Localize to Classify and Classify to Localize: Mutual Guidance in Object Detection

Heng Zhang, Elisa Fromont, Sébastien Lefèvre, Bruno Avignon

Localize to Classify and Classify to Localize: Mutual Guidance in Object Detection

Heng ZHANG[1,3][0000−0001−6093−1729], Elisa FROMON[1,4][0000−0003−0133−3491], Sébastien LEFEVRE[2][0000−0002−2384−8202], and Bruno AVIGNON[3]

[1] Univ Rennes, IRISA, France
[2] Univ Bretagne Sud, IRISA, France
[3] ATERMES, France
[4] IUF, Inria, France

Abstract. Most deep learning object detectors are based on the anchor mechanism and resort to the Intersection over Union (IoU) between predefined anchor boxes and ground truth boxes to evaluate the matching quality between anchors and objects. In this paper, we question this use of IoU and propose a new anchor matching criterion guided, during the training phase, by the optimization of both the localization and the classification tasks: the predictions related to one task are used to dynamically assign sample anchors and improve the model on the other task, and vice versa. Despite the simplicity of the proposed method, our experiments with different state-of-the-art deep learning architectures on PASCAL VOC and MS COCO datasets demonstrate the effectiveness and generality of our Mutual Guidance strategy.

1 Introduction

Object detection is a popular task in computer vision that aims at localizing objects through bounding boxes and assigning each of them to a predefined class. Deep learning-based methods largely dominate this research field and most recent methods are based on the anchor mechanism [1–12]. Anchors are predefined reference boxes of different sizes and aspect ratios uniformly stacked over the whole image. They help the network to handle object scale and shape variations by converting the object detection problem into an anchor-wise bounding box regression and classification problem. Most state-of-the-art anchor-based object detectors resort to the Intersection over Union (IoU) between the predefined anchor boxes and the ground truth boxes (called IoU_{anchor} in the following) to assign the sample anchors to an object (positive anchors) or a background (negative anchors) category. These assigned anchors are then used to minimize the bounding box regression and classification losses during training.

This IoU_{anchor}-based anchor matching criterion is reasonable under the assumption that anchor boxes with high IoU_{anchor} are appropriate for localization and classification. However, in reality, the IoU_{anchor} is insensitive to objects' content/context, thus not "optimal" to be used, as such, for anchor matching. In Figure 1, we show several examples where IoU_{anchor} does not well reflect the

t

Fig. 1. Anchors A and anchors B have the same IoU with ground truth box but different visual semantic information. The ground truth in each image is marked as dotted-line box. Better viewed in colour.

matching quality between anchors and objects: anchors A and anchors B have exactly the same IoU_{anchor} but possess very different matching qualities. For example, on the first line of Figure 1, anchors A covers a more representative and informative part of the object than anchors B; On the second line, anchors B contains parts of a nearby object which hinders the prediction on the jockey/left person.

Deep learning-based object detection involves two sub-tasks: instance localization and classification. Predictions for these two tasks tell us "where" and "what" objects are on the image respectively. During the training phase, both tasks are jointly optimized by gradient descent, but the static anchor matching strategy does not explicitly benefit from the joint resolution of the two tasks, which may then yield to a task-misalignment problem, i.e., during the evaluation phase, the model might generate predictions with correct classification but imprecisely localized bounding boxes as well as predictions with precise localization but wrong classification. Both predictions significantly reduce the overall detection quality.

To address these two limitations of the existing IoU_{anchor}-based strategy, we propose a new, adaptive anchor matching criterion guided by the localization and by the classification tasks mutually, i.e., resorting to the bounding box regression prediction, we dynamically assign training anchor samples for optimizing classification and vice versa. In particular, we constrain anchors that are well-localized to also be well-classified (*Localize to Classify*), and those well-classified to also be well-localized (*Classify to Localize*). These strategies lead to a content/context-sensitive anchor matching and avoid the task-misalignment problem. Despite the simplicity of the proposed strategy, *Mutual Guidance* brings consistent Average Precision (AP) gains over the traditional static strategy with different deep learning architectures on PASCAL VOC [13] and MS COCO [14] datasets, especially

on strict metrics such as AP75. Our method is expected to be more efficient on applications that require a precise instance localization, e.g., autonomous driving, robotics, outdoor video surveillance, etc.

The main contributions of our work can be summarized as follows:

- We suggest a new perspective for anchor matching in object detection, i.e., assign anchor labels for one task according to the predictions on the other task and vice versa.
- We visualize the difference of anchor matching and object detection results between static methods and *Mutual Guidance*, to explore the reasons of performance improvements.

The rest of this paper is organized as follows: in Section 2, we discuss some representative related work in object detection. Section 3 provides implementation details of the proposed *Mutual Guidance*. Section 4 compares our dynamic anchor matching criterion to the traditional static criterion with different deep learning architectures on different public object detection datasets, and discusses reasons for the precision improvements. Section 5 brings concluding remarks.

