The report for Challenge 2

Group 20: Haineng Huang, Yanchen Liu, Alonso Buitano Tang
Andrew ID: haineng, yanchen2, abuitano

All videos for this challenge can be found at:
https://drive.google.com/drive/folders/1x3ULtVteiVKssjxk31R8cR-v-eS7DE3o?usp=sharing
All textures with different patches can be found at:
https://drive.google.com/drive/folders/1mpJKvSEV44C-7MK1XTiRqo0ppIqeIRo_?usp=sharing

1 Exercise one

Illustrate and analyze the difference of performance between two models in IoU curves (w.r.t. timesteps) and AP metrics.

1.1 mAP Metrics Mesults:

1.1.1 The result of Fast R-CNN :

environment	env1	env2	env3	env4	mean	std
stopsign	0.213	0.407	0.572	0.508	0.425	0.157
stopsign1	0.176	0.356	0.408	0.353	0.323	0.101
stopsign2	0.179	0.335	0.544	0.390	0.362	0.150

Table: the mAP of AP@[0.5:0.05:0.95] under 4 environments of 3 stop sign textures, for Fast R-CNN model, with mean and standard deviation

1.1.2 The result of YOLO v5:

environment	env1	env2	env3	env4	mean	std
stopsign	0.486	0.695	0.783	0.795	0.690	0.143
stopsign1	0.406	0.530	0.728	0.709	0.593	0.154
stopsign2	0.363	0.528	0.713	0.653	0.564	0.155

Table: the mAP of AP@[0.5:0.05:0.95] under 4 environments of 3 stop sign textures, for YOLO v5 model, with mean and standard deviation

1.2 Illustration and analysis of the two index

Illustrate and analyze the difference of performance for two models

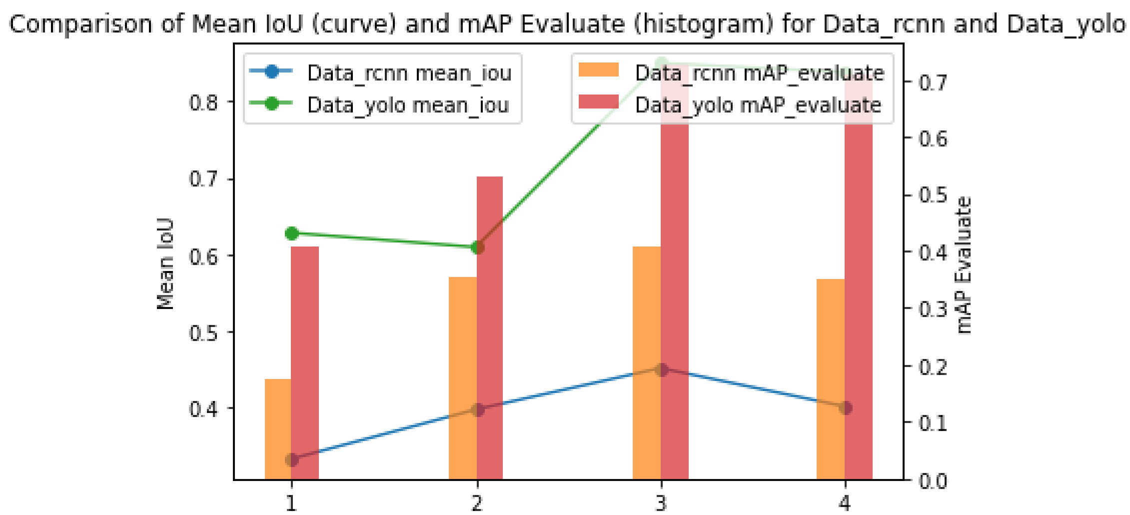

Figure: comparison between two model for stopsign_1

AP Metrics:
Faster R-CNN is known for its high accuracy, and it often achieves better AP scores than one-stage detectors, especially on datasets with small objects or complex scenes. The two-stage process allows it to have a more focused and accurate detection, which contributes to higher AP scores.

IoU Curves:
The IoU curves for YOLOv5 are generally more fluctuating than those of Faster R-CNN, as YOLOv5 sacrifices some localization accuracy for faster inference. However, YOLOv5 still performs well in many scenarios and has been improved significantly compared to previous versions of YOLO in terms of IoU.

