Progressive Hard-case Mining across Pyramid Levels in Object Detection

Binghong Wu, Yehui Yang*, Dalu Yang,
Junde Wu, Haifeng Huang, Lei Wang, Junwei Liu, Yanwu Xu

Artificial Intelligence Group, Baidu Inc.
No.10 Xibeiwang East Road, Baidu Technology Park Building No.2
Haidian District, Beijing, China, 100193

Abstract

In object detection, multi-level prediction (e.g., FPN, YOLO) and resampling skills (e.g., focal loss, ATSS) have drastically improved one-stage detector performance. However, how to improve the performance by optimizing the feature pyramid level-by-level remains unexplored. We find that, during training, the ratio of positive over negative samples varies across pyramid levels (*level imbalance*), which is not addressed by current one-stage detectors. To mediate the influence of level imbalance, we propose a Unified Multi-level Optimization Paradigm (UMOP) consisting of two components: 1) an independent classification loss supervising each pyramid level with individual resampling considerations; 2) a progressive hard-case mining loss defining all losses across the pyramid levels without extra level-wise settings. With UMOP as a plug-and-play scheme, modern one-stage detectors can attain a $\sim$ 1.5AP improvement with fewer training iterations and no additional computation overhead. Our best model achieves **55.1** AP on COCO *test-dev*. Code is available at https://github.com/zimoqingfeng/UMOP.

Introduction

One-stage object detectors are popular in practical applications because of their higher efficiency and lower computation cost compared with multi-stage detectors (Zou et al. 2019). Recently, one-stage detectors gradually catches up with multi-stage detectors, benefiting from in-depth studies such as the improvements of model architectures, loss functions, target assignment strategies, etc. (Liu et al. 2016; Lin et al. 2017; Zhang et al. 2021; Rezatofighi et al. 2019; Tian et al. 2019; Zhang et al. 2020)

Equipped with Feature Pyramid Network (FPN) (Lin et al. 2017) and focal loss (Lin et al. 2017), one-stage detectors can provide more accurate predictions with dense candidates. FPN provides dense candidates with diverse receptive fields. It improves the performance by solving the mismatching problems between the receptive field and the target scale. Focal loss reweights all the potential proposals according to the margin between the assigned label and its own probability. It improves the performance by enabling online hard-case mining. These two techniques have been widely used in the leading studies, including anchor-based detectors and anchor-free detectors (Lin et al. 2017; Tian et al. 2019; Zhu, He, and Savvides 2019; Zhang et al. 2020; Tan, Pang, and Le 2020; Kong et al. 2020).

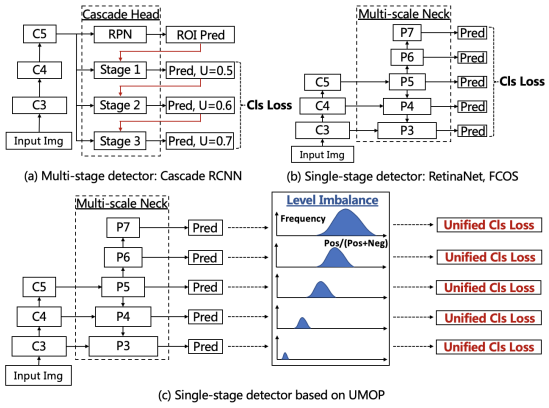

Figure 1: Differences and similarities. (a) Multi-stage detectors resample proposals with different IOU thresholds, shown as U in heads. (b) One-stage detectors utilize various pyramid levels for dense prediction, with a single reweighting loss form for all potentials. (c) UMOP optimizes all samples with a dynamic hyperparameter adjusting strategy, based on the convergence situation at which level they are.

Despite their individual contributions, the synergy between focal loss and FPN is not fully explored. Focal loss is designed from a *global* optimization perspective, using two hyperparameters to control the hard-case mining degree to mediate class imbalance. In contrast, FPN improves the performance with its divide-and-conquer solution by capturing different object scales with *different pyramid levels* (Chen et al. 2021; Zhang et al. 2020). Yet, shown as Fig. 2, the ratio of positive over negative samples can vary across pyramid levels, which we refer to as *level imbalance*. Therefore, it seems a single global setting of focal loss may mislead the optimization of individual pyramid levels.

Multi-stage detectors can alleviate imbalance by applying different resampling mechanism for different stages, based on the localization quality (Cai and Vasconcelos 2018). Such a framework can progressively improve the performance by refine the locations and confidence scores (Ren et al. 2015; Cai and Vasconcelos 2018; Pang et al. 2019; Chen et al. 2019). Also, multi-stage detectors such as Cascade R-CNN

*Corresponding Author: yangyehuisw@126.com

(Cai and Vasconcelos 2018) and HTC (Chen et al. 2019) benefit from the resampling mechanism in different stages with the consideration of the matching quality. A natural question arises: can we utilize these resampling mechanisms on multi-level one-stage detectors to avoid the level imbalance phenomenon through the lens of divide-and-conquer optimization?