2 Related work

Modern CNN-based object detection methods can be divided into two major categories: two-stage detectors and single-stage ones. Both categories give similar performance with a small edge in accuracy for the former and in efficiency for the latter. Besides, both categories of detectors are massively based on the anchor mechanism which usually resorts to IoU_{anchor} for evaluating the matching quality between anchors and objects when assigning training labels and computing the bounding box regression and classification losses for a training example. Our method aims to improve this anchor matching criterion.

2.1 Anchor-based object detection

Two-stage object detectors. Faster RCNN [1] defines the generic paradigm for two-stage object detectors: it first generates a sparse set of Regions of Interest (RoIs) with a Region Proposal Network (RPN), then classifies these regions and refines their bounding boxes. The RoIs are generated by the anchor mechanism. Multiple improvements have been proposed based on this framework: R-FCN [2] suggests position-sensitive score maps to share almost all computations on the entire image; FPN [3] uses a top-down architecture and lateral connections to build high-level semantic feature maps at all scales; PANet [4] enhances the multi-scale feature fusion by adding bottom-up path augmentation to introduce accurate localization signals in lower layers; Libra RCNN [5] proposes the Balanced Feature Pyramid to further integrate multi-scale information into FPN; TridentNet [15] constructs a parallel multi-branch architecture and adopts a scale-aware training scheme for training object scale specialized detection branches. Cascade RCNN [6] further extends the two-stage paradigm into a multi-stage paradigm, where a sequence of detectors are trained stage by stage.

Single-stage object detectors. SSD [7] and YOLO [16] are the fundamental methods for single-stage object detection. From this basis, many other works have been proposed: FSSD [10] aggregates contextual information into the detector by concatenating features of different scales; RetinaNet [9] proposes the Focal Loss to tackle the imbalanced classification problem that arises when trying to separate the actual object to detect from the massive background; RFBNet [11] proposes Receptive Field Block, which takes the relationship between the size and the eccentricity of the reception fields into account; RefineDet [17] introduces an additional stage of refinement for anchor boxes; M2Det [12] stacks multiple thinned U-shape modules to tackle the so-called appearance-complexity variations. While these methods introduce novel architectures to improve results for the object detection task, they all rely on the standard IoU_{anchor}-based matching. We identify this component as a possible limitation and propose a novel matching criterion, that could be adapted to any existing deep architecture for object detection.

2.2 Anchor-free object detection

The idea of anchor-free object detection consists in detecting objects not from predefined anchors boxes, but directly from particular key-points [18–21] or object centres [22–25]. However, these methods do not lead to a substantial accuracy advantage compared to anchor-based methods. The main idea of our *Mutual Guidance* could also be applied to this class of object detectors, and the experimental results with anchor-free detectors are included in the supplementary material.

3 Approach

As already sketched in the introduction, in order to train an anchor-based object detector, the predefined anchors should be assigned as *positive* ("it is a true object") or *negative* ("it is a part of the background") according to an evaluation of the matching between the anchors and the ground truth objects. Then, the bounding box regression loss is optimized according to the positive anchors, and the instance classification loss is optimized according to the positive as well as the negative anchors. When training an anchor-based single-stage object detector with a static anchor matching strategy, the IoU between predefined anchor boxes and ground truth boxes (IoU_{anchor}) is the usual matching criterion. As shown in the IoU_{anchor} column of Figure 2, anchors with more than 50% of IoU_{anchor} are labelled as "positive", those with less than 40% of IoU_{anchor} are labelled as "negative", the rest are "ignored anchors". Note that at least one anchor should be assigned as positive, hence if there is no anchor with more than 50% of IoU_{anchor}, the anchor with the highest IoU_{anchor} is considered.

The proposed *Mutual Guidance* consists of two components: *Localize to Classify* and *Classify to Localize*.

Fig. 2. Illustration of different anchor matching strategies for the boat image resorting to IoU_{anchor} (static), $IoU_{regressed}$ (*Localize to Classify*) and $IoU_{amplified}$ (*Classify to Localize*). Anchors A-M are predefined anchor boxes around the boat on the left side image, for simplicity, we only mark the position of the anchor F and the anchor H. Better viewed in colour.

3.1 Localize to Classify

If an anchor is capable to precisely localize an object, this anchor must cover a good part of the semantically important area of this object and thus could be considered as an appropriate positive sample for classification. Drawing on this, we propose to leverage the IoU between regressed bounding boxes (i.e., the network's localization predictions) and ground truth boxes (noted $IoU_{regressed}$) to better assign the anchor labels for classification. Inspired by the usual IoU_{anchor}, we compare $IoU_{regressed}$ to some given thresholds (discussed in the next paragraph) and then define anchors with $IoU_{regressed}$ greater than a high threshold as positive samples, and those with $IoU_{regressed}$ lower than a low threshold as negative samples (see $IoU_{regressed}$ column of Figure 2).