2 Exercise two

Include your customized patch, and how you attach them on the stop sign texture, report the AP@[0.5:0.05:0.95] results under two object detection models.

2.1 how you attach them on the stop sign texture

We get our customized patch (a screenshot of bike sign) from Pennsylvania Driver's Manual, resize it to 200*200 pixels and then attach it to the center of the stop sign texture by OpenCV Python package.

```
import cv2
mask = cv2.imread('mask.jpg')
sign = cv2.imread('stopsign.jpg')
resized_mask = cv2.resize(mask, dsize=(200, 200))
sign[210:410, 210:410] = resized_mask
cv2.imwrite('new_image_bike.jpg',sign,[cv2.IMWRITE_JPEG_QUALITY,100])
```

Figure: the screenshot of Pennsylvania Driver's Manual, the bike sign, and the attached texture

2.2 AP@[0.5:0.05:0.95] results under two object detection models

environment	1	2	3	4	mean	std
RCNN	0.185	0.376	0.483	0.383	0.357	0.124
YOLO	0.440	0.631	0.714	0.768	0.639	0.143

Table: the mAP of AP@[0.5:0.05:0.95] under 4 environments for two models, with mean and standard deviation

3 Exercise three

Give two groups of comparison (in the form table or plot) to evaluate the impact of geometric transformations to AP@[0.5:0.05:0.95] under two object detection models.

3.1 Patch Size ([50, 100, 150, 200]) difference

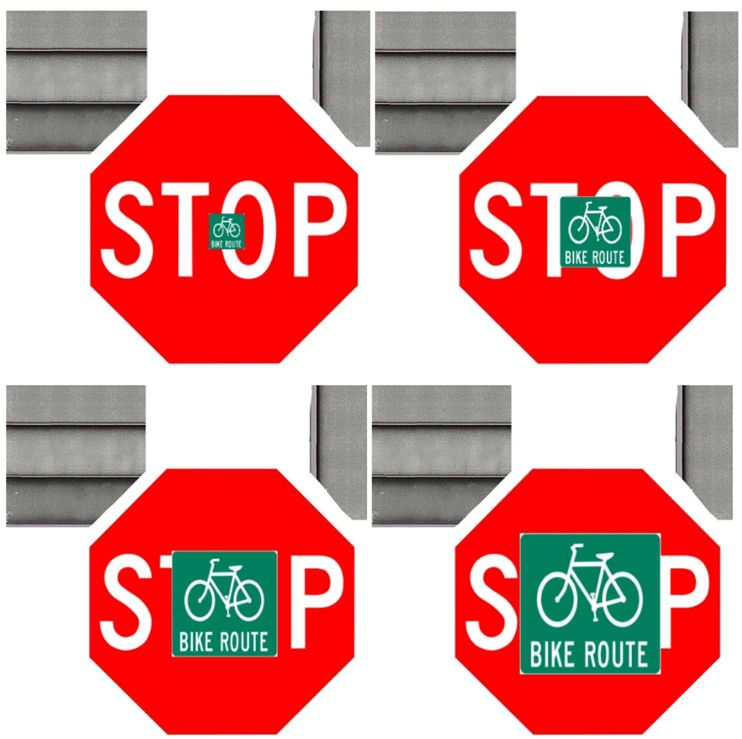

Figure: the textures with different size path, in [50,100,150,200]

3.1.1 The mAP comparison for different patch size on FAST R-CNN

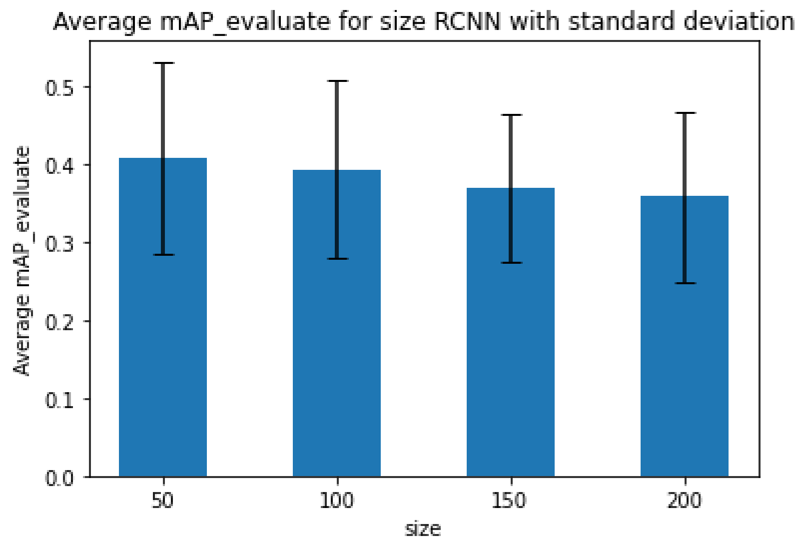

Figure: Bar plot of the Faster-RCNN mAP for each size increment of the patch.

