Generalized Focal Loss: Towards Efficient Representation Learning for Dense Object Detection

Xiang Li, Chengqi Lv, Wenhai Wang, Gang Li, Lingfeng Yang, and Jian Yang

Abstract—Object detection is a fundamental computer vision task that simultaneously predicts the category and localization of the targets of interest. Recently one-stage (also termed "dense") detectors have gained much attention over two-stage ones due to their simple pipeline and friendly application to end devices. Dense object detectors basically formulate object detection as dense classification and localization (i.e., bounding box regression). The classification is usually optimized by Focal Loss and the box location is commonly learned under Dirac delta distribution. A recent trend for dense detectors is to introduce an *individual* prediction branch to estimate the quality of localization, which facilitates the classification to improve detection performance. This paper delves into the *representations* of the above three fundamental elements: quality estimation, classification and localization. Three problems are discovered in existing practices, including (1) the inconsistent usage of the quality estimation and classification between training and inference, (2) the inflexible Dirac delta distribution for localization, and (3) the deficient and implicit guidance for accurate quality estimation. To address these problems, we design new representations for these elements. Specifically, we merge the quality estimation into the class prediction vector to form a joint representation, use a vector to represent arbitrary distribution of box locations, and extract discriminant feature descriptors from the distribution vector for more reliable quality estimation. The improved representations eliminate the inconsistency risk and accurately depict the flexible distribution in real data, but contain *continuous* labels, which is beyond the scope of Focal Loss. We then propose Generalized Focal Loss (GFocal) that generalizes Focal Loss from its discrete form to the *continuous* version for successful optimization. Extensive experiments demonstrate the effectiveness of our method, without sacrificing the efficiency both in training and inference. Based on GFocal, we construct a considerably fast and lightweight detector termed NanoDet under mobile settings, which is 1.8 AP higher, 2x faster and 6x smaller than scaled YoloV4-Tiny.

Index Terms—Object detection, dense object detection, representation learning, generalized focal loss, deep learning

1 Introduction

Object detection is one of the most fundamental computer vision tasks, which simultaneously predicts a bounding box with a category label for each target instance in an image. With the rapid development of Deep Learning [16], [19], the detection performance had ever been significantly improved by the two-stage R-CNN [12], [13], [15], [38] series, where the detection is conducted by first producing region proposals and then generating category and localization results based on the extracted region features. Despite the high accuracy of the two-stage approaches, they suffer in the complex pipeline and lack training/inference efficiency, limiting the wide application to end devices. Instead, one-stage (i.e., dense) object detector [25], [26], [31], [34], [35], [42], [53], which predicts pixel-level object categories and bounding boxes over a regular, *dense* sampling of object locations, has recently become increasingly popular due to its simple and effective framework. Based on dense detectors, researchers focus more on the *representation* of bounding boxes and their Localization Quality Estimation (LQE) recently, leading to an encouraging advancement [42], [47] in the field. Specifically, bounding box *representation* is modeled as a simple Dirac delta Distribution [17], [28], [42], [50], [51], which is widely used over past years. As popularized in FCOS [42], predicting an additional localization quality (e.g., IoU score [47] or Centerness score [42]) brings consistent improvements of detection accuracy, when LQE is combined (usually multiplied) with classification confidence as final scores [18], [20], [42], [47], [55] for the rank process of Non-Maximum Suppression (NMS) during inference. Despite their great success, we observe three following problems of these representations (classification, localization and LQE) in existing practices of dense detectors:

Inconsistent Usage of LQE and Classification Score Between Training and Inference. (1) In recent dense detectors, the LQE and classification score are usually trained independently but compositely utilized (e.g., multiplication) during inference [42], [47] (Fig. 1a); (2) The supervision of the LQE is currently assigned for positive samples only [18], [20], [42],

- *Xiang Li and Jian Yang are with the PCA Lab, College of Computer Science, Nankai University, Tianjin 300071, China. E-mail: xiang.li.implus@qq.com, csjyang@nankai.edu.cn.*
- *Chengqi Lv and Wenhai Wang are with Shanghai AI Laboratory, Shanghai 200041, China. E-mail: lvchengqi@pjlab.org.cn, wangwenhai362@gmail.com.*
- *Gang Li and Lingfeng Yang are with the School of Computer Science and Engineering, Nanjing University of Science and Technology, Nanjing 210094, China. E-mail: gang.li@njust.edu.cn, yanglfnjust@163.com.*