We now discuss a dynamic solution to set the thresholds. A fixed threshold (e.g., 50% or 40%) does not seem optimal since the network's localization ability gradually improves during the training procedure and so does the $IoU_{regressed}$ for each anchor, leading to the assignment of more and more positive anchors which destabilizes the training. To address this issue, we propose a dynamic thresholding strategy. Even though the IoU_{anchor} is not the best choice to accurately indicate the matching quality between anchors and objects, the number of assigned *positive* and *ignored* anchors does reflect the global matching conditions (brought by the size and the aspect ratio of the objects to detect), thus these numbers could be considered as reference values for our dynamic criterion. As illustrated in Figure 2, while applying the IoU_{anchor}-based anchor matching strategy with the thresholds being 50% and 40%, the number of positive anchors (N_p) and ignored anchors (N_i) are noted ($N_p = 6$ and $N_i = 3$ for the boat). We then use these numbers to label the N_p highest $IoU_{regressed}$ anchors as positive, and the following N_i anchors as ignored. More formally, we exploit the N_p-th

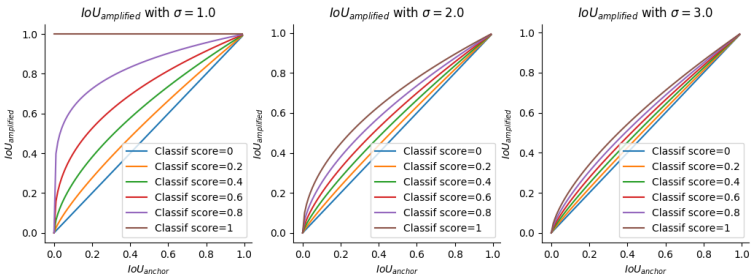

Fig. 3. Illustration of $IoU_{amplified}$ with different σ values (1, 2 or 3). $IoU_{amplified} = IoU_{anchor}$ when $Classif\ score = 0$.

largest $IoU_{regressed}$ as our positive anchor threshold, and the (N_p+N_i)-th largest $IoU_{regressed}$ as our ignored anchor threshold. Using this, our *Localize to Classify* anchor matching strategy evolves with the network's localization capacity and maintains a consistent number of anchor samples assigned to both categories (positive/negative) during the whole training procedure.

3.2 Classify to localize

As with the *Localize to Classify* process, the positive anchor samples in *Classify to Localize* are assigned according to the network's classification predictions (noted $Classif\ score$). Specifically, $Classif\ score$ is the predicted classification score for the object category, e.g., the $Classif\ score$ of Figure 2 indicates the classification score for the *boat* category.

Nevertheless, this $Classif\ score$ is not effective enough to be used directly for assigning good positive anchors for the bounding box regression optimization. It is especially true at the beginning of the training process, when the network's weights are almost random values and all predicted classification scores are close to zero. The $IoU_{regressed}$ is optimized on the basis of the IoU_{anchor}, therefore we have $IoU_{regressed} \geq IoU_{anchor}$ in most cases (even at the beginning of the training), and this property helps to avoid such cold start problem and ensures training stability. Symmetrically to the *Localize to Classify* strategy, we now propose a *Classify to Localize* strategy based on an $IoU_{amplified}$ defined as:

$$IoU_{amplified} = (IoU_{anchor})^{\frac{\sigma - p}{\sigma}} \tag{1}$$

where σ is a hyper-parameter aiming at adjusting the degree of amplification, p represents the mentioned $Classif\ score$. Its behavior is shown in Figure 3. The $IoU_{amplified}$ is always higher than the IoU_{anchor}, and the amplification is proportional to the predicted $Classif\ score$. In particular, the amplification is stronger for smaller σ (note that σ should be larger than 1), and disappears when σ becomes large.

Similarly to the *Localize to Classify* strategy, we apply a dynamic thresholding strategy to keep the number of assigned positive samples for the localization task and for the classification task consistent, e.g., we assign in Figure 2, the top 6 anchors with the highest $IoU_{amplified}$ as positive samples. Note that there is no need for selecting ignored or negative anchors for the localization task since the background does not have an associated ground truth box.

As discussed in Section 1, IoU_{anchor} is not sensitive to the content or the context of an object. Our proposed *Localize to Classify* and *Classify to Localize*, however, attempt to adaptively label the anchor samples according to their visual content and context information. Considering anchor F and anchor H in Figure 2, one can tell that anchor H is better than anchor F for recognizing this boat, even with a smaller IoU_{anchor}. Using both our strategies, anchor H has been promoted to positive thanks to its excellent prediction quality on both tasks whereas anchor F has been labelled as negative even though it has a large IoU_{anchor}.

