Probabilistic Anchor Assignment with IoU Prediction for Object Detection

Kang Kim[1★] and Hee Seok Lee[2]

[1] XL8 Inc.
kai@xl8.ai
[2] Qualcomm Korea YH
heeseokl@qti.qualcomm.com

Abstract. In object detection, determining which anchors to assign as positive or negative samples, known as *anchor assignment*, has been revealed as a core procedure that can significantly affect a model's performance. In this paper we propose a novel anchor assignment strategy that adaptively separates anchors into positive and negative samples for a ground truth bounding box according to the model's learning status such that it is able to reason about the separation in a probabilistic manner. To do so we first calculate the scores of anchors conditioned on the model and fit a probability distribution to these scores. The model is then trained with anchors separated into positive and negative samples according to their probabilities. Moreover, we investigate the gap between the training and testing objectives and propose to predict the Intersection-over-Unions of detected boxes as a measure of localization quality to reduce the discrepancy. The combined score of classification and localization qualities serving as a box selection metric in non-maximum suppression well aligns with the proposed anchor assignment strategy and leads significant performance improvements. The proposed methods only add a single convolutional layer to RetinaNet baseline and does not require multiple anchors per location, so are efficient. Experimental results verify the effectiveness of the proposed methods. Especially, our models set new records for single-stage detectors on MS COCO test-dev dataset with various backbones. Code is available at https://github.com/kkhoot/PAA.

1   Introduction

Object detection in which objects in a given image are classified and localized, is considered as one of the fundamental problems in Computer Vision. Since the seminal work of R-CNN [8], recent advances in object detection have shown rapid improvements with many innovative architectural designs [19,26,38,40], training objectives [3,7,20,27] and post-processing schemes [2,12,13] with strong CNN backbones [5,10,15,17,28,29,33]. For most of CNN-based detectors, a dominant paradigm of representing objects of various sizes and shapes is to enumerate anchor boxes of multiple scales and aspect ratios at every spatial location. In

---

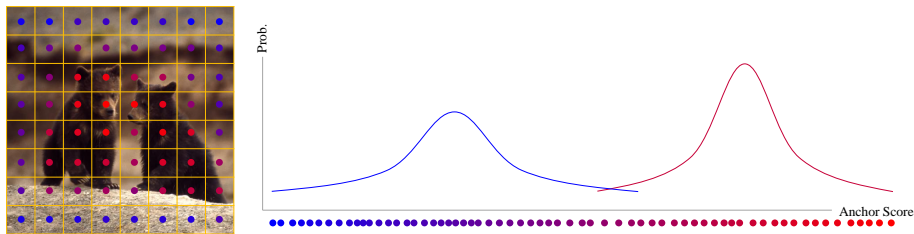★ Work done while at Qualcomm Korea YH.

Fig. 1: An examplary case of anchor scores calculated by a detector model and their distribution. The scores are based on the loss objectives of classification and localization to reflect how each anchor contains meaningful cues identifiable by the model to detect a target object. We model the scores as samples from a probability distribution using Gaussian Mixture Model of two modalities (one for positive and the other for negative samples). Anchors are assigned as positive or negative samples according to their probabilities. Image source: [1]

this paradigm, *anchor assignment* procedure in which anchors are assigned as positive or negative samples needs to be performed. The most common strategy to determine positive samples is to use Intersection-over-Union (IoU) between an anchor and a ground truth (GT) bounding box. For each GT box, one or more anchors are assigned as positive samples if its IoU with the GT box exceeds a certain threshold. Target values for both classification and localization (i.e. regression offsets) of these anchors are determined by the object category and the spatial coordinate of the GT box.

Although the simplicity and intuitiveness of this heuristic make it a popular choice, it has a clear limitation in that it ignores the actual *content* of the intersecting region, which may contain noisy background, nearby objects or few meaningful parts of the target object to be detected. Several recent studies [14, 18, 31, 37, 39] have identified this limitation and suggested various new anchor assignment strategies. These works include selecting positive samples based on the detection-specific likelihood [39], the statistics of anchor IoUs [37] or the cleanness score of anchors [14, 18]. All these methods show improvements compared to the baseline, and verify the importance of anchor assignment in object detection.

In this paper we would like to extend some of these ideas further and propose a novel anchor assignment strategy. In order for an anchor assignment strategy to be effective, a flexible number of anchors should be assigned as positives (or negatives) not only on IoUs between anchors and a GT box but also on how probable it is that a model can reason about the assignment. In this respect, the model needs to take part in the assignment procedure, and positive samples need to vary depending on the model. When no anchor has a high IoU for a GT box, some of the anchors need to be assigned as positive samples to reduce the impact of the improper anchor design. In this case, anchors in which the model finds the most meaningful cues about the target object (that may not

necessarily be anchors of the highest IoU) can be assigned as positives. On the other side, when there are many anchors that the model finds equally of high quality and competitive, all of these anchors need to be treated as positives not to confuse the training process. Most importantly, to satisfy all these conditions, the quality of anchors as a positive sample needs to be evaluated reflecting the *model's current learning status*, i.e. its parameter values.

With this motivation, we propose a probabilistic anchor assignment (PAA) strategy that adaptively separates a set of anchors into positive and negative samples for a GT box according to the learning status of the model associated with it. To do so we first define a score of a detected bounding box that reflects both the classification and localization qualities. We then identify the connection between this score and the training objectives and represent the score as the combination of two loss objectives. Based on this scoring scheme, we calculate the scores of individual anchors that reflect how the model finds useful cues to detect a target object in each anchor. With these anchor scores, we aim to find a probability distribution of two modalities that best represents the scores as positive or negative samples as in Figure 1. Under the found probability distribution, anchors with probabilities from the positive component are high are selected as positive samples. This transforms the anchor assignment problem to a maximum likelihood estimation for a probability distribution where the parameters of the distribution is determined by anchor scores. Based on the assumption that anchor scores calculated by the model are samples drawn from a probability distribution, it is expected that the model can infer the sample separation in a probabilistic way, leading to easier training of the model compared to other non-probabilistic assignments. Moreover, since positive samples are adaptively selected based on the anchor score distribution, it does not require a pre-defined number of positive samples nor an IoU threshold.

