CS771 - Homework 3, FCOS implementation

Ramapriya Ranganath, Ruthvika Reddy Loka, Mahitha Pillodi

Implementation details:

FCOS Classification Head:

This is very straightforward. All we did was pass the features of a particular feature pyramid to the convolutional layers and stack all the outputs.

FCOS Regression Head:

This is also very straightforward. Here, we will have to take care of both the regression output (which has 4 values) and the centerness output (which comes from 0-1). All we did is to pass the features of a particular feature pyramid to the convolutional layers and stack all the outputs into 2 separate lists.

FCOS Compute loss:

$$L(\{\boldsymbol{p}_{x,y}\}, \{\boldsymbol{t}_{x,y}\}) = \frac{1}{N_{\text{pos}}} \sum_{x,y} L_{\text{cls}}(\boldsymbol{p}_{x,y}, c^*_{x,y}) + \frac{\lambda}{N_{\text{pos}}} \sum_{x,y} \mathbb{1}_{\{c^*_{x,y}>0\}} L_{\text{reg}}(\boldsymbol{t}_{x,y}, \boldsymbol{t}^*_{x,y}),$$

This is pretty tricky. It basically has 3 terms, one classification loss (sigmoid focal loss), one regression loss (GIoU loss) and one for centerness loss (BCE loss).

Only when the class is predicted i.e (c*x,y > 0), then only the prediction of regression targets and centerness makes sense. Else, the point will be considered as background. When there are multiple predicted boxes for one ground truth, we take the targets which have a minimum area.

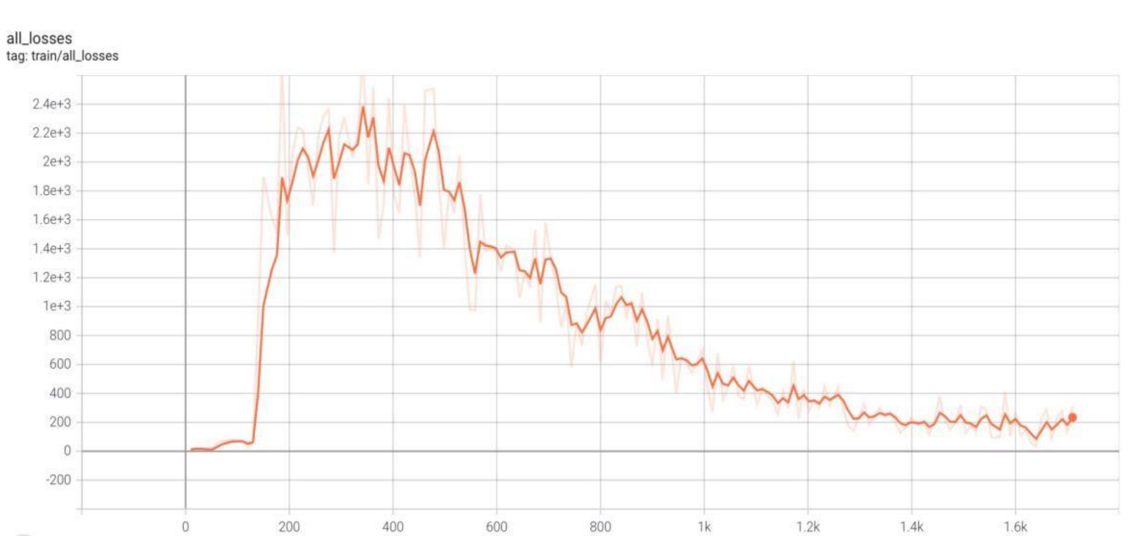
FCOS Inference:

$$l^* = x - x_0^{(i)}, \quad t^* = y - y_0^{(i)},$$
$$r^* = x_1^{(i)} - x, \quad b^* = y_1^{(i)} - y.$$

$$\text{centerness}^* = \sqrt{\frac{\min(l^*, r^*)}{\max(l^*, r^*)} \times \frac{\min(t^*, b^*)}{\max(t^*, b^*)}}.$$

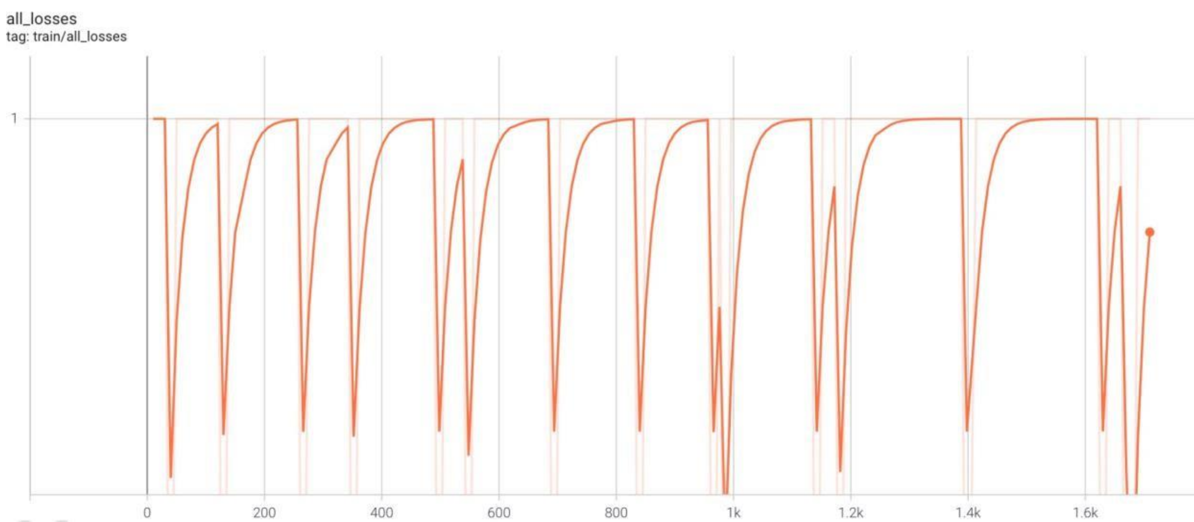
With an input image, we pass it through the network, obtaining classification scores and regression predictions for each location on the feature maps *Fi*. As per [15], we designate locations with classification scores > 0.05 as positive samples. Inverting the equation, we derive the predicted boxes (denoted by * with predicted coordinates, where *x*,*y* are box centers, and (*i*) corresponds to the ground truth labels for a box). During inference, non-max suppression is executed to eliminate boxes with a threshold below the *nms_threshold* initialized in the code.