Manuscript received 25 May 2021; revised 4 Apr. 2022; accepted 25 May 2022. Date of publication 9 June 2022; date of current version 3 Feb. 2023.
This work was supported in part by the Postdoctoral Innovative Talent Support Program of China under Grants BX20200168 and 2020M681608, and in part by the National Science Fund of China under Grant U1713208.
(Corresponding author: Jian Yang.)
Recommended for acceptance by M. Maire.
Digital Object Identifier no. 10.1109/TPAMI.2022.3180392

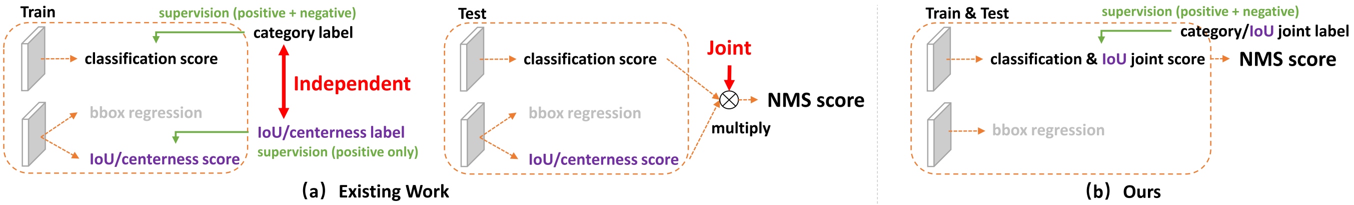


Fig. 1. *Comparisons between existing separate representation and proposed joint representation of classification and localization quality estimation.* (a): Current practices [20], [42], [47], [50], [55] for the separate usage of the quality branch (i.e., IoU or Centerness score) during training and test. (b): Our joint representation of classification and localization quality enables high consistency between training and inference.

[47], [55], which is unreliable as negatives may get chances to have uncontrollably higher quality predictions (Fig. 2a). These two factors result in a gap between training and test, and would potentially degrade the detection performance, e.g., negative instances with randomly high-quality scores could rank in front of positive examples with lower quality prediction during NMS. Specifically, based on the statistics over COCO [29] `minival` by FCOS [42], 24.54% of the dislocations are solely due to the separated unreliable localization estimates, and the resulted false positives can finally lead to ∼1 AP_{50} performance drop.

Inflexible Representation of Bounding Boxes. The widely used bounding box representation can be viewed as Dirac delta Distribution [1], [12], [15], [22], [28], [38], [42], [50] of the target box coordinates. However, it fails to consider the ambiguity and uncertainty in datasets (see the unclear boundaries of the figures in Fig. 4). Although some recent works [7], [17] model boxes as Gaussian Distributions, it is too simple to capture the real distribution of the locations of bounding boxes. In fact, the real distribution can be more arbitrary and flexible [17], without the necessity of being symmetric like the Gaussian function.

Lack of Explicit Guidance for Reliable LQE. Many previous works [21], [25], [26], [34], [35], [36], [37], [42], [44], [47], [49], [50], [54] have explored LQE. For example, the YOLO family [35], [36], [37] first adopt Objectness to describe the localization quality, which is defined as the Intersection-over-Union (IoU) between the predicted and ground-truth box. After that, IoU is further explored and proved to be effective in IoU-Net[20], IoU-aware [47], PAA [21], GFL [26] and VFNet [49]. Recently, FCOS [42] and ATSS [50] introduce Centerness, the distance degree to the object center, to suppress low-quality detection results. Generally, the aforementioned methods share a common characteristic that they are all based on *vanilla convolutional features*, e.g., features of points, borders or regions (see Figs. 5a, 5b, 5c, 5d, 5e, 5f, and 5g), to estimate the localization quality. Unfortunately, these abstract convolutional features fail to have explicit meaning to infer LQE scores, but are only implicitly supervised by localization quality signals, making it less efficient for reliable LQE.