3.1.2 The mAP comparison for different patch size on YOLOv5

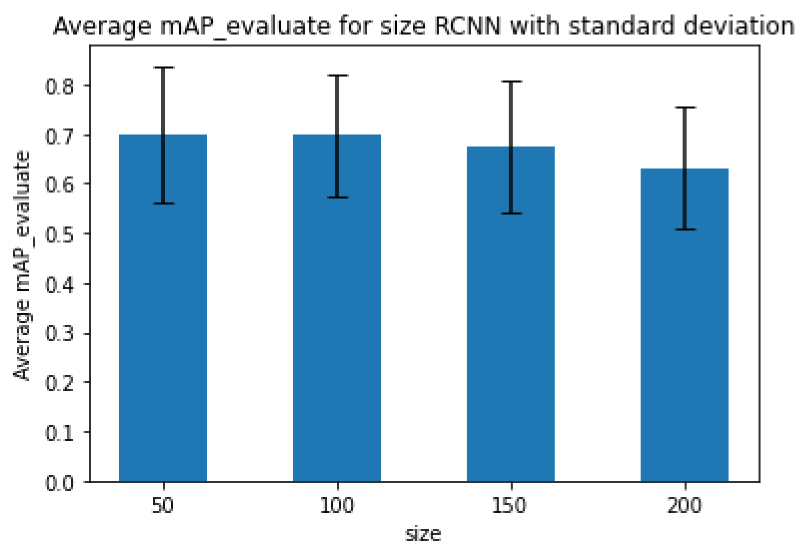

Figure: Bar plot of the YOLOv5 mAP for each size increment of the patch.

3.1.3 Comparison of both object detection models:

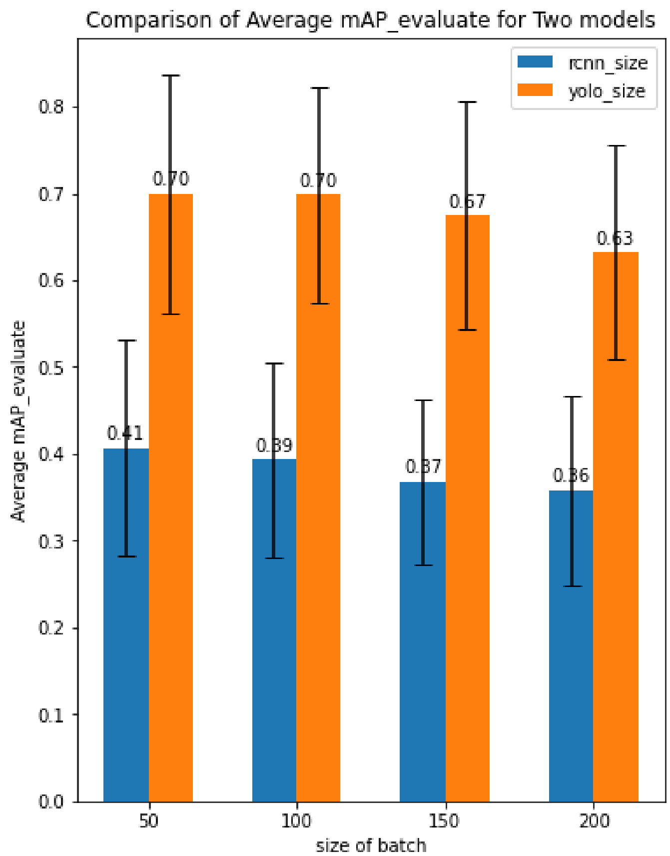

Figure: Bar plot of the mAP of both object detection models for each patch size increment