3.3 About the task-misalignment problem

Since *Localize to Classify* and *Classify to Localize* are independent strategies, they could possibly assign contradictory labels (e.g, the anchor C in Figure 2 is labelled negative for the classification task but positive for the bounding box regression task). This happens when one anchor entails a good prediction on one task and a poor prediction on the other (i.e. they are misaligned predictions). Dealing with such contradictory labels, as we do with *Mutual Guidance*, does not destabilize the training process. On the contrary, it tackles the task-misalignment problem since the labels for one task is assigned according to the prediction quality on the other task, and vice versa. This mechanism enforces the network to generate aligned predictions.

For example, if one anchor is good at classification but poor at localization, for the localization task, the *Mutual Guidance* will assign this anchor positive to encourage this anchor to be good at localization as well; in the same time, for the classification task, considering its localization prediction is poor, in order to avoid misaligned predictions, the *Mutual Guidance* will assign this anchor negative to encourage this anchor to be classified as "background". In fact, the predicted classification score of this mislocalized anchor should be low enough to be suppressed by the NMS procedure of evaluation phase. If one anchor is good at localization but poor at classification, its classification label will be assigned as positive to encourage good predictions on both tasks, however, considering its classification prediction is poor (i.e., this anchor contains one part of the object but it is classified as "background"), its localization has no need to be optimized since this detection will not survive in the NMS procedure of evaluation phase.

If the assigned labels are always the same for both tasks (as done in IoU_{anchor}-based method), it cannot be guaranteed that one anchor's prediction on both tasks (localization and classification) are good at the same time. The NMS will only keep detection results with the highest classification score, so we may end up with keeping predictions with good classification and poor localization for evaluation, which is harmful for strict metrics such as AP75.

4 Experiments

4.1 Experimental Setting

Network architecture and parameters. In order to test the generalization performance of the proposed method, we implement our method on the single-stage object detectors FSSD [10], RetinaNet [9] and RFBNet [11] using both ResNet-18 [26] or VGG-16 [27] as backbone networks in our experiments. Note that RFBNet is not implemented with ResNet-18 as backbone since the two architectures are not compatible. The backbone networks are pre-trained on ImageNet-1k classification dataset [28]. We adopt the Focal Loss [9] and Balanced L1 Loss [5] as our instance classification and bounding box regression loss functions respectively for all experiments. The input image resolution is fixed to 320×320 pixels for all experiments (single scale training and evaluation). Unless specified, all other implementation details are the same as in [11]. Following the results of Figure 3, we decided to fix our only new hyper-parameter σ to 2 for all experiments. σ is used to set the degree of amplification when computing $IoU_{amplified}$ in Eq. (1). It needs to be greater than 1 for the exponent to be positive and lower than 3 since this does not bring any amplification as shown in Figure 3.

Datasets and evaluation metrics. Extensive experiments are performed on two benchmark datasets: PASCAL VOC [13] and MS COCO [14]. PASCAL VOC dataset has 20 object categories. Similarly to previous works, we utilize the combination of VOC2007 and VOC2012 trainval sets for training, and rely on the VOC2007 test for evaluation. MS COCO dataset contains 80 classes. Our experiments on this dataset are conducted on the train2017 and val2017 set for training and evaluation respectively. For all datasets, we use the evaluation metrics introduced in the MS COCO benchmark: the Average Precision (AP) averaged over 10 IoU thresholds from 0.5 to 0.95, but also AP50, AP75, AP_s, AP_m, AP_l. AP50 and AP75 measure the average precision for a given IoU threshold (50% and 75%, respectively). The last three aim at focusing on small ($area < 32^2$), medium ($32^2 < area < 96^2$) and large ($area > 96^2$) objects respectively. Since the size of the objects greatly varies between MS COCO and PASCAL VOC, these size-dependent measures are ignored when experimenting with PASCAL VOC dataset.

4.2 Results

Experiments on PASCAL VOC. We evaluate the effectiveness of both components (*Localize to Classify* and *Classify to Localize*) of our proposed approach w.r.t. the usual IoU_{anchor}-based matching strategy when applied on the same deep learning architectures. The results obtained on the PASCAL VOC dataset are given in Table 1. Both proposed anchor matching strategies consistently boost the performance of the "vanilla" networks and their combination (*Mutual Guidance*) leads to the best AP and all other evaluation metrics.