On top of that, we identify that in most modern object detectors, there is inconsistency between the testing scheme (selecting boxes according to the classification score only during NMS) and the training scheme (minimizing both classification and localization losses). Ideally, the quality of detected boxes should be measured based not only on classification but also on localization. To improve this incomplete scoring scheme and at the same time to reduce the discrepancy of objectives between the training and testing procedures, we propose to predict the IoU of a detected box as a localization quality, and multiply the classification score by the IoU score as a metric to rank detected boxes. This scoring is intuitive, and allows the box scoring scheme in the testing procedure to share the same ground not only with the objectives used during training, but also with the proposed anchor assignment strategy that brings both classification and localization into account, as depicted in Figure 2. Combined with the proposed PAA, this simple extension significantly contributes to detection performance. We also compare the IoU prediction with the centerness prediction [30, 37] and show the superiority of the proposed method.

With an additional improvement in post-processing named score voting, each of our methods shows clear improvements as revealed in the ablation studies.

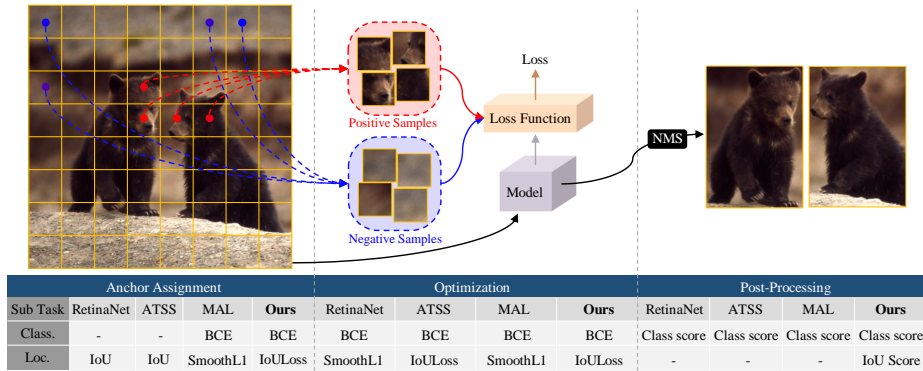| Sub Task	Anchor Assignment				Optimization				Post-Processing			
	RetinaNet	ATSS	MAL	**Ours**	RetinaNet	ATSS	MAL	**Ours**	RetinaNet	ATSS	MAL	**Ours**
Class.	-	-	BCE	BCE	BCE	BCE	BCE	BCE	Class score	Class score	Class score	Class score
Loc.	IoU	IoU	SmoothL1	IoULoss	SmoothL1	IoULoss	SmoothL1	IoULoss	-	-	-	IoU Score

Fig. 2: Illustration of three key procedures of object detectors and comparison between RetinaNet [20], ATSS [37], MAL [14] and ours about in which form classification and localization tasks are concerned in each procedure. Unlike others, we bring both tasks into account for all three procedures. For the localization task, we use IoU-based metrics to align the objectives of each procedure.

In particular, on COCO test-dev set [21] all our models achieve new state-of-the-art performance with significant margins. Our model only requires to add a single convolutional layer, and uses a single anchor per spatial locations similar to [37], resulting in a smaller number of parameters compared to RetinaNet [20]. The proposed anchor assignment can be parallelized using GPUs and does not require extra computes in testing time. All this evidence verifies the efficacy of our proposed methods. The contributions of this paper are summarized as below:

1. We model the anchor assignment as a probabilistic procedure by calculating anchor scores from a detector model and maximizing the likelihood of these scores for a probability distribution. This allows the model to infer the assignment in a probabilistic way and adaptively determines positive samples.

2. To align the objectives of anchor assignment, optimization and post-processing procedures, we propose to predict the IoU of detected boxes and use the unified score of classification and localization as a ranking metric for NMS. On top of that, we propose the score voting method as an additional post-processing using the unified score to further boost the performance.

3. We perform extensive ablation studies and verify the effectiveness of the proposed methods. Our experiments on MS COCO dataset with five backbones set up new AP records for all tested settings.

2   Related Work

2.1   Recent Advances in Object Detection

Since Region-CNN [8] and its improvements [7, 26], the concept of anchors and offset regression between anchors and ground truth (GT) boxes along with ob-

ject category classification has been widely adopted. In many cases, multiple anchors of different scales and aspect ratios are assigned to each spatial location to cover various object sizes and shapes. Anchors that have IoU values greater than a threshold with one of GT boxes are considered as positive samples. Some systems use two-stage detectors [7, 8, 19, 26], which apply the anchor mechanism in a region proposal network (RPN) for class-agnostic object proposals. A second-stage detection head is run on aligned features [9, 26] of each proposal. Some systems use single-stage detectors [20, 22–24, 38, 40], which does not have RPN and directly predict object categories and regression offsets at each spatial location. More recently, anchor-free models that do not rely on anchors to define positive and negative samples and regression offsets have been introduced. These models predict various key points such as corners [16], extreme points [41], center points [6,30] or arbitrary feature points [35] induced from deformable convolution [5]. [42] combines anchor-based detectors with anchor-free detection by adding additional anchor-free regression branches. It has been found in [37] that anchor-based and anchor-free models show similar performance when they use the same anchor assignment strategy.

2.2   Anchor Assignment in Object Detection

The task of selecting which anchors (or locations for anchor-free models) are to be designated as positive or negative samples has recently been identified as a crucial factor that greatly affects a model's performance [34, 37, 39]. In this regard, several methods have been proposed to overcome the limitation of the IoU-based hard anchor assignment. MetaAnchor [34] predicts the parameters of the anchor functions (the last convolutional layers of detection heads) dynamically and takes anchor shapes as an argument, which provides the ability to change anchors in training and testing. Rather than enumerating pre-defined anchors across spatial locations, GuidedAnchoring [31] defines the locations of anchors near the center of GTs as positives and predicts their shapes. FreeAnchor [39] proposes a detection-customized likelihood that considers both the recall and precision of samples into account and determines positive anchors based on the estimated likelihood. ATSS [37] suggests an adaptive anchor assignment that calculates the mean and standard deviation of IoU values from a set of close anchors for each GT. It assigns anchors whose IoU values are higher than the sum of the mean and the standard deviation as positives. Although these works show some improvements, they either require additional layers and complicated structures [31, 34], or force only one anchor to have a full classification score which is not desirable in cases where multiple anchors are of high quality and competitive [39], or rely on IoUs between pre-defined anchors and GTs and consider neither the actual content of the intersecting regions nor the model's learning status [37].