In this paper, we first perform a statistical analysis to verify that the ratio of positive over negative samples varies across pyramid levels, i.e. the *level imbalance* phenomenon widely exists. Inspired by the multi-stage detector architecture, we propose a Unified Multi-level Optimization Paradigm (UMOP) to address the level imbalance and improve overall performance. As shown in Fig. 1, UMOP consists of two components: 1) an independent classification loss supervising each pyramid level with individual resampling considerations; 2) a progressive hard-case mining loss defining all losses across the pyramid levels, solving the level imbalance without extra level-wise settings. We provide detailed method descriptions, ablation studies, and comparison results in the following sections.

Our main contributions are:

- To the best of our knowledge, we are the first to experimentally show that the performance of FPN is limited by the level imbalanced problem to some extent.
- With our proposed methods to mediate level imbalance, modern one-stage detectors can attain a $\sim$ 1.5AP improvement with fewer training iterations and no additional computation overhead.
- Our best model achieves **55.1** AP on MS COCO *test-dev*, which is by far the SOTA in one-stage detectors.

Related Works

Resampling methods in Multi-stage Detectors. Originating from the sliding-window methods (Dalal and Triggs 2005; Felzenszwalb, Girshick, and McAllester 2010), the two-stage detector inherits the inherent paradigm of locating first and refining later for achieving a better performance (Girshick et al. 2014; Girshick 2015; Ren et al. 2015). From sliding-window methods to selective search, and then to Region Proposal Network, it is obvious that the development of ROI (region of interest) extraction methods promotes the performance of detectors significantly. From our point of view, the Region Proposal Network (Ren et al. 2015) could be regarded as a data-driven resampling scheduler according to the matching quality, discarding a large amount of low-quality negative samples to alleviate the target imbalance during optimization. Additionally, the head of multi-stage detectors (Cai and Vasconcelos 2018; Chen et al. 2019) could eliminate the low-quality predictions step by step for solving the quality mismatch during both training and inference. Recently, there are still many further studies on the resampling method with a variety of novel perspectives. In the training procedure, IoU-balanced sampling (Pang et al. 2019) has been proposed to assign samples based on matching quality in each pyramid level. From the view of model design, a lot of studies (Song, Liu, and Wang 2020; Zhu et al. 2021) try to decouple the recognition and localization task, promoting performance by sampling decoupled features from the backbone separately. Delving into high quality object detection, the IoU-guided NMS method could bring the prediction of localization quality into NMS for a better post-processing calibration (Jiang et al. 2018).

Resampling methods in One-stage Detectors. The studies on resampling mechanism make one-stage detectors totally comparable to two-stage detectors. Inspired by OHEM, focal loss (Lin et al. 2017) has been proposed to make the model automatically put more attention on hard cases by extremely down-weighting the easy cases' losses and slightly reducing the weights of hard cases, which has been widely used recently. With much more dense candidates provided by multi-level architectures (Liu et al. 2016; Lin et al. 2017; Redmon and Farhadi 2017), reweighting has been regarded as a key method to achieve a better performance stably. Such a design could also keep robust contributions on a variety of target assignment strategies, leading to an explosive growth of anchor-free detectors (Zhu, He, and Savvides 2019; Tian et al. 2019). Besides, under the influence of this breakthrough, a lot of studies take such an online hard example mining idea into account much more deeply. For example, a penalty-reduced pixel-wise logistic regression loss based on focal loss has been proposed to optimize the prediction of center points in CenterNet (Zhou, Wang, and Krähenbühl 2019), and an improvement of focal loss with respect to target IoU could also make a stable improvement (Zhang et al. 2021). In addition, a gradient harmonizing mechanism (Li, Liu, and Wang 2019) is proposed to ensure the optimization robustness during training. Meanwhile, there are still a lot of studies based on the focal loss for better quality estimation of the detector's proposals (Li et al. 2020, 2021), leading to remarkable improvements as well.

Resampling methods in Target assignment. From a unified perspective on both anchor-based detectors and anchor-free detectors, the issue of target assignment has been gradually noticed in recent years (Ke et al. 2020). In ATSS (Zhang et al. 2020), it has been proved that the key difference between RetinaNet and FCOS is the target assignment strategy, which leads the difference of target distribution during training and results in a performance gap. Correspondingly, a novel target assignment strategy with a dynamic IoU threshold has been well designed on each pyramid level for a better performance. Besides, AutoAssign (Zhu et al. 2020) improves the performance by a fully data-driven method with as few hand-crafted settings as possible. By defining it as an optimization problem, PAA (Kim and Lee 2020) and OTA (Ge et al. 2021) set specific target and find suitable online strategy separately during each training iteration.