Model	Matching strategy	*AP*	*AP*50	*AP*75
FSSD with ResNet-18 backbone	IoU_{anchor}-based	50.3%	75.5%	53.7%
	Localize to Classify	51.8%	76.1%	55.9%
	Classify to Localize	51.0%	76.1%	54.3%
	Mutual Guidance	**52.1%**	**76.2%**	**55.9%**
FSSD with VGG-16 backbone	IoU_{anchor}-based	54.1%	80.1%	58.3%
	Localize to Classify	56.0%	80.3%	60.6%
	Classify to Localize	54.4%	79.9%	58.5%
	Mutual Guidance	**56.2%**	**80.4%**	**61.4%**
RetinaNet with ResNet-18 backbone	IoU_{anchor}-based	51.1%	75.8%	54.8%
	Localize to Classify	53.4%	76.5%	57.2%
	Classify to Localize	51.9%	75.9%	55.8%
	Mutual Guidance	**53.5%**	**76.9%**	**57.4%**
RetinaNet with VGG-16 backbone	IoU_{anchor}-based	55.2%	80.2%	59.6%
	Localize to Classify	57.4%	81.1%	62.6%
	Classify to Localize	56.2%	80.1%	61.7%
	Mutual Guidance	**57.7%**	**81.1%**	**62.9%**
RFBNet with VGG-16 backbone	IoU_{anchor}-based	55.6%	80.9%	59.6%
	Localize to Classify	57.2%	80.9%	61.6%
	Classify to Localize	55.9%	80.8%	60.2%
	Mutual Guidance	**57.9%**	**81.5%**	**62.6%**

Table 1. Comparison of different anchor matching strategies (the usual IoU_{anchor}-based, proposed *Localize to Classify*, *Classify to Localize* and *Mutual Guidance*) for object detection. Experiments are conducted on the PASCAL VOC dataset. The best score for each architecture is in bold.

In particular, we observe that the improvements are small on AP50 (around 0.5%) but significant on AP75 (around 3%), which means that we obtain *more precise detections*. As analysed in Section 3.3, this comes from the task-misalignment problem faced with the usual static anchor matching methods. This issue leads to retain well-classified but poorly-localized predictions and suppress well-localized but poorly-classified predictions, which in turns results in a significant drop of the AP score at strict IoU thresholds, e.g., AP75. In *Mutual Guidance*, however, training labels for one task are dynamically assigned according to the prediction quality on the other task and vice versa. This connection makes the classification and localization tasks consistent along all training phases and as such avoids this task-misalignment problem.

We also notice that *Localize to Classify* alone brings, for all five architectures, a higher improvement than *Classify to Localize* alone. We hypothesize two possible reasons for this: 1) most object detection errors come from wrong classification instead of imprecise localization, so the classification task is more difficult than the localization task and thus, there is more room for the improvement on this task; 2) the amplification proposed in Eq. (1) may not be the most appropriate one to take advantage of the classification task for optimizing the bounding box regression task.

Model	Matching strategy	AP	$AP50$	$AP75$	AP_s	AP_m	AP_l
FSSD with	IoU_{anchor}-based	26.1%	42.8%	26.7%	8.6%	29.1%	41.0%
ResNet-18 backbone	*Mutual Guidance*	**27.0%**	**42.9%**	**28.2%**	**9.5%**	**29.7%**	**43.0%**
FSSD with	IoU_{anchor}-based	31.1%	48.9%	32.7%	13.3%	37.2%	44.7%
VGG-16 backbone	*Mutual Guidance*	**32.0%**	**49.3%**	**33.9%**	**13.7%**	**37.8%**	**46.4%**
RetinaNet with	IoU_{anchor}-based	27.8%	44.5%	28.6%	10.4%	31.6%	42.6%
ResNet-18 backbone	*Mutual Guidance*	**28.7%**	**44.9%**	**29.9%**	**11.0%**	**32.2%**	**44.8%**
RetinaNet with	IoU_{anchor}-based	32.3%	50.3%	34.0%	14.3%	37.9%	46.7%
VGG-16 backbone	*Mutual Guidance*	**33.6%**	**50.8%**	**35.7%**	**15.4%**	**38.9%**	**48.8%**
RFBNet with	IoU_{anchor}-based	33.4%	51.6%	35.1%	14.2%	38.3%	49.1%
VGG-16 backbone	*Mutual Guidance*	**34.6%**	**52.0%**	**36.8%**	**15.8%**	**39.0%**	**51.1%**

Table 2. AP performance of different architectures for object detection on MS COCO dataset using 2 different anchor matching strategies: the usual IoU_{anchor}-based one and our complete approach marked as *Mutual Guidance*. The best score for each architecture is in bold.

Experiments on MS COCO. We then conduct experiments on the more difficult MS COCO [14] dataset and report our results in Table 2. Note that according to the scale range defined by MS COCO, APs of small, medium and large objects are listed. In this dataset also, our *Mutual Guidance* strategy consistently brings some performance gains compared to the IoU_{anchor}-based baselines. We notice that our AP gains on large objects is significant (around 2%). This is because larger objects generally have more matched positive anchors, which offers more room for improvements to our method. Since the *Mutual guidance* strategy only involves the training phase, and since there is no difference between IoU_{anchor}-based and our method during the evaluation phase, these improvements can be considered cost-free.