Similar to our work, MultipleAnchorLearning (MAL) [14] and NoisyAnchor [18] define anchor score functions based on classification and localization losses. However, they do not model the anchor selection procedure as a likelihood maximization for a probability distribution; rather, they choose a fixed

number of best scoring anchors. Such a mechanism prevents these models from selecting a flexible number of positive samples according to the model's learning status and input. MAL uses a linear scheduling that reduces the number of positives as training proceeds and requires a heuristic feature perturbation to mitigate it. NoisyAnchor fixes the number of positive samples throughout training. Also, they either miss the relation between the anchor scoring scheme and the box selection objective in NMS [14] or only indirectly relate them using soft-labels [18].

2.3   Predicting Localization Quality in Object Detection

Predicting IoUs as a localization quality of detected bounding boxes is not new. YOLO and YOLOv2 [23, 24] predict "objectness score", which is the IoU of a detected box with its corresponding GT box, and multiply it with the classification score during inference. However, they do not investigate its effectiveness compared to the method that uses classification scores only, and their latest version [25] removes this prediction. IoU-Net [13] also predicts the IoUs of predicted boxes and proposed "IoU-guided NMS" that uses predicted IoUs instead of classification scores as the ranking keyword, and adjusts the selected box's score as the maximum score of overlapping boxes. Although this approach can be effective, they do not correlate the classification score with the IoU as a unified score, nor do they relate the NMS procedure and the anchor assignment process. In contrast to predicting IoUs, some works [4, 11] add an additional head to predict the variance of localization to regularize training [11] or penalize the classification score in testing [4].

3   Proposed Methods

3.1   Probabilistic Anchor Assignment Algorithm

Our goal here is to devise an anchor assignment strategy that takes three key considerations into account: Firstly, it should measure the quality of a given anchor based on how likely the model associated with it finds evidence to identify the target object with that anchor. Secondly, the separation of anchors into positive and negative samples should be adaptive so that it does not require a hyperparameter such as an IoU threshold. Lastly, the assignment strategy should be formulated as a likelihood maximization for a probability distribution in order for the model to be able to reason about the assignment in a probabilistic way. In this respect, we design an anchor scoring scheme and propose an anchor assignment that brings the scoring scheme into account.

Specifically, let us define the score of an anchor that reflects the quality of its bounding box prediction for the closest ground truth (GT) g. One intuitive way is to calculate a classification score (compatibility with the GT class) and a localization score (compatibility with the GT box) and multiply them:

$$S(f_\theta(a, x), g) = S_{cls}(f_\theta(a, x), g) \times S_{loc}(f_\theta(a, x), g)^\lambda \tag{1}$$

where S_{cls}, S_{loc}, and λ are the score of classification and localization of anchor a given g and a scalar to control the relative weight of two scores, respectively. x and f_θ are an input image and a model with parameters θ. Note that this scoring function is dependent on the model parameters θ. We can define and get S_{cls} from the output of the classification head. How to define S_{loc} is less obvious, since the output of the localization head is encoded offset values rather than a score. Here we use the Intersection-over-Union (IoU) of a predicted box with its GT box as S_{loc}, as its range matches that of the classification score and its values naturally correspond to the quality of localization:

$$S_{loc}(f_\theta(a, x), g) = \text{IoU}(f_\theta(a, x), g) \tag{2}$$

Taking the negative logarithm of score function S, we get the following:

$$-\log S(f_\theta(a, x), g) = -\log S_{cls}(f_\theta(a, x), g) - \lambda \log S_{loc}(f_\theta(a, x), g)$$
$$= \mathcal{L}_{cls}(f_\theta(a, x), g) + \lambda \mathcal{L}_{IoU}(f_\theta(a, x), g) \tag{3}$$

where $\mathcal{L}_{cls}$ and $\mathcal{L}_{IoU}$ denote binary cross entropy loss[3] and IoU loss [36] respectively. One can also replace any of the losses with a more advanced objective such as Focal Loss [20] or GIoU Loss [27]. It is then legitimate that the negative sum of the two losses can act as a scoring function of an anchor given a GT box.

To allow a model to be able to reason about whether it should predict an anchor as a positive sample in a probabilistic way, we model anchor scores for a certain GT as samples drawn from a probability distribution and maximize the likelihood of the anchor scores w.r.t the parameters of the distribution. The anchors are then separated into positive and negative samples according to the probability of each being a positive or a negative. Since our goal is to distinguish a set of anchors into two groups (positives and negatives), any probability distribution that can model the multi-modality of samples can be used. Here we choose Gaussian Mixture Model (GMM) of two modalities to model the anchor score distribution.

$$P(a|x, g, \theta) = w_1 \mathcal{N}_1(a; m_1, p_1) + w_2 \mathcal{N}_2(a; m_2, p_2) \tag{4}$$

where w_1, m_1, p_1 and w_2, m_2, p_2 represent the weight, mean and precision of two Gaussians, respectively. Given a set of anchor scores, the likelihood of this GMM can be optimized using Expectation-Maximization (EM) algorithm.

With the parameters of GMM estimated by EM, the probability of each anchor being a positive or a negative sample can be determined. With these probability values, various techniques can be used to separate the anchors into two groups. Figure 3 illustrates different examples of separation boundaries based on anchor probabilities. The proposed algorithm using one of these boundary schemes is described in Procedure 1. To calculate anchor scores, anchors are first allocated to the GT of the highest IoU (Line 3). To make EM efficient, we collect top K anchors from each pyramid level (Line 5-11) and perform EM (Line 12). Non-top K anchors are assigned as negative samples (Line 16).