The Statistical Analysis on Level Imbalance

Multi-level architectures are widely used in one-stage detectors and have improved the detection performance drastically. In SSD and YOLO series, the multi-level prediction framework improves the detector performance by providing much more dense candidates (Liu et al. 2016; Redmon and Farhadi 2017). Unlike SSD and YOLO, where predictions are directly made at each level of the feature maps, FPN adds

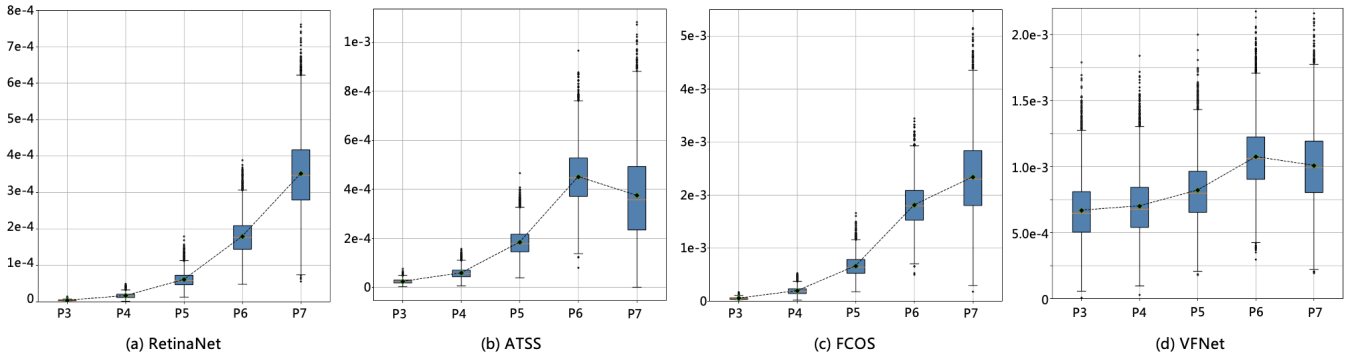

Figure 2: The statistical analysis on level imbalance. P3 to P5 is defined as the level index from FPN, indicating the results from different pyramid level predictions. In each training iteration, the proportions of positive samples to total samples in each pyramid level are recorded and summarized as box plots. The height of each box surround by upper quartile and lower quartile indicates the variance of the recorded proportions per pyramid level, and the mean values are drawn as dots and connected with dotted lines.

a top-down pathway and lateral connections for feature combination. In our work, we analyze both anchor-based detectors and anchor-free detectors to confirm the level imbalance issue from the general point of view.

The experiment settings on level imbalance. Without loss of generality, we perform the analysis based on two anchor-based detectors and two anchor-free detectors: RetinaNet, ATSS, FCOS and VFNet (Lin et al. 2017; Zhang et al. 2020; Tian et al. 2019; Zhang et al. 2021). We use MS COCO (Lin et al. 2014) dataset for all analysis, which contains 115K images in the *trainval35k* used for training one epoch. We apply FPN as the model neck for C3 to C5 from the backbone, generating feature maps named as P3 to P7 of five different resolutions (shown in Figure 1 (b) and (c)). Besides, we fix all the other settings for a fair comparison.

The statistical analysis on level imbalance. In our analysis, we carefully observe the influence of target distribution across levels, while keeping the image resolution and the complexity of the model fixed.

There are many ways to assign targets for training. Anchor-based detectors assign the targets based on the matching quality between anchors and ground truths, and anchor-free detectors assign the targets according to the distance of key points and object scales. For a general analysis without one specific circumstance, the statistical analyses on four different detectors have been done within a whole training epoch: during each training iteration, the proportions of positive samples to total samples in each pyramid level has been recorded, and summarized as box-plot in Figure 2.

As shown in Figure 2, it is obvious that the target distribution among each pyramid level is totally different. According to all the four detectors, each candidate in P7 is of a higher probability to match a positive sample during training. The results clearly reveal the level imbalance phenomenon in multi-level detectors, no matter in what circumstance. Therefore, we doubt that focal loss with fixed hyperparameter settings could not make a good trade-off for all pyramid levels at the same time.

Unified Multi-level Optimization Paradigm

We propose a UMOP to mediate the level imbalance in one-stage detectors. The proposed method consists of two parts: 1) a Level-wise Resampling Paradigm (LRP), which sets an independent classification loss supervising each pyramid level with individual resampling considerations, and 2) a Progressive Focal Loss, which adjusts the hard-case mining degree progressively, based on the prediction of positive samples in each pyramid level.

Level-wise Resampling Paradigm

Multi-stage detectors mitigates the imbalance phenomenon by optimizing with an iterative resampling mechanism: alternating between proposals rescoring and location refinement. This mechanism can exclude the easy low-quality samples and prevent the quality mismatch problem, making an improvement on high-quality predictions. We propose Level-wise Resampling Paradigm (LRP), a similar paradigm in one-stage detectors for high-quality predictions based on diverse hard-case metrics among pyramid levels.