4.3 Qualitative analysis

Label assignment visualization. Here, we would like to explore the reasons for the performance improvements by visualizing the difference in the label assignment between the IoU_{anchor}-based strategy and the *Mutual Guidance* strategy during training. Some examples are shown in Figure 4. White dotted-line boxes represent ground truth boxes; Red anchor boxes are assigned as positive by IoU_{anchor}-based strategy, while considered as negative or ignored by *Localize to Classify* (the top two lines in Figure 4) or *Classify to Localize* (the bottom two lines in Figure 4); Green anchor boxes are assigned as positive by *Localize to Classify* but negative or ignored by IoU_{anchor}-based; Yellow anchor boxes are assigned as positive by *Classify to Localize* but negative or ignored by IoU_{anchor}-based. From these examples, we can conclude that the IoU_{anchor}-based strategy only assigns the "positive" label to anchors with sufficient IoU with the ground truth box, regardless of their content/context, whereas our proposed *Localize to Classify* and *Localize to Classify* strategies dynamically assign "positive" labels

Fig. 4. Visualization of the difference in the label assignment during training phase (images are resized to 320×320 pixels). Red, yellow and green anchor boxes are positive anchors assigned by IoU_{anchor}-based, *Localize to Classify* and *Classify to Localize* respectively. Zoom in to see details.

to anchors covering semantic discriminant parts of the object (e.g., upper body of a person, main body of animals), and assign "negative" labels to anchors with complex background, occluded parts, or anchors containing nearby objects. We believe that our proposed instance-adaptive strategies make the label assignment more reasonable, which is the main reason for performance increase.

Detection results visualization. Figure 5 illustrates on a few images from the PASCAL VOC dataset the different behaviours shown by our *Mutual Guidance* method and the baseline anchor matching strategy. As analysed in Section 3.3, we can find misaligned predictions (good at classification but poor at localization) from IoU_{anchor}-based anchor matching strategy. As shown in the figure, our method gives better results when different objects are close to each other in the image, e.g. "man riding a horse" or "man riding a bike". With the usual IoU_{anchor}-based anchor matching strategy, the instance localization and classification tasks are optimized independently of each other. Hence, it is possible that, during the evaluation phase, the classification prediction relies on one object whereas the bounding box regression targets the other object. However, such a problem is rarer with the *Mutual Guidance* strategy. Apparently, our anchor matching strategies introduce interactions between both tasks and makes the