---

[3] We assume a binary classification task. Extending it to a multi-class case is straightforward.
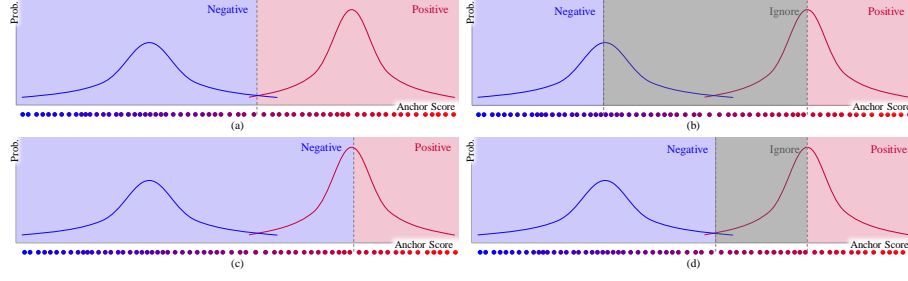
Fig. 3: Different boundary schemes to separate anchors using their probabilities.

Note that the number of positive samples is adaptively determined depending on the estimated probability distribution conditioned on the model's parameters. This is in contrast to previous approaches that ignore the model [37] or heuristically determine the number of samples as a hyperparameter [14, 18] without modeling the anchor assignment as a likelihood maximization for a probability distribution. FreeAnchor [39] defines a detection-customized likelihood and models the product of the recall and the precision as the training objective. But their approach is significantly different than ours in that we do not separately design likelihoods for recall and precision, nor do we restrict the number of anchors that have a full classification score to one. In contrast, our likelihood is based on a simple one-dimensional GMM of two modalities conditioned on the model's parameters, allowing the anchor assignment strategy to be easily identified by the model. This results in easier learning compared to other anchor assignment methods that require complicated sub-routines (e.g. the mean-max function to stabilize training [39] or the anchor depression procedure to avoid local minima [14]) and thus leads to better performance as shown in the experiments.

To summarize our method and plug it into the training process of an object detector, we formulate the final training objective for an input image x (we omit x for brevity):

$$\underset{\theta}{\mathrm{argmax}} \prod_{g} \prod_{a \in \mathcal{A}_g} P_{pos}(a,\theta,g) S_{pos}(a,\theta,g) + P_{neg}(a,\theta,g) S_{neg}(a,\theta) \qquad (5)$$

$$
\begin{aligned}
S_{pos}(a,\theta,g) &= S(f_\theta(a),g) \\
&= \exp(-\mathcal{L}_{cls}(f_\theta(a),g) - \lambda \mathcal{L}_{IoU}(f_\theta(a)),g)
\end{aligned}
\qquad (6)
$$

$$S_{neg}(a,\theta) = \exp(-\mathcal{L}_{cls}(f_\theta(a),\varnothing)) \qquad (7)$$

where $P_{pos}(a,\theta,g)$ and $P_{neg}(a,\theta,g)$ indicate the probability of an anchor being a positive or a negative and can be obtained by the proposed PAA. $\varnothing$ means the background class. Our PAA algorithm can be viewed as a procedure to compute P_{pos} and P_{neg} and approximate them as binary values (i.e. separate anchors into two groups) to ease optimization. In each training iteration, after

---

Procedure 1 Probabilistic anchor assignment algorithm.

Input: $\mathcal{G}$, $\mathcal{A}$, $\mathcal{A}_i$, $\mathcal{L}$, $\mathcal{K}$

    $\mathcal{G}$ is a set of ground-truth boxes

    $\mathcal{A}$ is a set of all anchor boxes

    $\mathcal{A}_i$ is a set of anchor boxes from i_{th} pyramid level

    $\mathcal{L}$ is the number of pyramid levels

    $\mathcal{K}$ is the number of candidate anchors for each pyramid

Output: $\mathcal{P}$, $\mathcal{N}$, $\mathcal{I}$

    $\mathcal{P}$ is a set of positive samples

    $\mathcal{N}$ is a set of negative samples

    $\mathcal{I}$ is a set of ignoring samples

 1: $\mathcal{P} \leftarrow \varnothing, \mathcal{N} \leftarrow \varnothing$

 2: **for** $g \in \mathcal{G}$ **do**

 3:    $\mathcal{A}_g \leftarrow$ GetAnchors($\mathcal{A}$, g, $\mathcal{G}$) {Get all anchors that has g as best GT w.r.t. IoU.}

 4:    $\mathcal{C}_g \leftarrow \varnothing$

 5:    **for** $i = 1$ **to** $\mathcal{L}$ **do**

 6:       $\mathcal{A}_i^g \leftarrow \mathcal{A}_i \cap \mathcal{A}_g$

 7:       $\mathcal{S}_i \leftarrow$ ComputeAnchorScores($\mathcal{A}_i^g$, g) {Negative of Eq.3}

 8:       $t_i \leftarrow$ FindKthLargest($s_i, \mathcal{K}$)

 9:       $\mathcal{C}_g^i \leftarrow \{a_j \in \mathcal{A}_i^g \mid t_i \leq s_j \in \mathcal{S}_i\}$

10:      $\mathcal{C}_g \leftarrow \mathcal{C}_g \cup \mathcal{C}_g^i$

11:    **end for**

12:    $\mathcal{B}, \mathcal{F} \leftarrow$ FitGMM($\mathcal{C}_g$, 2) {$\mathcal{B}, \mathcal{F}$: Probabilties of two Gaussians for $\mathcal{C}_g$}

13:    $\mathcal{N}_g, \mathcal{P}_g \leftarrow$ SeparateAnchors($\mathcal{C}_g, B, F$) {Separate anchors using one of Fig.3.}

14:    $\mathcal{P} \leftarrow \mathcal{P} \cup \mathcal{P}_g, \mathcal{N} \leftarrow \mathcal{N} \cup \mathcal{N}_g, \mathcal{I} \leftarrow \mathcal{I} \cup (\mathcal{C}_g - \mathcal{P}_g - \mathcal{N}_g)$

15: **end for**

16: $\mathcal{N} \leftarrow \mathcal{N} \cup (\mathcal{A} - \mathcal{P} - \mathcal{N} - \mathcal{I})$

17: **return** $\mathcal{P}$, $\mathcal{N}$, $\mathcal{I}$

---

estimating P_{pos} and P_{neg}, the gradients of the loss objectives w.r.t. θ can be calculated and stochastic gradient descent can be performed.