LRP is a multi-level optimization paradigm addressing sample imbalance across different pyramid levels. As shown in Fig.1 (c), we calculate the classification loss in each pyramid level independently during training. The following Eq. (1) is the total classification loss, which is defined as the mean of all the level-wise classification losses.

$$\text{Loss}_{cls} = \frac{1}{L}\sum_{l=1}^{L}\text{Loss}_l(P_l, Y_l). \tag{1}$$

In Eq. (1), L is the number of pyramid levels used for prediction in the one-stage detector, P_l is the level-wise prediction results generated only from the l^{th} level, and Y_l indicates the assigned label by a specific target assignment strategy. Loss_l indicates the total loss of the l^{th} level. This level-wise loss term gives each pyramid level flexibility to adapt for the specific sample imbalance it faces during optimization.

Progressive Focal Loss

For level-wise optimization without any unnecessary settings, we propose Progressive Focal Loss (PFL) to automatically adjust the degree of hard case mining based on the prediction of positive samples in each pyramid level. Besides, the proposed loss can keep effective gradients for hard samples, with the respect to the metrics on the convergence situation.

For binary classification, the sigmoid focal loss is (2):

$$\mathrm{FL}(p_i, y_i) = \begin{cases} -\alpha(1-p_i)^{\gamma}\log(p_i), & y_i = 1 \\ -(1-\alpha)p_i^{\gamma}\log(1-p_i), & y_i = 0. \end{cases} \tag{2}$$

In Eq. (2), p_i is the model's final prediction result. $y_i \in \{0, 1\}$ is the assigned label for the specific prediction in each grid on every pyramid level. In the original focal loss, the hyperparameter α is a well-tuned constant value to keep the gradient balance between the positive and the negative samples, and the γ is designed to alleviate the imbalance between easy samples and hard samples, through a dynamic sampling method according to the model's prediction probability. With appropriate α and γ, the gradients of massive easy and negative samples are significantly compressed. In practice, a lower α is always corresponding to a higher γ, to modulate the emphasis on positive samples when more easy negatives are hugely down-weighted (Lin et al. 2017), and keep the balance among all losses for model convergence as well.

From a divide-and-conquer perspective, we propose a novel dynamic hyperparameter adjusting strategy for α and γ. The proposed strategy could dynamically adjust the strength of hard-case mining according to their own convergence situation in each pyramid level. For the imbalance diversity among pyramid levels, the actual hyperparameter applied in different level is totally different.

During training, the cases are generally harder in the early training stage while relatively easier in the later. For a corresponding guide in the optimization procedure, the hyperparameter with our proposed adjusting strategy could be automatically tuned guided by the convergence situation in each pyramid level independently. PFL could be shown in the following Eq. (3): all the original settings have been kept except for the proposed adjusting schedule on α_{ad} and γ_{ad}.

$$\mathrm{PFL}(p_i, y_i) = \begin{cases} -\alpha_{ad}(1-p_i)^{\gamma_{ad}}\log(p_i), & y_i = 1 \\ -(1-\alpha_{ad})p_i^{\gamma_{ad}}\log(1-p_i), & y_i = 0. \end{cases} \tag{3}$$

As a hard case mining setting, γ_{ad} is designed with the consideration of the positive samples' prediction quality in each pyramid level independently. Meanwhile, inspired by the previous experimental analysis (Lin et al. 2017), α_{ad} is set to follow γ_{ad} adjustment timely, which could keep the balance between positive samples and negative samples. The detailed definitions could be shown in Eq. (4) and Eq. (5).

$$\gamma_{ad} = -\log(\frac{1}{n_{pos}}\sum_{i=1}^{n} \mathrm{y}_i \cdot \mathrm{p}_i). \tag{4}$$

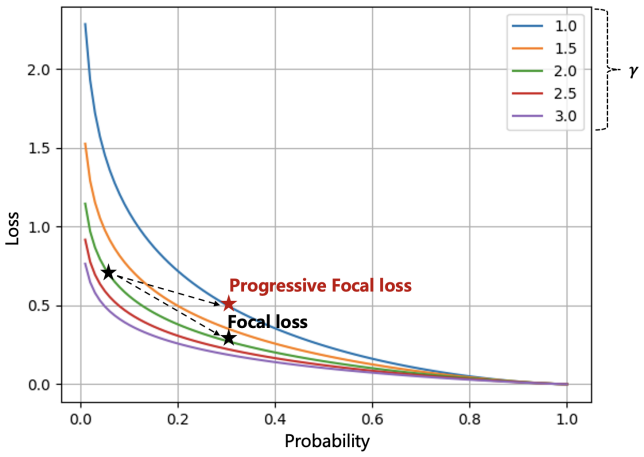

Figure 3: The comparison between Progressive Focal Loss and Focal Loss. During training, the cases are generally harder in the early training stage while relatively easier in the later. Progressive Focal Loss could strengthen the hard-case mining degree progressively.