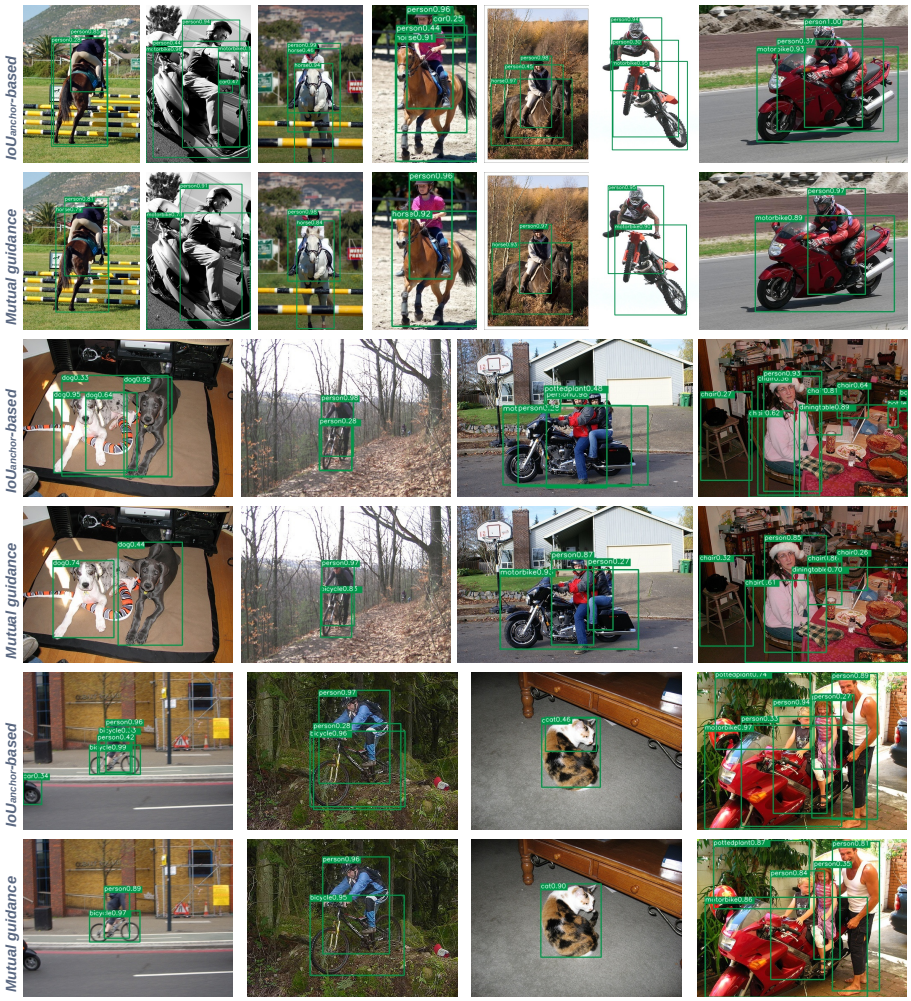

Fig. 5. Examples of detection results using an IoU_{anchor}-based anchor matching strategy (odd lines) and our proposed *Mutual Guidance* one (even lines). The results are given for all images after applying a Non-Maximum Suppression process with a IoU threshold of 50%. Zoom in to see details.

predictions of localization and classification aligned, which substantially eliminated such false positive predictions.

5 Conclusion

In this paper, we question the use of the IoU between predefined anchor boxes and ground truth boxes as a good criterion for anchor matching in object detection and study the interdependence of the two sub-tasks (i.e. localization and classification) involved in the detection process. We propose a *Mutual Guidance* mechanism, which provides an adaptive matching between anchors and objects by assigning anchor labels for one task according to the prediction quality on the other task and vice versa. We assess our method on different architectures and different public datasets and compare it with the traditional static anchor matching strategy. Reported results show the effectiveness and generality of this *Mutual Guidance* mechanism in object detection.

References

1. Ren, S., He, K., Girshick, R.B., Sun, J.: Faster R-CNN: towards real-time object detection with region proposal networks. In Cortes, C., Lawrence, N.D., Lee, D.D., Sugiyama, M., Garnett, R., eds.: Advances in Neural Information Processing Systems 28: Annual Conference on Neural Information Processing Systems 2015, December 7-12, 2015, Montreal, Quebec, Canada. (2015) 91–99
2. Dai, J., Li, Y., He, K., Sun, J.: R-FCN: object detection via region-based fully convolutional networks. In Lee, D.D., Sugiyama, M., von Luxburg, U., Guyon, I., Garnett, R., eds.: Advances in Neural Information Processing Systems 29: Annual Conference on Neural Information Processing Systems 2016, December 5-10, 2016, Barcelona, Spain. (2016) 379–387
3. Lin, T., Dollár, P., Girshick, R.B., He, K., Hariharan, B., Belongie, S.J.: Feature pyramid networks for object detection. In: 2017 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2017, Honolulu, HI, USA, July 21-26, 2017, IEEE Computer Society (2017) 936–944
4. Liu, S., Qi, L., Qin, H., Shi, J., Jia, J.: Path aggregation network for instance segmentation. In: 2018 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2018, Salt Lake City, UT, USA, June 18-22, 2018, IEEE Computer Society (2018) 8759–8768
5. Pang, J., Chen, K., Shi, J., Feng, H., Ouyang, W., Lin, D.: Libra R-CNN: towards balanced learning for object detection. In: IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2019, Long Beach, CA, USA, June 16-20, 2019, Computer Vision Foundation / IEEE (2019) 821–830
6. Cai, Z., Vasconcelos, N.: Cascade R-CNN: delving into high quality object detection. In: 2018 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2018, Salt Lake City, UT, USA, June 18-22, 2018, IEEE Computer Society (2018) 6154–6162
7. Liu, W., Anguelov, D., Erhan, D., Szegedy, C., Reed, S.E., Fu, C., Berg, A.C.: SSD: single shot multibox detector. In Leibe, B., Matas, J., Sebe, N., Welling, M., eds.: Computer Vision - ECCV 2016 - 14th European Conference, Amsterdam, The

