16319058 许俊杰 16340082 黄俊凯 16340041 陈亚楠

Generalized Intersection over Union: A Metric and A Loss for Bounding Box Regression

Hamid Rezatofighi, Nathan Tsoi, JunYoung Gwak, Amir Sadeghian, Ian Reid, Silvio Savarese

CVPR 2019

Summary

Highlights

·IoU has weaknesses on optimizing in the case of non-overlapping bounding boxes.

·GIoU is to be a new metric for comparing any two arbitrary shapes.

·Apply GIoU to three popular object detection algorithms and compared.

·An analytical solution for using GIoU as loss between two axis-aligned rectangles or generally n-orthotopes.

The problem to tackle is that as the most popular benchmark in the area of object detection, IoU (Intersection over Union) doesn’t show a strong connection to distance losses in model optimization. This means that there is a gap between optimizing the commonly used distance losses for regressing the parameters of a bounding box and maximizing IoU. The weakness of IoU: (1) When IoU(A, B) = 0, it does not reflect if two shapes are in vicinity of each other or very far from each other; (2) IoU does not reflect how overlap between two objects occurs. IoU value for different alignments of two shapes is identical as long as the volume (area) of their intersection is equal; (3) IoU is invariant in scale while l-norm is sensitive to scale. Therefore, a new metric is needed to act as an optimization target instead of using a proxy loss function. The difficulty to solve this problem is that it’s hard to overcome the shortage of IoU while preserving its advantages. In most cases, IoU can be directly used as a loss function for optimization. However, when there is no overlapping between two bounding boxes, IoU is 0 and its derivate is 0, which is hard to optimize. Thus, the main point of this is to make a more explanatory benchmark which better describes the characteristics of different kinds of phenomena (overlapping and non-overlapping). And it tries to identify different kinds of overlapping (not only in the dimension of area) and non-overlapping (taking distance into account).

In this paper, a generalized version of IoU as a new metric for comparing any two arbitrary shapes, GIoU, is introduced to solve the problem. GIoU is established on the basis of IoU.

Here is the computing process of GIoU: For two arbitrary convex shapes (volumes) A, B ⊆ S ∈Rn, the first step is to find the smallest convex shape(volume) C ⊆ S ∈ Rn enclosing both A and B. Secondly, we can calculate a ratio between the volume (area) occupied by C excluding A and B and divide by the total volume (area) occupied by C. The last step is to subtract this ratio calculated in step 2 from the IoU value.

The novelty of the paper is its focus of optimizing the algorithm switch to creating a new evaluation metric. Instead of optimizing the algorithm on the level of architecture of model network, this paper begin with the loss function utilized in the most widely used algorithm of object detection and find that there is a gap between optimizing the commonly used distance losses for regressing the parameters of a bounding box and maximizing IoU. On the basis of the work mentioned above, this paper finds a metric which can be directly used as an optimization target in evaluation process.

In this paper, there are comparisons between the performance of different algorithms including YOLOv3, Faster R-CNN and Mask R-CNN using different loss functions like MSE, l1-smooth, LIoU and LGIoU put forward in this paper. The datasets of this experiment are MS COCO 2018 and PASCAL VOC 2007.  From the result, it is safe to draw a conclusion that using LGIoU as a loss function shows greater improvement on YOLOv3 than Faster R-CNN and Mask R-CNN. Therefore, we will analyses its performance on YOLOv3 in details in the following part.

From the Table 3 of this paper, we can see the result of comparison between the performance of YOLO v3 trained using its own loss (MSE) as well as using LIoU and LGIoU losses on the test set of MS COCO 2018. It shows that the AP(Average Precision ) value of MSE is 0.311 and that of LIoU is 0.312 whose improvement is 0.32%. The AP value of LGIoU is 0.329 which is 5.47% improved. The AP75(Average Precision with threshold 0.75) value of MSE is 0.330 and that of LIoU is 0.338 whose improvement is 2.37%. The AP75 value of LGIoU is 0.359 which is 8.79% improved. It means that LGIoU enables the model to have better performance on AP and have higher accuracy for higher thresholds.

The advantages of GIoU is that it not only overcomes the shortage of IoU but also preserves its advantages. (1) It successfully handles the problems that IoU does not reflect if two shapes are in vicinity of each other or very far from each other when its value is 0. (2) And it settles the problem that IoU does not reflect how overlap between two objects occurs. IoU value for different alignments of two shapes is identical as long as the volume (area) of their intersection is equal. From our perspective, the disadvantage of GIoU is that when two bounding boxes are symmetric to each other, their GIoUs are identical. However, when bounding box B1 covers the area which contains more information such as faces while bounding box B2 covers the edge of object such as legs. In this situation, their GIoUs are identical as long as they are symmetric to each other. But this is not the desirable result. Because it is obvious that the area covered by B1 is more valuable than that of B2. The second disadvantage is that there is no practical solution for deriving an analytic solution for GIoU in the case of two rotating rectangular cuboids.

As we mentioned above, GIoUs of two different bounding boxes are identical as long as they are symmetric to each other. However, they may cover areas of different values for the problem. In object detection, face may carry more information. Therefore, we hope that a method can be proposed to distinguish these two kinds of situations and assign different values to them accordingly. Secondly, a practical method should be proposed to derive an analytic solution for GIoU in the case of two rotating rectangular cuboids. This extension and incorporating it as a loss could have great potential to improve the performance of 3D object detection frameworks.