3.2   IoU Prediction as Localization Quality

The anchor scoring function in the proposed anchor assignment is derived from the training objective (i.e. the combined loss of two tasks), so the anchor assignment procedure is well aligned with the loss optimization. However, this is not the case for the testing procedure where the non-maximum suppression (NMS) is performed solely on the classification score. To remedy this, the localization quality can be incorporated into NMS procedure so that the same scoring function (Equation 1) can be used. However, GT information is only available during training, and so IoU between a detected box and its corresponding GT box cannot be computed at test time.

    Here we propose a simple solution to this: we extend our model to predict the IoU of a predicted box with its corresponding GT box. This extension is straightforward as it requires a single convolutional layer as an additional prediction head that outputs a scalar value per anchor. We use Sigmoid activation

on the output to obtain valid IoU values. The training objective then becomes (we omit input x for brevity):

$$\mathcal{L}(f_\theta(a), g) = \mathcal{L}_{cls}(f_\theta(a), g) + \lambda_1 \mathcal{L}_{IoU}(f_\theta(a), g) + \lambda_2 \mathcal{L}_{IoUP}(f_\theta(a), g) \quad (8)$$

where L_{IoUP} is IoU prediction loss defined as binary cross entropy between predicted IoUs and true IoUs. With the predicted IoU, we compute the unified score of the detected box using Equation 1 and use it as a ranking metric for NMS procedure. As shown in the experiments, bringing IoU prediction into NMS significantly improves performance, especially when coupled with the proposed probabilistic anchor assignment. The overall network architecture is exactly the same as the one in FCOS [30] and ATSS [37], which is RetinaNet with modified feature towers and an auxiliary prediction head. Note that this structure uses only a single anchor per spatial location and so has a smaller number of parameters and FLOPs compared to RetinaNet-based models using nine anchors.

3.3   Score Voting

As an additional improvement method here we propose a simple yet effective post-processing scheme. The proposed score voting method works on each box b of remaining boxes after NMS procedure as follows:

$$p_i = e^{-(1-\text{IoU}(b, b_i))^2 / \sigma_t} \quad (9)$$

$$\hat{b} = \frac{\sum_i p_i s_i b_i}{\sum_i p_i s_i} \text{ subject to IoU}(b, b_i) > 0 \quad (10)$$

where $\hat{b}$, s_i and σ_t is the updated box, the score computed by Equation 1 and a hyperparameter to adjust the weights of adjacent boxes b_i respectively. It is noted that this voting algorithm is inspired by "variance voting" described in [11] and p_i is defined in the same way. However, we do not use the variance prediction to calculate the weight of each neighboring box. Instead we use the unified score of classification and localization s_i as a weight along with p_i.

We found that using p_i alone as a box weight leads to a performance improvement, and multiplying it by s_i further boost the performance. In contrast to the variance voting, detectors without the variance prediction are capable of using the score voting by just weighting boxes with p_i. Detectors with IoU prediction head, like ours, can multiply it by s_i for better accuracy. Unlike the classification score only, s_i can act as a reliable weight since it does not assign large weights to boxes that have a high classification score and a poor localization quality.

4   Experiments

In this section we conduct extensive experiments to verify the effectiveness of the proposed methods on MS COCO benchmark [21]. We follow the common practice of using 'trainval35k' as training data (about 118k images) for all experiments. For ablation studies we measure accuracy on 'minival' of 5k images and comparisons with previous methods are done on 'test-dev' of about 20k images. All accuracy numbers are computed using the official COCO evaluation code.

Table 1: Ablation studies on COCO minival set with Res50 backbone. **Left**: Comparison of anchor separation boundaries in Figure 3, fixed numbers of positives (FNP) and fixed positive score ranges (FSR). **Right**: Effects of individual methods.

Anchor Sep.	AP	AP50	AP75
Fig.3. (a)	40.5	58.8	43.4
Fig.3. (b)	40.8	59.1	44.0
Fig.3. (c)	40.6	58.8	43.7
Fig.3. (d)	40.7	59.1	44.0
FNP (5)	39.5	58.0	42.7
FNP (10)	40.1	58.5	43.3
FNP (20)	40.0	58.5	43.1
FSR (> 0.1)	23.8	38.5	25.2
FSR (> 0.2)	19.3	33.2	19.8
FSR (> 0.3)	training failed		

Method	Aux. task	Voting	AP	AP50	AP75
IoU			34.6	53.0	36.7
IoU	IoU pred.		36.0	54.0	38.9
PAA			39.9	59.1	42.8
PAA	Center pred.		39.8	58.3	43.2
PAA	IoU pred.		40.8	59.1	44.0
PAA	IoU pred.	✓	**41.0**	**59.1**	**44.4**
ATSS	Center pred.		39.4	57.4	42.4
ATSS	IoU pred.		39.8	57.9	43.2

4.1 Training Details

We use a COCO training setting which is the same as [37] in the batch size, frozen Batch Normalization, learning rate, etc. The exact setting can be found in the supplementary material. For ablation studies we use Res50 backbone and run 135k iterations of training. For comparisons with previous methods we run 180k iterations with various backbones. Similar to recent works [30, 37], we use GroupNorm [32] in detection feature towers, Focal Loss [20] as the classification loss, GIoU Loss [27] as the localization loss, and add trainable scalars to the regression head. λ_1 is set to 1 to compute anchor scores and 1.3 when calculating Equation 3. λ_2 is set to 0.5 to balance the scales of each loss term. σ_t is set to 0.025 if the score voting is used. Note that we do *not* use "centerness" prediction or "center sampling" [30, 37] in our models. We set $\mathcal{K}$ to 9 although our method is not sensitive to its value similar to [37]. For GMM optimization, we set the minimum and maximum score of the candidate anchors as the mean of two Gaussians and set the precision values to one as an initialization of EM.