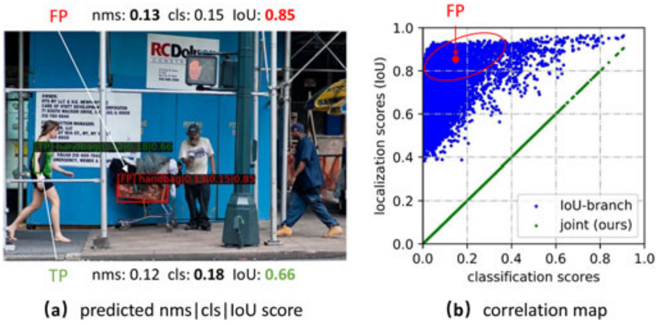


Fig. 2. *Unreliable IoU predictions of current dense detector with IoU-branch.* (a): We demonstrate the (hard) background patch with extremely high predicted quality scores (e.g., IoU score ≥ 0.85), based on the optimized IoU-branch model in Fig. 1a. The scatter diagram in (b) denotes the randomly sampled instances with their predicted scores, where the blue points clearly illustrate the weak correlation between predicted classification scores and predicted IoU scores for separate representations. The part in red circle contains many possible False Positives (FP) with large localization quality predictions, which may potentially rank in front of True Positives (TP) and impair the performance (see the detection cases in (a)). Instead, our joint representation (green points) forces them to be equal and thus alleviates such risks.

To address the above problems, we design new representations for object classification, bounding box and its localization quality as follows:

For object classification representation, we propose to merge it with the LQE score into a single and unified representation: a classification vector where its value at the ground-truth category index refers to its corresponding localization quality (typically the IoU score between the predicted box and the corresponding ground-truth box in this paper). In this way, we unify classification score and IoU score into a joint and single variable (denoted as “Classification-IoU Joint Representation”), which can be trained in an end-to-end fashion, whilst directly utilized during inference (Fig. 1b). As a result, it eliminates the training-test inconsistency (Fig. 1b) and enables the strongest correlation (Fig. 2b) between localization quality and classification. Further, the negatives will be supervised with 0 quality scores, thereby the overall quality predictions become more confidential and reliable. It is especially beneficial for dense object detectors as they rank all candidates regularly sampled across an entire image.

For bounding box representation, we propose to represent the arbitrary distribution (denoted as “General Distribution” in this paper) of box locations by directly learning the discretized probability distribution over its continuous space, without introducing any other stronger priors (e.g., Gaussian [7], [17]). The learned arbitrary distribution provides the flexibility in modelling the complexity of the essence of real data, which can not only predict precise bounding box regression (Table 6), but also reflect the informative underlying uncertainty estimations (Figs. 11 and 12).

For localization quality representation, different from previous works, we explore a brand new perspective to conduct LQE – by directly utilizing *the statistics of bounding box distributions*, instead of using the *vanilla convolutional features* (see Fig. 5). According to our observations, the statistic of the General Distribution tends to have a strong correlation with its real localization quality, as illustrated in Fig. 3b. More specifically in