Netherlands, October 11-14, 2016, Proceedings, Part I. Volume 9905 of Lecture Notes in Computer Science., Springer (2016) 21–37
8. Redmon, J., Farhadi, A.: Yolov3: An incremental improvement. CoRR **abs/1804.02767** (2018)
9. Lin, T., Goyal, P., Girshick, R.B., He, K., Dollár, P.: Focal loss for dense object detection. In: IEEE International Conference on Computer Vision, ICCV 2017, Venice, Italy, October 22-29, 2017, IEEE Computer Society (2017) 2999–3007
10. Li, Z., Zhou, F.: FSSD: feature fusion single shot multibox detector. CoRR **abs/1712.00960** (2017)
11. Liu, S., Huang, D., Wang, Y.: Receptive field block net for accurate and fast object detection. In Ferrari, V., Hebert, M., Sminchisescu, C., Weiss, Y., eds.: Computer Vision - ECCV 2018 - 15th European Conference, Munich, Germany, September 8-14, 2018, Proceedings, Part XI. Volume 11215 of Lecture Notes in Computer Science., Springer (2018) 404–419
12. 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: The Thirty-Third AAAI Conference on Artificial Intelligence, AAAI 2019, The Thirty-First Innovative Applications of Artificial Intelligence Conference, IAAI 2019, The Ninth AAAI Symposium on Educational Advances in Artificial Intelligence, EAAI 2019, Honolulu, Hawaii, USA, January 27 - February 1, 2019, AAAI Press (2019) 9259–9266
13. Everingham, M., Gool, L.V., Williams, C.K.I., Winn, J.M., Zisserman, A.: The pascal visual object classes (VOC) challenge. Int. J. Comput. Vis. **88** (2010) 303–338
14. Lin, T., Maire, M., Belongie, S.J., Hays, J., Perona, P., Ramanan, D., Dollár, P., Zitnick, C.L.: Microsoft COCO: common objects in context. In Fleet, D.J., Pajdla, T., Schiele, B., Tuytelaars, T., eds.: Computer Vision - ECCV 2014 - 13th European Conference, Zurich, Switzerland, September 6-12, 2014, Proceedings, Part V. Volume 8693 of Lecture Notes in Computer Science., Springer (2014) 740–755
15. Li, Y., Chen, Y., Wang, N., Zhang, Z.: Scale-aware trident networks for object detection. In: 2019 IEEE/CVF International Conference on Computer Vision, ICCV 2019, Seoul, Korea (South), October 27 - November 2, 2019, IEEE (2019) 6053–6062
16. Redmon, J., Divvala, S.K., Girshick, R.B., Farhadi, A.: You only look once: Unified, real-time object detection. In: 2016 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2016, Las Vegas, NV, USA, June 27-30, 2016, IEEE Computer Society (2016) 779–788
17. Zhang, S., Wen, L., Bian, X., Lei, Z., Li, S.Z.: Single-shot refinement neural network for object detection. In: 2018 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2018, Salt Lake City, UT, USA, June 18-22, 2018, IEEE Computer Society (2018) 4203–4212
18. Law, H., Deng, J.: Cornernet: Detecting objects as paired keypoints. In Ferrari, V., Hebert, M., Sminchisescu, C., Weiss, Y., eds.: Computer Vision - ECCV 2018 - 15th European Conference, Munich, Germany, September 8-14, 2018, Proceedings, Part XIV. Volume 11218 of Lecture Notes in Computer Science., Springer (2018) 765–781
19. Law, H., Teng, Y., Russakovsky, O., Deng, J.: Cornernet-lite: Efficient keypoint based object detection. CoRR **abs/1904.08900** (2019)

20. Zhou, X., Zhuo, J., Krähenbühl, P.: Bottom-up object detection by grouping extreme and center points. In: IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2019, Long Beach, CA, USA, June 16-20, 2019, Computer Vision Foundation / IEEE (2019) 850–859
21. Duan, K., Bai, S., Xie, L., Qi, H., Huang, Q., Tian, Q.: Centernet: Keypoint triplets for object detection. In: 2019 IEEE/CVF International Conference on Computer Vision, ICCV 2019, Seoul, Korea (South), October 27 - November 2, 2019, IEEE (2019) 6568–6577
22. Zhou, X., Wang, D., Krähenbühl, P.: Objects as points. CoRR **abs/1904.07850** (2019)
23. Zhu, C., He, Y., Savvides, M.: Feature selective anchor-free module for single-shot object detection. In: IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2019, Long Beach, CA, USA, June 16-20, 2019, Computer Vision Foundation / IEEE (2019) 840–849
24. Tian, Z., Shen, C., Chen, H., He, T.: FCOS: fully convolutional one-stage object detection. In: 2019 IEEE/CVF International Conference on Computer Vision, ICCV 2019, Seoul, Korea (South), October 27 - November 2, 2019, IEEE (2019) 9626–9635
25. Kong, T., Sun, F., Liu, H., Jiang, Y., Shi, J.: Foveabox: Beyond anchor-based object detector. CoRR **abs/1904.03797** (2019)
26. He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition. In: 2016 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2016, Las Vegas, NV, USA, June 27-30, 2016, IEEE Computer Society (2016) 770–778
27. Simonyan, K., Zisserman, A.: Very deep convolutional networks for large-scale image recognition. In Bengio, Y., LeCun, Y., eds.: 3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings. (2015)
28. Deng, J., Dong, W., Socher, R., Li, L., Li, K., Li, F.: Imagenet: A large-scale hierarchical image database. In: 2009 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR 2009), 20-25 June 2009, Miami, Florida, USA, IEEE Computer Society (2009) 248–255