4.2 Ablation Studies

Comparison between different anchor separation points Here we compare the anchor separation boundaries depicted in Figure 3. The left table in Table 1 shows that choosing any of the separation scheme works well. This shows the stability of the proposed anchor assignment method as it is not sensitive to small boundary changes. We also compare our method with two simpler methods, namely fixed numbers of positives (FNP) and fixed positive score ranges (FSR). FNP defines a pre-defined number of top-scoring samples as positives while FSR treats all anchors whose scores exceed a certain threshold as positives. As the results in the right of Table 1 show, both methods show worse performance than

Table 2: **Left**: Performance comparison on COCO minival dataset with Res50 backbone. All models were trained with 135K iterations. **Right**: Average errors of IoU prediction on COCO minival set for various backbones.

Method	AP	AP50	AP75	APs	APm	APl		Backbone	IoU Pred. Err.
RetinaNet	36.7	55.8	39.0	19.7	40.1	49.1		Res50	0.093
MAL [14]	38.4	56.8	41.1	-	-	-		Res101	0.092
ATSS [37]	39.4	57.4	42.4	23.0	42.9	51.9		ResNext101	0.09
Ours	**41.0**	**59.1**	**44.4**	**24.2**	**45.2**	**54.2**		Res101-DCN	0.086

PAA. FSR (> 0.3) fails because the model cannot find anchors whose scores are within the range at early iterations. This shows an advantage of PAA that adaptively determines the separation boundaries without hyperparameters that require careful hand-tuning and so are hard to be adaptive per data.

Effects of individual modules In this section we verify the effectiveness of individual modules of the proposed methods. Accuracy numbers for various combinations are in Table 1. Changing anchor assignment from the IoU-based hard assignment to the proposed PAA shows improvements of 5.3% in AP score. Adding IoU prediction head and applying the unified score function in NMS procedure further boosts the performance to 40.8%. To further verify the impact of IoU prediction, we compare it with centerness prediction used in [30, 37]. As can be seen in the results, centerness does not bring improvements to PAA. This is expected as weighting scores of detected boxes according to its centerness can hinder the detection of acentric or slanted objects. This shows that centerness-based scoring does not generalize well and the proposed IoU-based scoring can overcome this limitation. We also verify that IoU prediction is more effective than centerness prediction for ATSS [37] (39.8% vs. 39.4%). Finally, applying the score voting improves the performance to 41.0%, surpassing previous methods with Res50 backbone in Table 2.Left with significant margins.

Accuracy of IoU prediction We calculate the average error of IoU prediction for various backbones in Table 2.Right. All backbones show less than 0.1 errors, showing that IoU prediction is plausible with an additional convolutional head.

Visualization of anchor assignment We visualize positive and negative samples separated by PAA in Figure 4a. As training proceeds, the distinction between positive and negative samples becomes clearer. Note that the positive anchors do not necessarily have larger IoU values with the target bounding box than the negative ones. Also, many negative anchors in the iteration 30k and 50k have high IoU values. Methods with a fixed number of positive samples [14, 18] can assign these anchors as positives, and the model might predict these anchors with high scores during inference. Finally, many positive anchors have
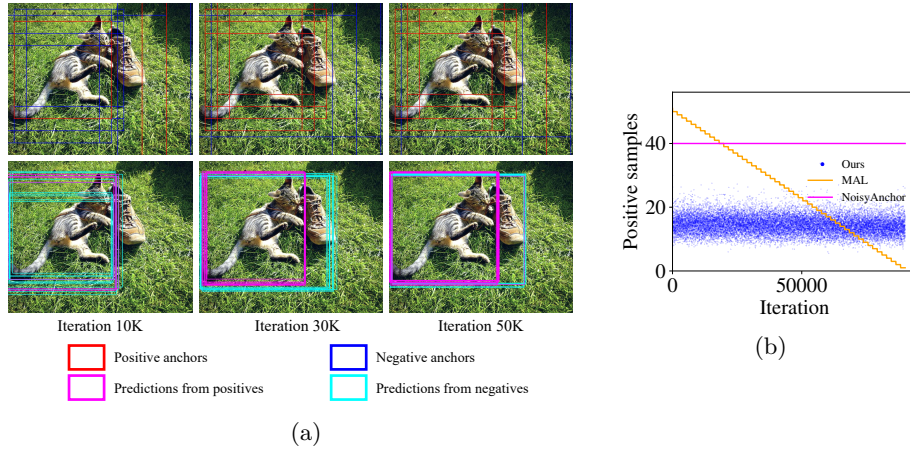
Fig. 4: (a) Evolution of anchor assignment and predicted boxes during training. (b) Plot of the number of positive samples per single GT box throughout training iterations. For our method, the numbers are averaged over a GPU for better visualization (individual values vary between 1 and 40).

more accurate localization as training proceeds. In contrast to ours, methods like FreeAnchor [39] penalize all these anchors except the single best one, which can confuse training.

Statistics of positive samples To compare our method and recent works that also select positive samples by scoring anchors, we plot the number of positive samples according to training iterations in Figure 4b. Unlike methods that either fix the number of samples [18] or use a linear decay [14], ours choose a different number of samples per iteration, showing the adaptability of the method.

4.3 Comparison with State-of-the-art Methods

To verify our methods with previous state-of-the-art ones, we conduct experiments with five backbones as in Table 3. We first compare our models trained with Res10 and previous models trained with the same backbone. Our Res101 model achieves 44.8% accuracy, surpassing previous best models [14,37] of 43.6 %. With ResNext101 our model improves to 46.6% (single-scale testing) and 49.4% (multi-scale testing) which also beats the previous best model of 45.9% and 47.0% [14]. Then we extend our models by applying the deformable convolution to the backbones and the last layer of feature towers same as [37]. These models also outperform the counterparts of ATSS, showing 1.1% and 1.3% improvements. Finally, with the deformable ResNext152 backbone, our models set new records for both the single scale testing (50.8%) and the multi-scale testing (53.5%).