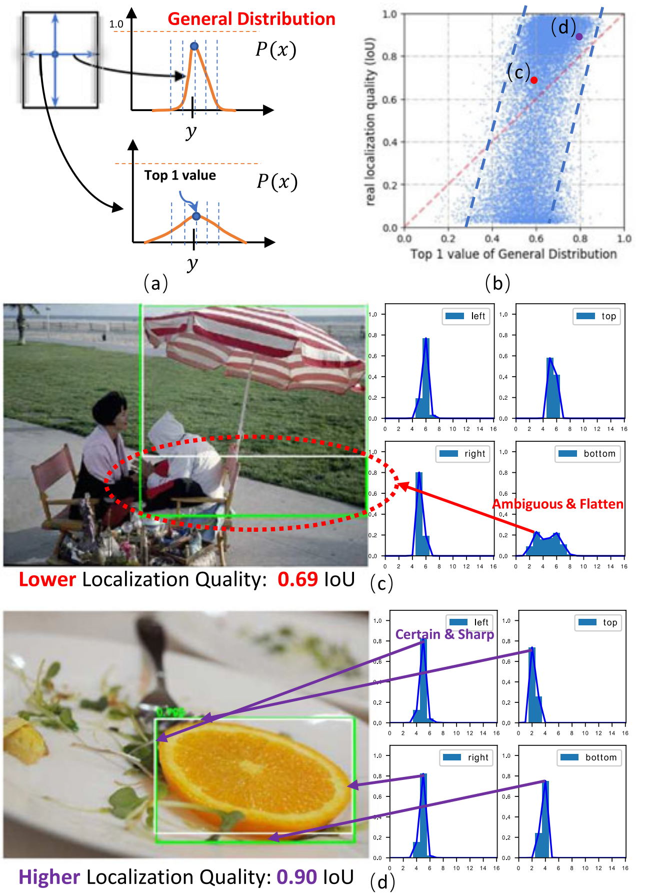


Fig. 3. *Motivation of utilizing the highly relevant statistics of learned bounding box distributions to guide the better generation of its estimated localization quality.* (a): The illustration of General Distribution to represent bounding boxes, which models the probability distribution of the predicted edges in a form of discrete probability vector. (b): The scatter diagram of the relation between Top-1 (mean of four sides) value of General Distribution of predicted boxes and their real localization quality (IoU between the prediction and ground-truth), calculated over all validation images on COCO [29] dataset. (c) and (d): Two specific examples from (b), where the sharp distribution usually corresponds to higher quality, whilst the flat one stands for lower quality usually. Green: predicted bounding boxes; White: ground-truth bounding boxes.

Figs. 3c and 3d, the shape (flatness) of bounding box distribution can clearly reflect the localization quality of the predicted results: the sharper the distribution, the more accurate the predicted bounding box, and vice versa. Consequently, it can potentially be easier and very efficient to conduct better LQE by the guidance of the distribution information, as the input (distribution statistics of bounding boxes) and the output (LQE scores) are highly correlated potentially.

The improved representations then pose challenges for optimization. Traditionally for dense detectors, the classification branch is optimized with Focal Loss [28] (FL). FL can successfully handles the class imbalance problem via reshaping the standard cross entropy loss. However, for the case of the proposed Classification-IoU Joint Representation, in addition to the imbalance risk that still exists, we face a new problem with continuous IoU label (0∼1) as supervisions, as the original FL only supports discrete $\{1, 0\}$ category label currently. We successfully solve the problem by extending FL from $\{1, 0\}$ discrete version to its continuous variant, termed Generalized Focal Loss (GFocal). Different from FL, GFocal considers a much general case in which the globally optimized solution is able to target at any desired continuous value, rather than the discrete ones. In addition, we further apply a simplified format of GFocal to facilitate the learning of arbitrary bounding box distributions. More specifically in this paper, GFocal can be specialized into Quality Focal Loss (QFL) and Distribution Focal Loss (DFL), for optimizing the improved representations: QFL focuses on a sparse set of hard examples and simultaneously produces their *continuous* 0∼1 quality estimations on the corresponding category; DFL makes the network to rapidly focus on learning the probabilities of values around the *continuous* locations of target bounding boxes, under an arbitrary and flexible distribution.

In summary, the contributions or advantages of the proposed GFocal are as follows:

- It first bridges the gap between training and test when one-stage detectors are facilitated with additional quality estimation, leading to a simpler, joint and effective representation of both classification and localization quality. The effectiveness of the joint representation is further validated in the following works, including both 2D [49] and 3D (point cloud) [11] object detection;
- It well models the flexible underlying distribution for bounding boxes, which provides more informative and accurate box locations. Furthermore, the flexible distribution for bounding box regression also offers the possibilities in distilling rich localization information for object detection, as already evidenced by [52];

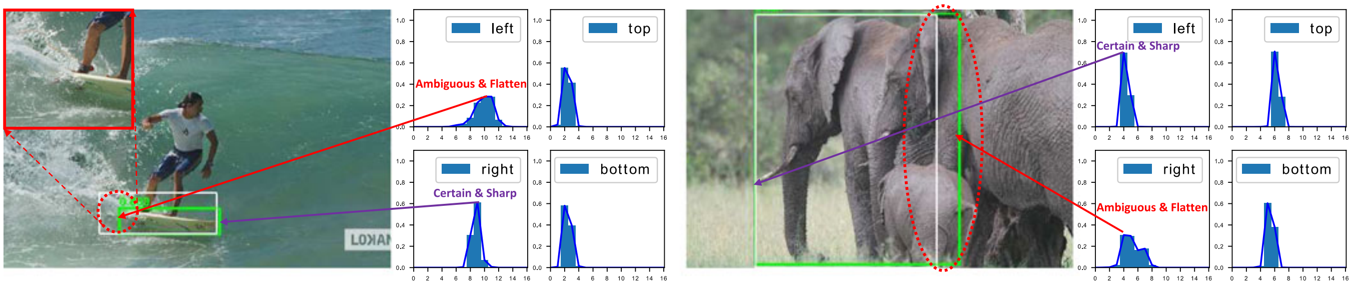


Fig. 4. *Ambiguity in bounding boxes.* Due to occlusion, shadow, blur, etc., the boundaries of many objects are not clear enough, so that the ground-truth labels (white boxes) are sometimes not credible and Dirac delta Distribution is limited to indicate such issues. Instead, the proposed learned representation of General Distribution for bounding boxes can reflect the underlying information by its shape, where a flatten distribution denotes the unclear and ambiguous boundaries (see red circles) and a sharp one stands for the clear cases. The predicted boxes by our model are marked green.