3.1.4 Conclusion for Patch size difference

In this example of geometric transformation of the size of the patch, we can see that both models trend to a lower mAP as the patch gets larger. YOLOv5 appears to not be very affected when the patch is increased from 50 to 100 pixels, but there is a clear drop when it goes to 150 and 200 pixels. In contrast, the faster-RCNN model is more sensitive to the patch transformations at 100 and 150 than YOLOv5. Interestingly, the difference in mAP between size 50 and 200 is larger in YOLOv5 (0.7) than faster-RCNN (0.5). This can mean that faster-RCNN is more sensitive to smaller patches than YOLOv5, but is overall more robust to larger patches, this could also be noted by the lower decrease in mAP for faster-RCNN between size 150 and 200 (0.1), vs the drop in the YOLOv5 model (0.4).

3.2 Rotation degree ([90, 180, 270, 360]) difference

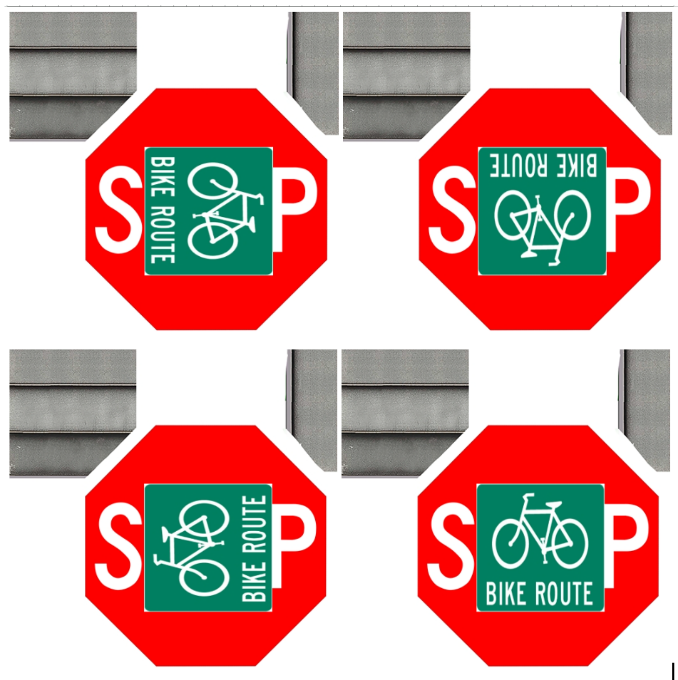

Figure: the textures with different patch rotation degrees, in [90, 180, 270, 360].

3.2.1 The mAP comparison for different rotation degrees on FAST R-CNN

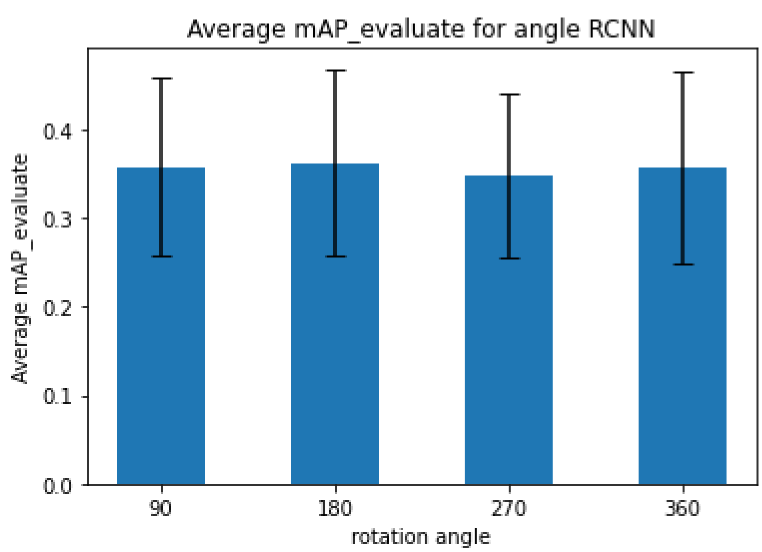

Figure: Bar plot of the Faster-RCNN mAP for each rotation of the patch.

