In this paper, we focus on the long-term tracking. The main difference between long-term tracking and short-term tracking is that the former has to deal with the cases in which the target disappears and reappears frequently. We think that a global search mechinaism is necessary [1] for long-term tracking, because in the local search setting, once the target run out of the frame, the track cannot set the window correctly, while the global serach mechinaism can rework when the target go back immidiately.

Directly change the local search tracker to the global search tracker is sub-optimal, because of the small object. As we can see in the object detection field, the smaller, the lower performance. Note that small object is not an issue in local search tracker, because they resize object to a fixed size to avoid this problem.

Now, the question is, how to improve the ability to track the small object in the global search long-term tracking setting. To begin with, we need to analyse that how the size of the object influence the tracking performace and why. As we can see in experiment, the full image tracking performance is strongly relavent to the object size.

Now let us analyse why the full image tracker can not work well on small object. The reson can be catogeried into the following reasons:

- Network depth. The deeper, the bad for small object because of the respective field will involve more background infomation when the network going deeper until the noise feature overwheel the object feature. Similar experiment results can be found in [1] and [2].
- The unfair IoU metric. if we miss one pixel of gt box, the large box degree small while the small box degree big.
- Unfair loss. The large area have more loss. The small area have less loss. Focal loss try to fix this problem.
- Dataset imblance of difference size. If the training set object is all large, how we respect to have good performance on small object?

To handle the above problems, we do the following improve:

- We do deep supervision on shallow and deep layers.
- We not use IoU bet use the point. If the top point fall into the box, it's ok.
- We only use a fixed number of point for different size object.
- We do data augment, have every size.

1 Related Work

1.1 Global Trackers

Siam R-CNN, GlobalTrack,

1.2 Loss function

The focal loss change the weight between the easy and hard example. The hard example mine loss try to increase the hard example weight.

1.3 Handle small objects in object detection

FPN use pyramid features, and detect small objects on

1.4 Object/Tracking as Points

In these methods, they regard the center of object as point as use L1 loss to Fitting a Gaussian distribution. However, we think every point in the box is ok, and do not use l1 to fit the gaussion.

1.5 Sampling strategy

Sampling means how we select examples to train the loss. OHEM try to mine hard examples. [3] think that we need to sample uniformly according to distance.

2 Method

2.1 Loss

If the max point in feature map is fall into the GT box, then we let it go to 1. If the max point in the feature map is fall into the background, then we let it go to 0. Because we let the max in the background to 0, so we do the hard negtive mineing. One drawback is, at the begining of training, maybe the top point always at the background, so we always let max value to 0, so the network learn a trival solution, it is not good. So we need to modify, always selet a positive point and a negative point: Let the max point at the box go to 1, and let the max point at background go to 0. Now, if a positive point is larger than 0.5, we do not need to train it at all. (or a negtive point is less than 0.5). So we can focus on the harder points. Another benifit is it can advoid a trival solution: loss always 1: neg=0 and pos=1

3 Experiments

3.1 Ablation Studies

3.1.1 Imblance training set objecet size

First, we see if the size is imblance in the training set. Second, we see if the imbalce of training data size really influence the test ablity.

3.2 Layer number

3.3 Sample point number

References

[1] Zikai Zhang, Bineng Zhong, Shengping Zhang, Zhenjun Tang, Xin Liu, and Zhaoxiang Zhang, "Distractor-aware fast tracking via dynamic convolutions and mot philosophy," *arXiv preprint arXiv:2104.12041*, 2021.

[2] Zhipeng Zhang and Houwen Peng, "Deeper and wider siamese networks for real-time visual tracking," in *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2019.

[3] Chao-Yuan Wu, R Manmatha, Alexander J Smola, and Philipp Krahenbuhl, "Sampling matters in deep embedding learning," in *Proceedings of the IEEE International Conference on Computer Vision*, 2017, pp. 2840–2848.