Table 3: Results on COCO test-dev set. * indicates multi-scale testing. Bold text means the best performance among models with the same or a similar backbone.

Method	Backbone	AP	AP50	AP75	APs	APm	APl
RetinaNet [20]	ResNet101	39.1	59.1	42.3	21.8	42.7	50.2
FCOS [30]	ResNet101	41.5	60.7	45.0	24.4	44.8	51.6
NoisyAnchor [18]	ResNet101	41.8	61.1	44.9	23.4	44.9	52.9
FreeAnchor [39]	ResNet101	43.1	62.2	46.4	24.5	46.1	54.8
MAL [14]	ResNet101	43.6	61.8	47.1	25.0	46.9	55.8
ATSS [37]	ResNet101	43.6	62.1	47.4	26.1	47.0	53.6
Ours	ResNet101	**44.8**	**63.3**	**48.7**	**26.5**	**48.8**	**56.3**
FCOS [30]	ResNeXt-64x4d-101	43.2	62.8	46.6	26.5	46.2	53.3
NoisyAnchor [18]	ResNeXt101	44.1	63.8	47.5	26.0	47.4	55.0
FreeAnchor [39]	ResNeXt-64x4d-101	44.9	64.3	48.5	26.8	48.3	55.9
ATSS [37]	ResNeXt-64x4d-101	45.6	64.6	49.7	28.5	48.9	55.6
MAL [14]	ResNeXt101	45.9	65.4	49.7	27.8	49.1	57.8
Ours	ResNeXt-64x4d-101	**46.6**	**65.6**	**50.8**	**28.8**	**50.4**	**57.9**
MAL [14]*	ResNeXt101	47.0	66.1	51.2	30.2	50.1	58.9
Ours*	ResNeXt-64x4d-101	**49.4**	**67.7**	**54.9**	**32.7**	**51.9**	**60.9**
RepPoints [35]	ResNet101-DCN	45.0	**66.1**	49.0	26.6	48.6	57.5
ATSS [37]	ResNet101-DCN	46.3	64.7	50.4	27.7	49.8	58.4
Ours	ResNet101-DCN	**47.4**	65.7	**51.6**	**27.9**	**51.3**	**60.6**
ATSS [37]	ResNeXt-64x4d-101-DCN	47.7	66.5	51.9	29.7	50.8	59.4
Ours	ResNeXt-64x4d-101-DCN	**49.0**	**67.8**	**53.3**	**30.2**	**52.8**	**62.2**
ATSS [37]*	ResNeXt-64x4d-101-DCN	50.7	68.9	56.3	33.2	52.9	62.2
Ours*	ResNeXt-64x4d-101-DCN	**51.4**	**69.7**	**57.0**	**34.0**	**53.8**	**64.0**
Ours	ResNeXt-32x8d-152-DCN	**50.8**	**69.7**	**55.1**	**31.4**	**54.7**	**65.2**
Ours*	ResNeXt-32x8d-152-DCN	**53.5**	**71.6**	**59.1**	**36.0**	**56.3**	**66.9**

5   Conclusions

In this paper we proposed a probabilistic anchor assignment (PAA) algorithm in which the anchor assignment is performed as a likelihood optimization for a probability distribution given anchor scores computed by the model associated with it. The core of PAA is in determining positive and negative samples in favor of the model so that it can infer the separation in a probabilistically reasonable way, leading to easier training compared to the heuristic IoU hard assignment or non-probabilistic assignment strategies. In addition to PAA, we identified the discrepancy of objectives in key procedures of object detection and proposed IoU prediction as a measure of localization quality to apply a unified score of classification and localization to NMS procedure. We also provided the score voting method which is a simple yet effective post-processing scheme that is applicable to most dense object detectors. Experiments showed that the proposed methods significantly boosted the detection performance, and surpassed all previous methods on COCO test-dev set.

References

1. NPS   Photo,   https://www.nps.gov/features/yell/slidefile/mammals/grizzlybear/Images/00110.jpg
2. Bodla, N., Singh, B., Chellappa, R., Davis, L.S.: Soft-nms–improving object detection with one line of code. In: Proceedings of the IEEE international conference on computer vision. pp. 5561–5569 (2017)
3. Chen, K., Li, J., Lin, W., See, J., Wang, J., Duan, L., Chen, Z., He, C., Zou, J.: Towards accurate one-stage object detection with ap-loss. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 5119–5127 (2019)
4. Choi, J., Chun, D., Kim, H., Lee, H.J.: Gaussian yolov3: An accurate and fast object detector using localization uncertainty for autonomous driving. In: Proceedings of the IEEE International Conference on Computer Vision. pp. 502–511 (2019)
5. Dai, J., Qi, H., Xiong, Y., Li, Y., Zhang, G., Hu, H., Wei, Y.: Deformable convolutional networks. In: Proceedings of the IEEE international conference on computer vision. pp. 764–773 (2017)
6. Duan, K., Bai, S., Xie, L., Qi, H., Huang, Q., Tian, Q.: Centernet: Keypoint triplets for object detection. In: Proceedings of the IEEE International Conference on Computer Vision. pp. 6569–6578 (2019)
7. Girshick, R.: Fast r-cnn. In: Proceedings of the IEEE international conference on computer vision. pp. 1440–1448 (2015)
8. Girshick, R., Donahue, J., Darrell, T., Malik, J.: Rich feature hierarchies for accurate object detection and semantic segmentation. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 580–587 (2014)
9. He, K., Gkioxari, G., Dollár, P., Girshick, R.: Mask r-cnn. In: Proceedings of the IEEE international conference on computer vision. pp. 2961–2969 (2017)
10. He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 770–778 (2016)
11. He, Y., Zhu, C., Wang, J., Savvides, M., Zhang, X.: Bounding box regression with uncertainty for accurate object detection. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 2888–2897 (2019)
12. Hosang, J., Benenson, R., Schiele, B.: Learning non-maximum suppression. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 4507–4515 (2017)
13. Jiang, B., Luo, R., Mao, J., Xiao, T., Jiang, Y.: Acquisition of localization confidence for accurate object detection. In: Proceedings of the European Conference on Computer Vision (ECCV). pp. 784–799 (2018)
14. Ke, W., Zhang, T., Huang, Z., Ye, Q., Liu, J., Huang, D.: Multiple anchor learning for visual object detection. arXiv preprint arXiv:1912.02252 (2019)
15. Krizhevsky, A., Sutskever, I., Hinton, G.E.: Imagenet classification with deep convolutional neural networks. In: Advances in neural information processing systems. pp. 1097–1105 (2012)
16. Law, H., Deng, J.: Cornernet: Detecting objects as paired keypoints. In: Proceedings of the European Conference on Computer Vision (ECCV). pp. 734–750 (2018)
17. LeCun, Y., Boser, B., Denker, J.S., Henderson, D., Howard, R.E., Hubbard, W., Jackel, L.D.: Backpropagation applied to handwritten zip code recognition. Neural computation 1(4), 541–551 (1989)