3.2.2 The mAP comparison for different rotation degrees on YOLO v5

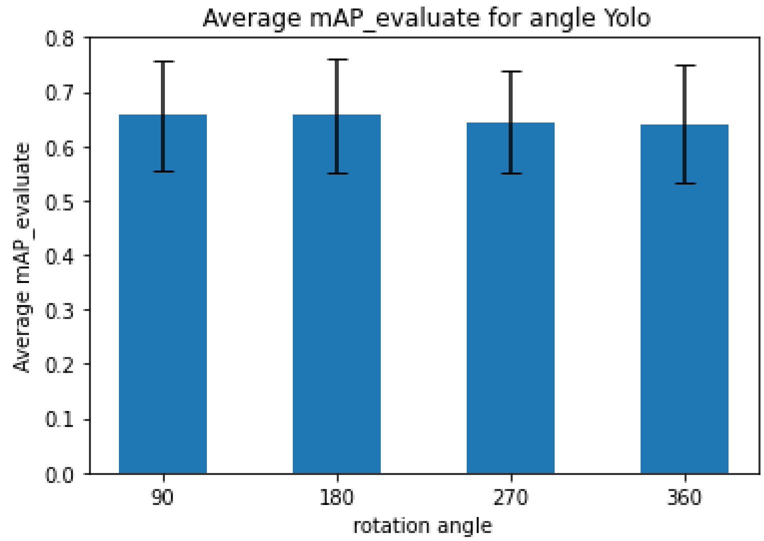

Figure: Bar plot of the YOLOv5 mAP for each rotation of the patch.

3.2.3 Comparison of both object detection models:

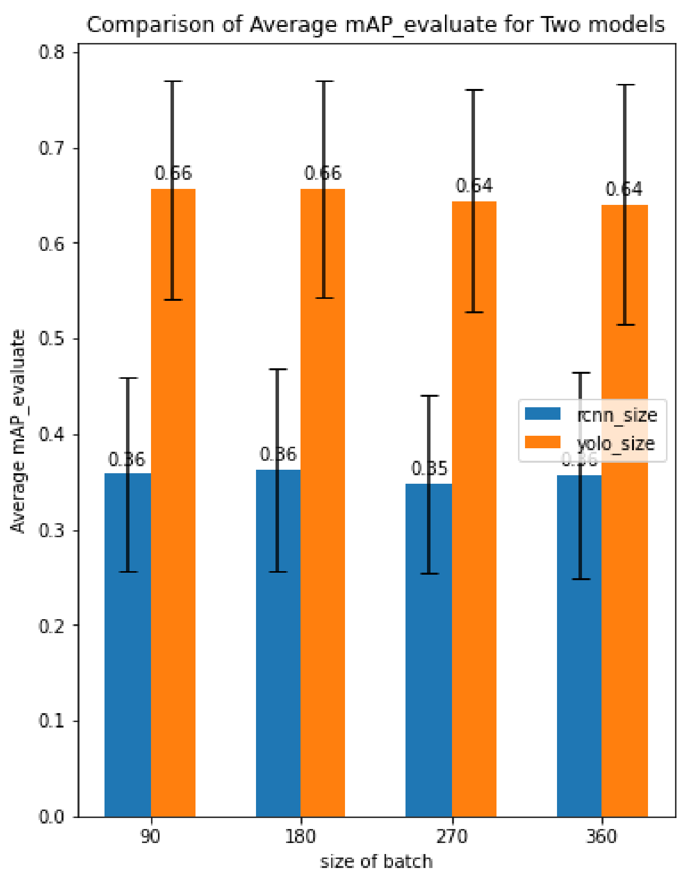

Figure: Bar plot of the mAP of both object detection models for each patch rotation.

3.2.4 Conclusion for Patch Rotation difference

For rotation transformations in intervals of 90 degrees, there doesn't appear to be any effect in the mAP of either model, both seemingly robust to these types of transformations. It may be of interest to perform these transformations in smaller degree intervals, and compare the results.