Algorithm 1: Unified Multi-level Optimization Paradigm

Input:
P is a set of prediction results
Y is a set of ground truth corresponding to prediction
Output: $\mathrm{Loss_{cls}}$ is the total classification loss

split P into $\mathrm{S_p} = [\mathrm{P_1}, \mathrm{P_2}, \dots]$ by each pyramid level.
split Y into $\mathrm{S_y} = [\mathrm{Y_1}, \mathrm{Y_2}, \dots]$ by each pyramid level.
for predictions $\mathrm{P_l} \in \mathrm{S_p}$ and ground truths $\mathrm{Y_l} \in \mathrm{S_y}$ **do**
    calculate γ_{ad} by $\mathrm{y_i} \in \mathrm{Y_l}$ and $\mathrm{p_i} \in \mathrm{P_l}$ according to Eq. (4).
    calculate α_{ad} by γ_{ad} according to Eq. (5).
    calculate $\mathrm{PFL_l}$ according to Eq. (3) as the $\mathrm{l^{th}}$ level loss.
end for
calculate the $\mathrm{Loss_{cls}}$ according to the Eq. (1)
return $\mathrm{Loss_{cls}}$

$$\alpha_{ad} = w/\gamma_{ad}. \tag{5}$$

The function of γ_{ad} is similar to the cross entropy (CE) loss, which naturally reflecting the convergence situation in a pyraimd level. In Eq. (4) for γ_{ad} adjusting, the strategy designed with the perspective of level-wise optimization: $\mathrm{y_i}$ indicates the assigned label for one sample in a specific pyramid level, and $\mathrm{p_i}$ is defined as the corresponding probability result. Therefore, $\mathrm{y}_i \cdot \mathrm{p}_i$ indicates the probability for the positive sample, and defined as 0 for the negative sample. n is defined as the number of total samples per pyramid levels, and n_{pos} indicates the number of total positive samples per pyramid levels. To ensure the training stability, the adjusted hyperparameter is clamped within a valid interval $\gamma_{ad} \in [\gamma - \delta, \gamma + \delta]$ with δ set as an constant for all the experiments as well.

In Eq. (5), the adjusting schedule on α_{ad} is designed to follow the trend of γ_{ad}. Specifically, w is a constant to make α_{ad} calculated by γ_{ad} from the negative correlation. Besides, as a level-wise resampling method, it is important to

note that α_{ad} and γ_{ad} is calculated instantly and do not generate any extra derivation operation during training.

Therefore, γ_{ad} reflects the whole level convergence situation with a bigger value at the beginning and then gradually decreasing with the optimization proceeding. Shown as Fig. 3, such a dynamic adjusting schedule could make the model emphasize hard cases at the beginning, and gradually increase the distinguishing ability when the hard and easy cases are not discriminative enough.

The total procedure of UMOP is described in Algorithm 1. We introduce the whole paradigm as a unified form for easily deployed into any one-stage detector with a multi-level structure.

Experiments

Implementation details

Based on the large-scale detection benchmark MS COCO (Lin et al. 2014), we follow the common practice in previous works (Ren et al. 2015; Tian et al. 2019) that set the COCO *trainval35k* split (115K images) for training and *minival* split (5K images) as validation. In the ablation study, the evaluation results have been listed under many conditions as detailed as possible. For a fair comparison to state of the art, our main results on the *test-dev* split (20K images) have also been reported by uploading the final results to the evaluation web server.

Network Setting. We keep all existing settings as default in the released code during training, including the model architecture and the related model-design settings. If not otherwise specified, we initialize our backbone networks with the ImageNet (ILSVC) pretrained weights. Besides, we also follow the anchor-related settings (i.e., the number of anchors, anchor scales, anchor aspect ratios, etc.) in RetinaNet and ATSS, keeping the original target distribution for a fair comparison. For single-scale training in ablation study, we resize the shorter side of images to 800 and the longer side to less or equal to 1333, keeping the aspect ratio. For multi-scale training, we randomly set the shorter side between 640 and 800.

Optimization. PFL is set to optimize each pyramid level independently. In PFL, w is set as $\alpha \cdot \gamma$ (0.5) and δ is set as 0.5 for our main results. Besides, all other original settings are kept. If not specified, the model is trained with stochastic gradient descent (SGD) for the same epochs with an initial learning rate of 0.01 and a minibatch of 16 images. A linear warmup schedule has been adopted in the first 500 iterations with a warmup ratio being 0.001. Weight decay is set as 0.0001, and momentum is set as 0.9. For bounding box regression, GIoU Loss is adopted on the ATSS, and L1 Loss has been applied for the RetinaNet respectively.

Ablation study

We perform the evaluations on RetinaNet and ATSS to analyse the general improvement over commonly used detectors. We also compare the convergence speed of ATSS with or without our method.

General Promotion on Different Detectors. We first investigate our general contribution on different detectors. The proposed paradigm can be easily deployed in almost all one-stage detectors as a plug-and-play component. Based on different backbones, the general improvements over RetinaNet and ATSS are in Table 1. Qualitative comparison between ATSS and our method are in Fig. 4. According to the visualization results, our method could solve a wide range of hard cases, including fuzzy objects(shown in 3-banana), highly overlapped objects(shown in 5-airplane, 7-bottle, 8-bench), small objects(shown in 1-kite) and objects with extremely aspect ratio (shown in 2-traffic light, 4-surfboard).

