

PhD

CornerNet Detecting Objects as Paired Keypoints ax1903 ijcv19

2019-12-21 11:38:20 AM

2 class-specific heat maps representing the probability of each pixel being the top left and bottom right corner of a box for each class

2 more feature maps encoding an associative embedding vector and refinement offsets for each pixel

new kind of max pooling called corner pooling to infer the presence of corners from image contents – for each pixel location, all pixels to the right and below that pixel are pooled and then added together for the top left feature map and all pixels above and to the left are pooled for the bottom right feature map

Performance seems to be slightly better than retina net on average, with difference becoming greater when precision is increased
does not outperform 2 stage detectors

hourglass network is used as the backbone which seems to be pretty much the same thing as a fully convolutional auto encoder with skip connections

A sort of Gaussian smoothing is used to spread out the impact of each GT corner by reducing the penalty in the loss function in the neighborhood of the actual corner

Lots of heuristics involved in converting the 4 feature maps into bounding boxes

- 3 x 3 max pooling as a form of NMS
- selecting top 100 boxes in both feature maps
- computing embedding vector distance between (presumably each pair of) corners
- discarding all with distances greater than 0.5 or from different categories
- some sort of soft NMS
- choosing the top one 100 boxes
- mean score of the two feature maps as the corresponding confidence