3.3 Patch Location difference (size of patch : 100*100) & heatmap

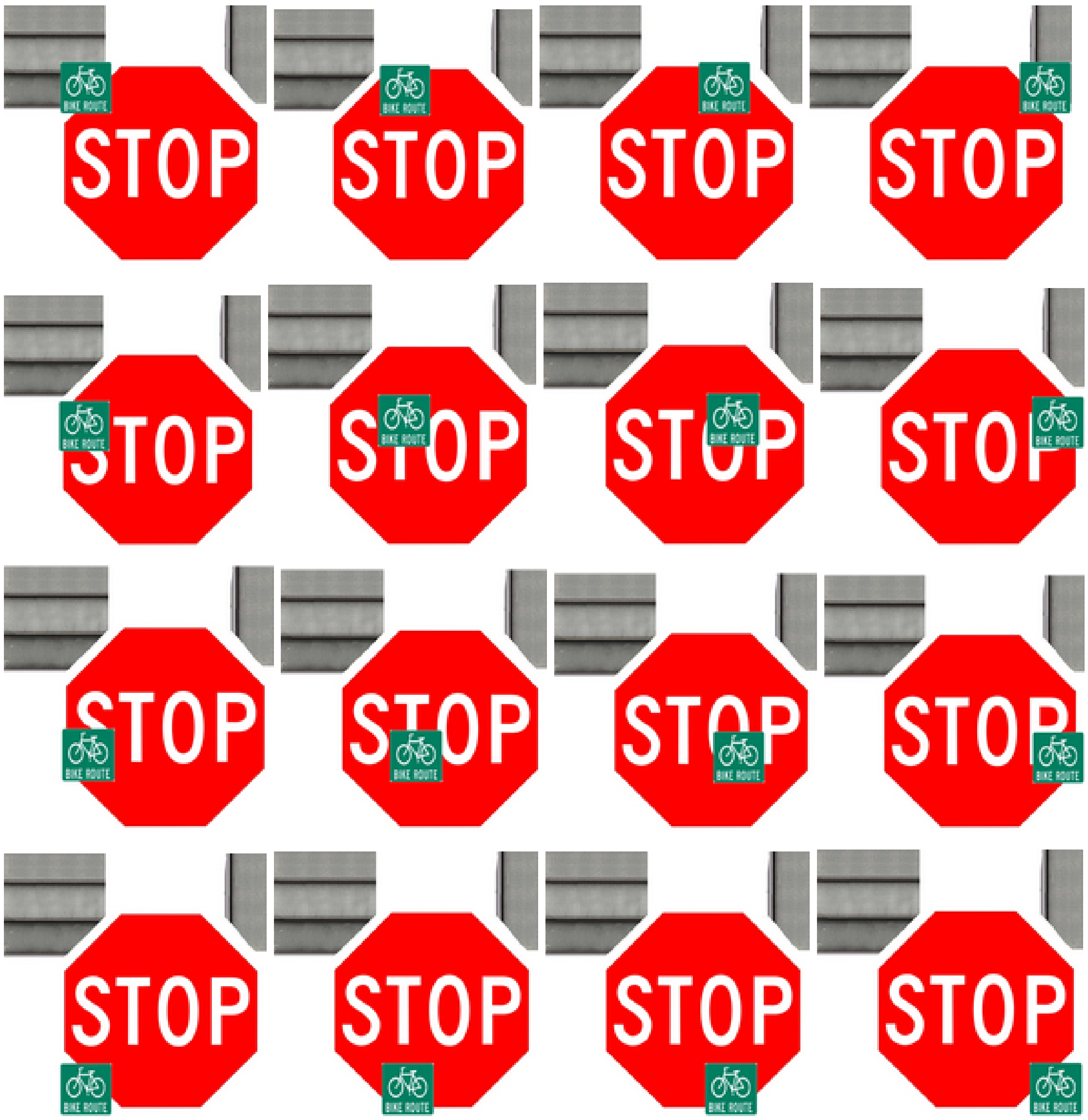

Figure: the textures with different patch locations, patch size 100*100,
around the texture center [310,310] ,from pixel [110,110] to [510,510]