Method	Backbone	w/ PFL	w/ LRP	AP
RetinaNet	R-50			35.7
RetinaNet	R-50	✓		36.7
RetinaNet	R-50	✓	✓	**36.9**
RetinaNet w / imprv.	R-101			38.9
RetinaNet w / imprv.	R-101	✓		39.7
RetinaNet w / imprv.	R-101	✓	✓	**40.5**
ATSS	R-50			39.3
ATSS	R-50	✓		40.1
ATSS	R-50	✓	✓	**40.4**
ATSS w / imprv.	R-101			46.1
ATSS w / imprv.	R-101	✓		46.7
ATSS w / imprv.	R-101	✓	✓	**47.6**
ATSS w / imprv.	X-101			47.7
ATSS w / imprv.	X-101	✓		48.4
ATSS w / imprv.	X-101	✓	✓	**48.8**

Table 1: Average precision (AP) improvements on COCO *minival*. 'R': ResNet, 'X': ResNeXt-64x4d. We show the performance improving when only applying PFL and the whole method. For a stronger baseline, we optionally applied multi-scale training and deformable convolutional layers on both detectors.

According to Table 1, for RetinaNet, the original work reported AP values of 35.7 for ResNet-50 backbone and 38.9 for ResNet-101 backbone (equipped with multi-scale training for the larger backbone). With PFL only, the AP values increase by 1.0 and 0.8, respectively. With LRP applied, the AP values further increase by 0.2 and 0.8. For ATSS, the original work reported AP values of 39.3, 46.1, and 47.7 for ResNet-50, ResNet-101, and ResNeXt-64x4d-101, respectively (equipped with multi-scale training and DCN-v2 for ResNet-101 and ResNeXt-64x4d-101). With PFL only, the AP values increase by 0.8, 0.6, and 0.7, respectively. With LRP applied, the AP values further increase by 0.3, 0.9, and 0.4. These results clearly show that our method could improve the performance on different detectors with different backbones.

Analysis on Convergence Speed. In this section, we compare our methods with the original ATSS on different backbones (ResNet-101 and ResNeXt-101). For a fair comparison, the same multi-scale training strategy and the deformable convolution layer (DCN-v2) are both equipped in our work and the baseline. Based on COCO *test-dev*, the performance comparison are as Table 2.

The original ATSS with ResNet-101 and ResNeXt-101 takes 24 epochs before convergence. Equipped with our method, the performance improves by 1.4 AP with fewer

(a) ATSS (b) ATSS w/ UMOP

Figure 4: Some detection results on COCO *minival*. ResNet-50 is used as the backbone and the score threshold for visualization is 0.3. As shown in the figure, UMOP works well with a wide range of objects including crowded, highly overlapped, and extremely small objects.

Method	Backbone	Epoch	AP	AP_{50}	AP_{75}	AP_s	AP_m	AP_l
ATSS	ResNet-101-DCN	24	46.3	64.7	50.4	27.7	49.8	58.4
ATSS w/ UMOP		**18**	**47.7**	**66.9**	**52.1**	**29.1**	**51.0**	**59.7**
ATSS	ResNeXt-101-64x4d-DCN	24	47.7	66.5	51.9	29.7	50.8	59.4
ATSS w/ UMOP		**18**	**49.1**	**68.5**	**53.6**	**30.8**	**52.6**	**61.1**

Table 2: Convergence speed analysis on COCO *test-dev*, our method achieves a better performance with fewer iterations.

training epochs. Also, with our method the AP_{50}, AP_{75}, and AP with different object scales are also improved.

Comparison with State of the Art

We evaluate the ATSS with UMOP on COCO *test-dev* and make a comparison with recent state-of-the-art models, including both one-stage detectors and two-stage detectors. Table 3 lists our results and the performance of some popular models over recent years. Here we combined our method with some more advanced works, more intensive computation, and the best hyperparameter settings to achieve a more competitive final result. All our results in Table 3 adopt the multi-scale training strategy, and the training epoch is set to 24 to ensure convergence. For training with Swin Transformer, 4 patches and 7 windows are kept default as its inner structure settings. '1K' in 2^{nd} column means the backbone is pretrained from the ImageNet-1K dataset, and '22K' indicates the ImageNet-22K correspondingly. Multi-scale testing is adopted on our best single model. Under such strategies, the images are resized correspondingly, with the shorter side varying from 800 to 1200 for the final results.

All our experiments are trained on 8 Tesla-V100-16GB GPUs, except that the experiments with Swin-L-22K are trained with 8 Tesla-P40-24GB GPUs and NVIDIA-Apex toolkit (utilizing automatic mixed precision for GPU memory saving). Compared with the high-performance detectors with long training epochs and large image resolution, our model achieves a high performance **55.1 AP** on COCO *test-dev* with a basic laboratory setting and the most commonly-used image resolution.