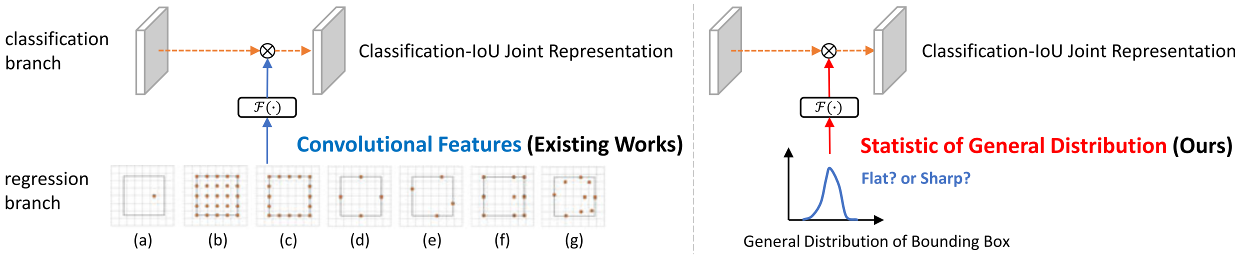

classification branch
Classification-IoU Joint Representation
$\mathcal{F}(\cdot)$
Convolutional Features (Existing Works)
regression branch
(a)
(b)
(c)
(d)
(e)
(f)
(g)
Classification-IoU Joint Representation
$\mathcal{F}(\cdot)$
Statistic of General Distribution (Ours)
Flat? or Sharp?
General Distribution of Bounding Box

Fig. 5. *Comparisons of guidance for predicting localization quality between existing works (left) and ours (right).* Existing works focus on different *spatial locations* of convolutional features, including (a): *point* [21], [26], [35], [36], [37], [42], [47], [50], (b): *region* [20], (c): *border* [34] *dense* points, (d): *border* [34] *middle* points, (e): *border* [34] *extreme* points, (f): *regular* sampling points [49], and (g): *deformable* sampling points [6], [9]. In contrast, we use the statistic of learned box distribution to produce reliable localization quality.

- It first leverages the explicit statistics of bounding box distributions to conduct reliable localization quality estimation in an end-to-end dense object detection framework. Given its improved accuracy and very simple structure, we believe this idea is not restricted in object detection but can potentially be extended into more generic regression tasks, especially under the demand of uncertainty estimation [4], [41];
- It is considerably efficient and lightweight in practice. The improved representations are universal and can also be easily applied into most dense object detectors with a consistent gain of 1$\sim$2 AP, and without loss of training/inference speed.

2 Related Work

Representation of Bounding Boxes. Dirac delta Distribution [1], [12], [15], [22], [28], [38], [42], [50] governs the representation of bounding boxes over past years. Specifically, most object detectors leverage box regression branches to predict offsets of four coordinates that cover entire object targets, which can be regarded as recovering the deviation along the X-axis of Dirac delta Distribution directly. It can be optimized by L_2 loss [13], $\mathit{Smooth}\ L_1$ loss [38], or IoU/GIoU losses [39].

Recently, Gaussian Distribution [7], [17] is adopted to learn the box regression by predicting the target coordinates and localization variance simultaneously. Its optimization involves Kullback-Leibler (KL) [14] loss, where the model seeks to estimate better variance, driven by data patterns.

Unfortunately, existing representations are either too rigid or too simplified, which can not reflect the complex underlying distribution in real data. In this paper, we further relax the assumption and directly learn the more arbitrary, flexible General Distribution of bounding boxes, whilst being more informative and accurate.