Results (visualization of plots):

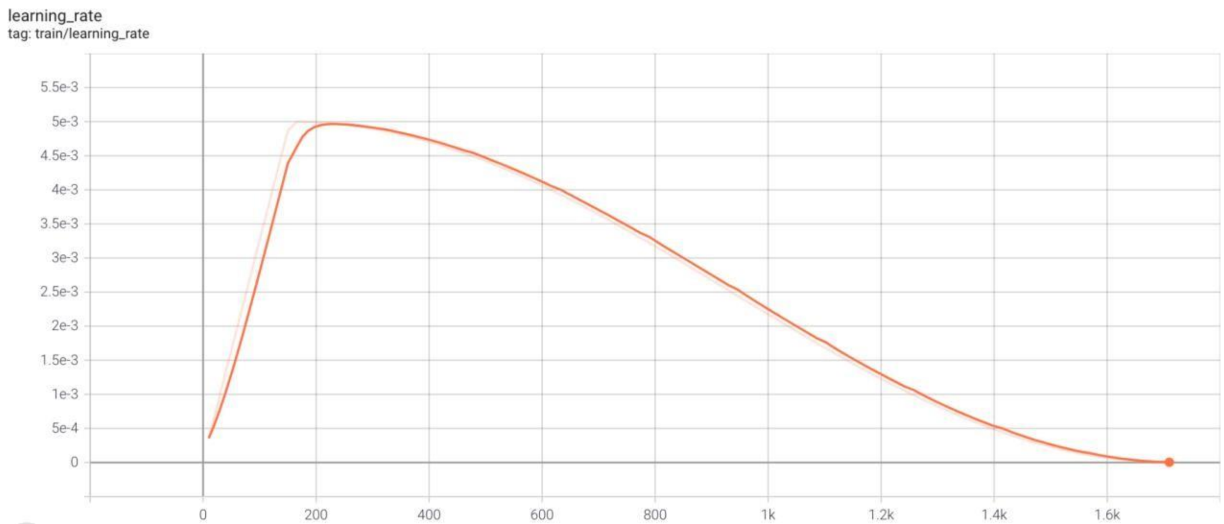


Classification loss

Centerness loss



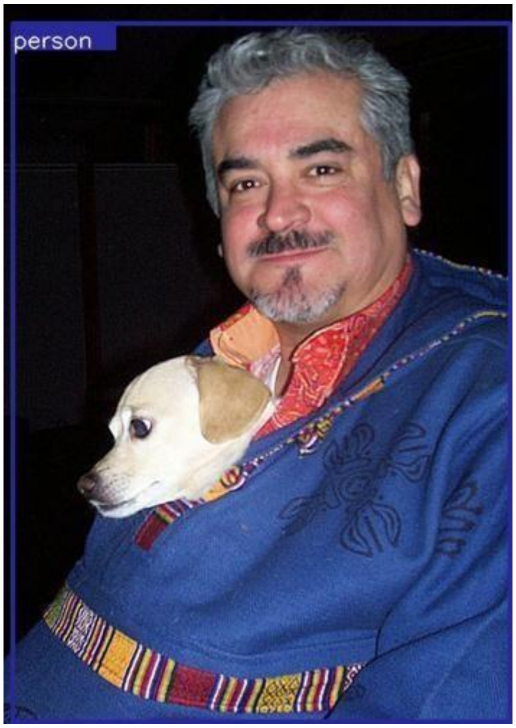
Regression loss



Learning rate (decaying)

Bonus Question :

The collective improvements introduced in the model architecture, data augmentations, hyperparameters, meaningful model changes, and dataset scale led to a substantial enhancement in both accuracy and efficiency. These modifications contribute to a more robust and effective FCOS implementation. The detailed insights from each improvement offer valuable considerations for future work in object detection tasks.


person


person
horse
person


person
horse


train
NREX

Contributions:

Mahitha Pillodi-, FCOS Loss function, Critique writeup

Ramapriya Ranganath- FCOS classification and regression forward, Bonus question, Report

Ruthvika Reddy Loka- Inference function, Bonus question, Report