Conclusion

We observed the level imbalance problem in one-stage object detectors. To mediate the level imbalance, we proposed a novel classification loss to progressively adjust the hard case mining during training. Our proposed method made a

Method	Backbone	Size	Epoch	AP	AP_{50}	AP_{75}	AP_S	AP_M	AP_L
multi-stage:									
Faster R-CNN w/ FPN (Lin et al. 2017)	R-101	800*	24	36.2	59.1	39.0	18.2	39.0	48.2
Cascade R-CNN (Cai and Vasconcelos 2018)	R-101	800*	18	42.8	62.1	46.3	23.7	45.5	55.2
CenterNet2 (Zhou, Koltun, and Krähenbühl 2021)	X-101-DCN	800*	24	50.2	68.0	55.0	31.2	53.5	63.6
one-stage:									
CornerNet (Law and Deng 2018)	Hg-104	512	200	40.6	56.4	43.2	19.1	42.8	54.3
CenterNet (Zhou, Wang, and Krähenbühl 2019)	Hg-104	512	190	44.9	62.4	48.1	25.6	47.4	57.4
FASF (Zhu, He, and Savvides 2019)	X-101	800*	18	42.9	63.8	46.3	26.6	46.2	52.7
FCOS (Tian et al. 2019)	X-101	800*	24	44.7	64.1	48.4	27.6	47.5	55.6
ATSS (Zhang et al. 2020)	X-101-DCN	800*	24	47.7	66.5	51.9	29.7	50.8	59.4
EfficientDet (Tan, Pang, and Le 2020)	EffNet-B7	1536	450	55.1	74.3	59.9	37.2	57.9	**68.0**
ours:									
ATSS w/ UMOP	R2-101-DCN	960*	24	50.3	69.5	54.9	31.8	53.8	63.1
ATSS w/ UMOP	Swin-S-1K	960*	24	50.3	70.0	54.9	32.0	53.6	63.1
ATSS w/ UMOP	Swin-B-22K	960*	24	51.9	71.6	56.6	33.4	55.4	65.0
ATSS w/ UMOP	Swin-L-22K	960*	24	53.1	72.7	58.0	34.9	56.5	66.4
ATSS w/ UMOP *(multi-scale testing)*	Swin-L-22K	**960***	**24**	**55.1**	**74.2**	**60.7**	**38.1**	**58.6**	66.8

Table 3: Single-model performance comparison with state-of-the-art detectors on COCO *test-dev*, 'R': ResNet, 'X': ResNeXt-64x4d, 'R2': Res2Net, 'Hg': Hourglass, 'EffNet': EfficientNet 'DCN': Deformable convolution network v2, 960*: resize the shorter side to 960 and the longer side less or equal to 1333 with the aspect ratio kept, MS_{train}: training image scale range 1333×[640:800] for 800*, and 1333×[480:960] for 960*.

stable improvement on common one-stage detectors in term of average precision, with fewer training iterations and no additional computation overhead.

References

[1] Cai, Z.; and Vasconcelos, N. 2018. Cascade r-cnn: Delving into high quality object detection. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 6154–6162.

[2] Chen, K.; Pang, J.; Wang, J.; Xiong, Y.; Li, X.; Sun, S.; Feng, W.; Liu, Z.; Shi, J.; Ouyang, W.; et al. 2019. Hybrid task cascade for instance segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 4974–4983.

[3] Chen, Q.; Wang, Y.; Yang, T.; Zhang, X.; Cheng, J.; and Sun, J. 2021. You only look one-level feature. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 13039–13048.

[4] Dalal, N.; and Triggs, B. 2005. Histograms of oriented gradients for human detection. In *2005 IEEE computer society conference on computer vision and pattern recognition (CVPR'05)*, volume 1, 886–893. Ieee.

[5] Felzenszwalb, P. F.; Girshick, R. B.; and McAllester, D. 2010. Cascade object detection with deformable part models. In *2010 IEEE Computer society conference on computer vision and pattern recognition*, 2241–2248. Ieee.

[6] Ge, Z.; Liu, S.; Li, Z.; Yoshie, O.; and Sun, J. 2021. OTA: Optimal Transport Assignment for Object Detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 303–312.

[7] Girshick, R. 2015. Fast r-cnn. In *Proceedings of the IEEE international conference on computer vision*, 1440–1448.

[8] Girshick, R.; Donahue, J.; Darrell, T.; and Malik, J. 2014. Rich feature hierarchies for accurate object detection and semantic segmentation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 580–587.

[9] Jiang, B.; Luo, R.; Mao, J.; Xiao, T.; and Jiang, Y. 2018. Acquisition of localization confidence for accurate object detection. In *Proceedings of the European conference on computer vision (ECCV)*, 784–799.

[10] Ke, W.; Zhang, T.; Huang, Z.; Ye, Q.; Liu, J.; and Huang, D. 2020. Multiple anchor learning for visual object detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 10206–10215.

