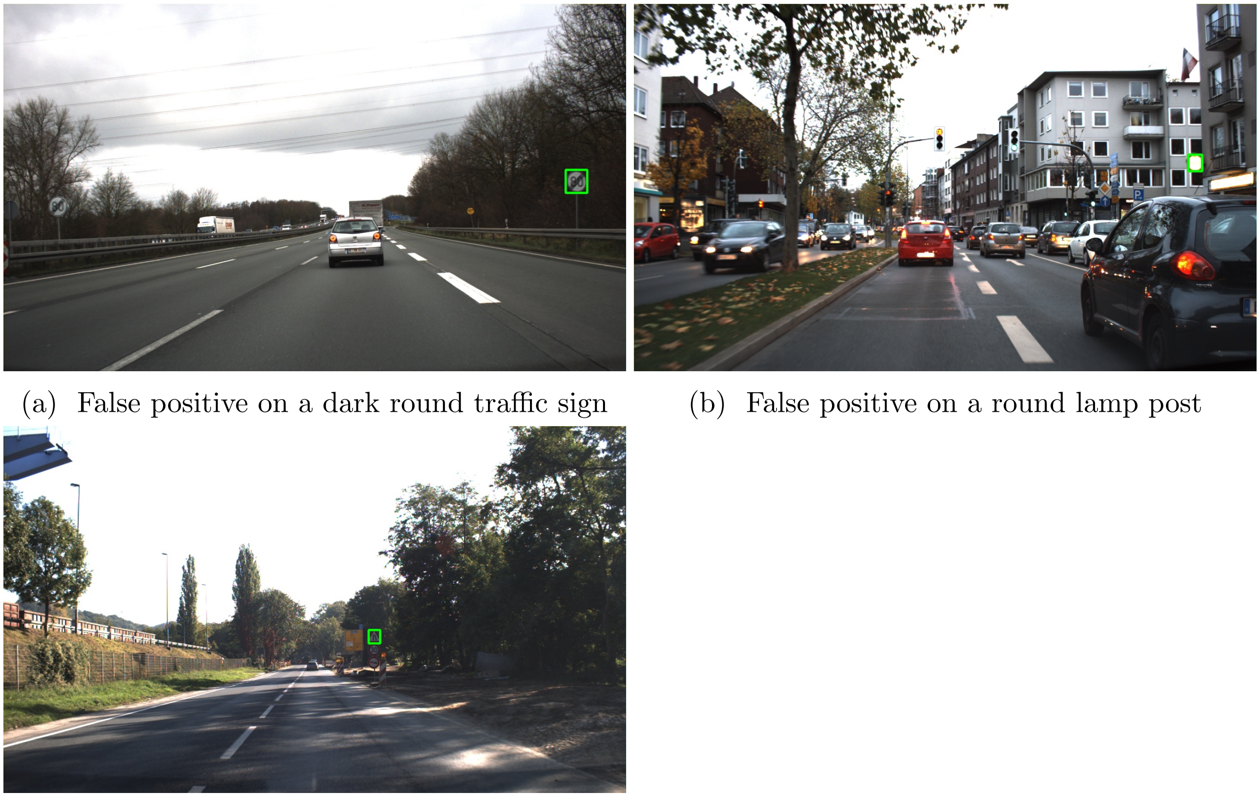


Figure 5: 10th category traffic sign

2.4.2 False Positives

Most of the false positives are mainly generated by 3 types of object :

- Round and not red traffic signs
- Round and red/orange objects
- Red traffic signs which are not round (stop signs or triangular traffic signs)



(a) False positive on a dark round traffic sign

(b) False positive on a round lamp post

(c) False positive on a red triangular traffic sign

2.5 Improvements

2.5.1 False Negatives

First of all, a good solution to decrease the number of false negatives for a specific category would be to add more examples of this category in the training set. Hence, we could add some more examples of the 10th category in the training set.

2.5.2 False Positives

Since most of the false positives are generated by the same types of objects we could add examples of these objects in the training set under a same 'background' category. We could also add random background objects which don't overlap with foreground objects in this category. This process should make the network more robust against false positives.

Moreover, just adding more images which don't contain any red round circle could also make the network more robust against false positives.

2.6 New Annotations

Since we added manually the ground truth for the categories 43 and 44 based on the predictions, it is impossible to evaluate the number of false negatives or the number of true positives for these categories.

Indeed, we introduced annotations for objects which were already detected by the network. Hence, we only artificially improved the results.

Therefore, this technique could be used to label faster some images but to train efficiently a network using manually labelled data will likely be better.