3.3.1 The mAP comparison for different patch location on FAST R-CNN

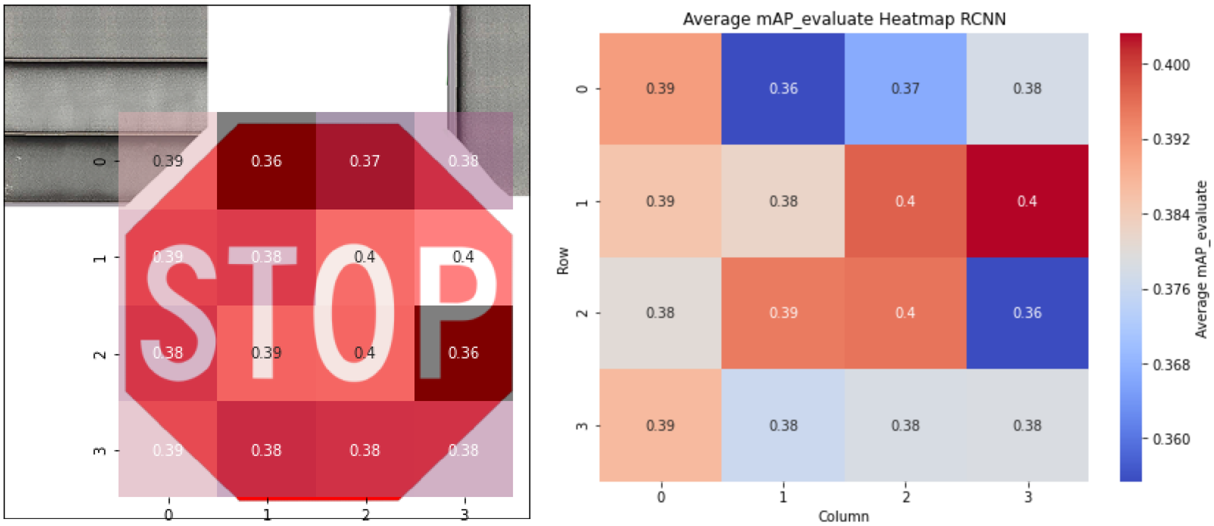

Figure : heatmap of the position's impact under RCNN model (size of patch : 100*100)

3.3.2 The mAP comparison for different patch location on YOLO v5

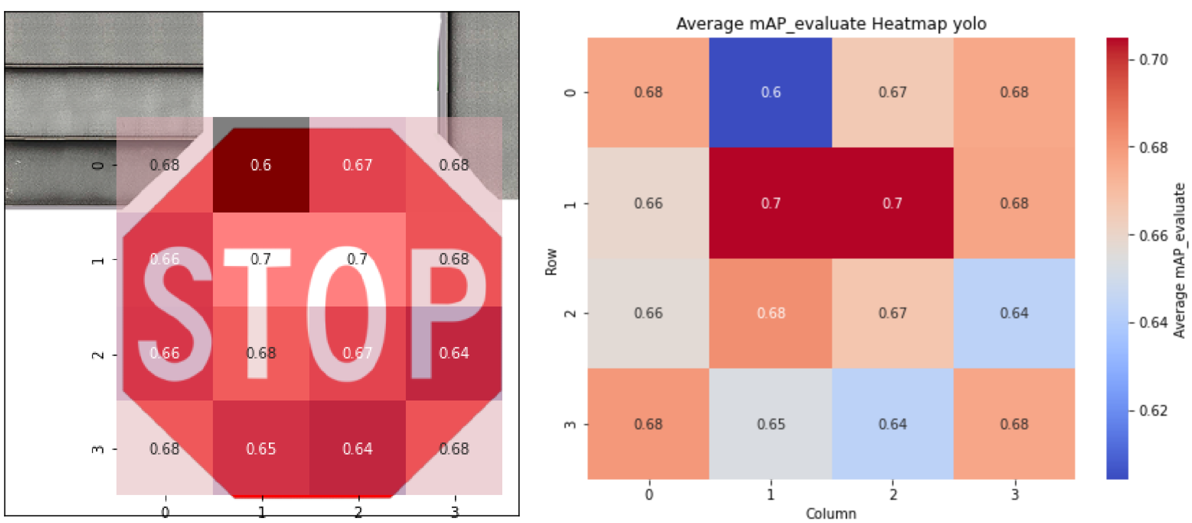

Figure : heatmap of the position's impact in YOLOv5 model (size of patch: 100*100)

3.3.3 Conclusion for Patch Location difference

According to the heatmap of patch location transformations in intervals of 100*100, the average mAP of R-CNN is close to 0.38, while the average mAP of YOLO is close to 0.66. There is some difference in the mAP of both models w.r.t the position of the patch, both seemingly somewhat sensitive to this transformation.

We noticed a lowest mAP on location row0, column1 (pixel coordinate x:210-310 y:110-210), but we may need more patch examples and experiments to tell whether it’s a feature of the customized “bike” patch we use or it’s a weak point for the models. It is also interesting that both models showed high mAP at the central positions, where the text of the sign is largely located, which was unexpected and may require further investigation as to why this happened.