[11] Kim, K.; and Lee, H. S. 2020. Probabilistic anchor assignment with iou prediction for object detection. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XXV 16*, 355–371. Springer.

[12] Kong, T.; Sun, F.; Liu, H.; Jiang, Y.; Li, L.; and Shi, J. 2020. Foveabox: Beyound anchor-based object detection. *IEEE Transactions on Image Processing*, 29: 7389–7398.

[13] Law, H.; and Deng, J. 2018. Cornernet: Detecting objects as paired keypoints. In *Proceedings of the European conference on computer vision (ECCV)*, 734–750.

[14] Li, B.; Liu, Y.; and Wang, X. 2019. Gradient harmonized single-stage detector. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, 8577–8584.

[15] Li, X.; Wang, W.; Hu, X.; Li, J.; Tang, J.; and Yang, J. 2021. Generalized focal loss v2: Learning reliable

localization quality estimation for dense object detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 11632–11641.

[16] Li, X.; Wang, W.; Wu, L.; Chen, S.; Hu, X.; Li, J.; Tang, J.; and Yang, J. 2020. Generalized focal loss: Learning qualified and distributed bounding boxes for dense object detection. *arXiv preprint arXiv:2006.04388*.

[17] Lin, T.-Y.; Dollár, P.; Girshick, R.; He, K.; Hariharan, B.; and Belongie, S. 2017. Feature pyramid networks for object detection. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2117–2125.

[18] Lin, T.-Y.; Goyal, P.; Girshick, R.; He, K.; and Dollár, P. 2017. Focal loss for dense object detection. In *Proceedings of the IEEE international conference on computer vision*, 2980–2988.

[19] Lin, T.-Y.; Maire, M.; Belongie, S.; Hays, J.; Perona, P.; Ramanan, D.; Dollár, P.; and Zitnick, C. L. 2014. Microsoft coco: Common objects in context. In *European conference on computer vision*, 740–755. Springer.

[20] Liu, W.; Anguelov, D.; Erhan, D.; Szegedy, C.; Reed, S.; Fu, C.-Y.; and Berg, A. C. 2016. Ssd: Single shot multibox detector. In *European conference on computer vision*, 21–37. Springer.

[21] Pang, J.; Chen, K.; Shi, J.; Feng, H.; Ouyang, W.; and Lin, D. 2019. Libra r-cnn: Towards balanced learning for object detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 821–830.

[22] Redmon, J.; and Farhadi, A. 2017. YOLO9000: better, faster, stronger. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 7263–7271.

[23] Ren, S.; He, K.; Girshick, R.; and Sun, J. 2015. Faster r-cnn: Towards real-time object detection with region proposal networks. *Advances in neural information processing systems*, 28: 91–99.

[24] Rezatofighi, H.; Tsoi, N.; Gwak, J.; Sadeghian, A.; Reid, I.; and Savarese, S. 2019. Generalized intersection over union: A metric and a loss for bounding box regression. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 658–666.

[25] Song, G.; Liu, Y.; and Wang, X. 2020. Revisiting the sibling head in object detector. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 11563–11572.

[26] Tan, M.; Pang, R.; and Le, Q. V. 2020. Efficientdet: Scalable and efficient object detection. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 10781–10790.

[27] Tian, Z.; Shen, C.; Chen, H.; and He, T. 2019. Fcos: Fully convolutional one-stage object detection. In *Proceedings of the IEEE/CVF international conference on computer vision*, 9627–9636.

[28] Zhang, H.; Wang, Y.; Dayoub, F.; and Sunderhauf, N. 2021. Varifocalnet: An iou-aware dense object detector. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 8514–8523.

[29] Zhang, S.; Chi, C.; Yao, Y.; Lei, Z.; and Li, S. Z. 2020. Bridging the gap between anchor-based and anchor-free detection via adaptive training sample selection. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 9759–9768.

[30] Zhou, X.; Koltun, V.; and Krähenbühl, P. 2021. Probabilistic two-stage detection. *arXiv preprint arXiv:2103.07461*.

[31] Zhou, X.; Wang, D.; and Krähenbühl, P. 2019. Objects as points. *arXiv preprint arXiv:1904.07850*.

[32] Zhu, B.; Song, Q.; Yang, L.; Wang, Z.; Liu, C.; and Hu, M. 2021. CPM R-CNN: Calibrating point-guided misalignment in object detection. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, 3248–3257.

[33] Zhu, B.; Wang, J.; Jiang, Z.; Zong, F.; Liu, S.; Li, Z.; and Sun, J. 2020. Autoassign: Differentiable label assignment for dense object detection. *arXiv preprint arXiv:2007.03496*.

[34] Zhu, C.; He, Y.; and Savvides, M. 2019. Feature selective anchor-free module for single-shot object detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 840–849.

[35] Zou, Z.; Shi, Z.; Guo, Y.; and Ye, J. 2019. Object detection in 20 years: A survey. *arXiv preprint arXiv:1905.05055*.