CenterNet with HarDNet backbone.

CenterNet is an anchor free object detection approach. The network is trained to predict the center of each object, thus treating the center of the object as the only keypoint that needs to be predicted.

After passing the image through a Convolutional network, the network outputs a heat map. The peaks of the heatmap, above a certain threshold are taken as the centers of the objects.

The output of the network is given by three heads.

Heatmap head: Predicts the heatmap, where the value of one signifies the predicted center and zero the background.

The ground truth has an interesting detail, where they smeared the ground truth keypoint pixel with a guassian. This is important as otherwise all incorrect pixels would give the same error value, instead here a gradient is created, leading the prediction from a very bad prediction to a good prediction gently.

This is also aided by the variant focal loss, which weighs answers of high confidence less like a normal focal loss. It also weighs low confidence answers, close to the true center high, thus helping the prediction reach the true center.

Dimension head: predicts the dimensions of the bounding box about the center, by minimizing L1 loss with the real dimensions.

Offset head: This head was a mechanism added to account for the loss caused due to discretization. The offset predicts how much the center has drifted from the real center due to error caused by the scaling down of the real image to the ground truth feature maps.

Total loss is takes as weighted sum of all three losses.

It is significantly different approach from the YOLO algorithm. The YOLO splits the image into cells. Each cell then produces bounding boxes based on anchors. Each box then produces a very specific output of probabilities and positions. Having said this, the YOLO V3 also uses the idea of the center of the object being an important keypoint. It tries to find that cell, which is central to the object. Also, YOLO V4 has made many specific BOF and BOS additions to make it faster.

Something that would be interesting is to use the Mish activation function that YOLOV4 uses. It is supposed to make the loss landscape much smoother than the RELU and hence might lead to better minima and faster training.

The idea to use the HarDNet as a backbone is pretty cool as the centerNet head can be used with any backbone. The HarDNet is a version of Densenet with significantly reduced number of skip connections. This reduced number of computations performed by the DRAM, leads to reduction in inference time.

The strength of center net lies in that it approximates a simpler function first, i.e. “locate the centers of objects” and then uses that information to solve another problem, i.e. “Locate the boundary of the object” making it a smaller network and hence the increase in speed. HarDNet provides the inference time boost from an engineering perspective.