18. Li, H., Wu, Z., Zhu, C., Xiong, C., Socher, R., Davis, L.S.: Learning from noisy anchors for one-stage object detection. arXiv preprint arXiv:1912.05086 (2019)
19. Lin, T.Y., Dollár, P., Girshick, R., He, K., Hariharan, B., Belongie, S.: Feature pyramid networks for object detection. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 2117–2125 (2017)
20. Lin, T.Y., Goyal, P., Girshick, R., He, K., Dollár, P.: Focal loss for dense object detection. In: Proceedings of the IEEE international conference on computer vision. pp. 2980–2988 (2017)
21. Lin, T.Y., Maire, M., Belongie, S., Hays, J., Perona, P., Ramanan, D., Dollár, P., Zitnick, C.L.: Microsoft coco: Common objects in context. In: European conference on computer vision. pp. 740–755. Springer (2014)
22. Liu, W., Anguelov, D., Erhan, D., Szegedy, C., Reed, S., Fu, C.Y., Berg, A.C.: Ssd: Single shot multibox detector. In: European conference on computer vision. pp. 21–37. Springer (2016)
23. Redmon, J., Divvala, S., Girshick, R., Farhadi, A.: You only look once: Unified, real-time object detection. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 779–788 (2016)
24. Redmon, J., Farhadi, A.: Yolo9000: better, faster, stronger. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 7263–7271 (2017)
25. Redmon, J., Farhadi, A.: Yolov3: An incremental improvement. arXiv preprint arXiv:1804.02767 (2018)
26. Ren, S., He, K., Girshick, R., Sun, J.: Faster r-cnn: Towards real-time object detection with region proposal networks. In: Advances in neural information processing systems. pp. 91–99 (2015)
27. Rezatofighi, H., Tsoi, N., Gwak, J., Sadeghian, A., Reid, I., Savarese, S.: Generalized intersection over union: A metric and a loss for bounding box regression. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 658–666 (2019)
28. Simonyan, K., Zisserman, A.: Very deep convolutional networks for large-scale image recognition. arXiv preprint arXiv:1409.1556 (2014)
29. Szegedy, C., Liu, W., Jia, Y., Sermanet, P., Reed, S., Anguelov, D., Erhan, D., Vanhoucke, V., Rabinovich, A.: Going deeper with convolutions. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 1–9 (2015)
30. Tian, Z., Shen, C., Chen, H., He, T.: Fcos: Fully convolutional one-stage object detection. In: Proceedings of the IEEE International Conference on Computer Vision. pp. 9627–9636 (2019)
31. Wang, J., Chen, K., Yang, S., Loy, C.C., Lin, D.: Region proposal by guided anchoring. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 2965–2974 (2019)
32. Wu, Y., He, K.: Group normalization. In: Proceedings of the European Conference on Computer Vision (ECCV). pp. 3–19 (2018)
33. Xie, S., Girshick, R., Dollár, P., Tu, Z., He, K.: Aggregated residual transformations for deep neural networks. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 1492–1500 (2017)
34. Yang, T., Zhang, X., Li, Z., Zhang, W., Sun, J.: Metaanchor: Learning to detect objects with customized anchors. In: Advances in Neural Information Processing Systems. pp. 320–330 (2018)
35. Yang, Z., Liu, S., Hu, H., Wang, L., Lin, S.: Reppoints: Point set representation for object detection. In: Proceedings of the IEEE International Conference on Computer Vision. pp. 9657–9666 (2019)

36. Yu, J., Jiang, Y., Wang, Z., Cao, Z., Huang, T.: Unitbox: An advanced object detection network. In: Proceedings of the 24th ACM international conference on Multimedia. pp. 516–520 (2016)
37. Zhang, S., Chi, C., Yao, Y., Lei, Z., Li, S.Z.: Bridging the gap between anchor-based and anchor-free detection via adaptive training sample selection. arXiv preprint arXiv:1912.02424 (2019)
38. Zhang, S., Wen, L., Bian, X., Lei, Z., Li, S.Z.: Single-shot refinement neural network for object detection. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 4203–4212 (2018)
39. Zhang, X., Wan, F., Liu, C., Ji, R., Ye, Q.: Freeanchor: Learning to match anchors for visual object detection. In: Advances in Neural Information Processing Systems. pp. 147–155 (2019)
40. Zhao, Q., Sheng, T., Wang, Y., Tang, Z., Chen, Y., Cai, L., Ling, H.: M2det: A single-shot object detector based on multi-level feature pyramid network. In: Proceedings of the AAAI Conference on Artificial Intelligence. vol. 33, pp. 9259–9266 (2019)
41. Zhou, X., Zhuo, J., Krahenbuhl, P.: Bottom-up object detection by grouping extreme and center points. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 850–859 (2019)
42. Zhu, C., He, Y., Savvides, M.: Feature selective anchor-free module for single-shot object detection. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 840–849 (2019)