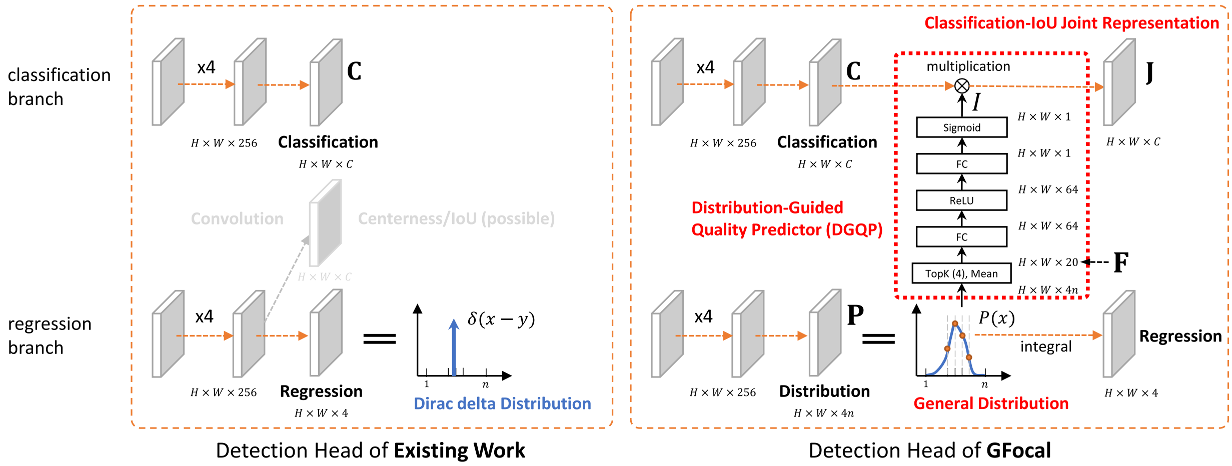
It is worth mentioning that for easier representation of neural networks, the implementation of our proposed General Distribution needs to discretize the continuous domain into a list of discrete bins, which is close to the practices of several recent works [33], [45]. However, our method differs from the related works in at least 2 aspects: (1) While other works heuristically adopt the range discretization, our implementation is deduced from continuous integral formula theoretically; (2) Unlike other works which take multiple steps to recover the regression target, we let the integral of discrete distribution become a part of neural network, thus making it end-to-end trainable and compatible for IoU/GIoU loss [39].

Representation of LQE. Early popular object detectors [1], [12], [15], [38] simply treat the classification confidence as the formulation of LQE score, but there is an obvious inconsistency between them, which inevitably degrades the detection performance. To alleviate this problem, AutoAssign [54] and BorderDet [34] employ additional localization features to rescore the classification confidence, but they still lack an explicit definition of LQE. SABL [45] introduces boundary buckets for coarse localization, and utilizes the averaged bucketing confidence as a formulation of LQE.

Existing practices like Fitness NMS [43], IoU-Net [20], MS R-CNN [18], FCOS [42], ATSS [50], PAA [21] and IoU-aware [47] utilize a separate branch to perform LQE in a form of IoU or Centerness (the distance degree to the center of the object) score. As mentioned in Introduction, this separate formulation causes the inconsistency between training and test as well as unreliable quality predictions. Instead of introducing an additional branch, PISA [3] and IoU-balance [46] assign different weights in the classification loss based on their localization qualities, aiming at enhancing the correlation between the classification score and localization accuracy. However, the weight strategy is of implicit and limited benefits since it does not change the optimum of the loss objectives for classification.

Guidance of LQE. As shown in the left part of Fig. 5, previous works directly use convolutional features as guidance for LQE, which only differ in the way of spatial sampling. Most existing methods [21], [26], [35], [36], [37], [42], [47], [50] adopt the point features (see Fig. 5a) to produce LQE scores for high efficiency. IoU-Net [20] predicts IoU based on the region features as shown in Fig. 5b. BorderDet [34] designs three types of border-sensitive features (see Figs. 5c, 5d, and 5e) to facilitate LQE. Similar with BorderDet, a star-shaped sampling manner (see Fig. 5f) is designed in VFNet [49]. Alternatively, HSD [2] and RepPoints [6], [48] focus on features with learned locations (see Fig. 5g) via the deformable convolution [9], [56].

The aforementioned methods mainly focus on extracting discriminating convolutional features with various spatial aspects for better LQE. Different from previous methods, our proposed methodology is designed in an artful perspective: predicting LQE scores by its directly correlated variables—the statistics of bounding box distributions (see the right part of Fig. 5). As later demonstrated in Table 9, compared with convolutional features shown in Figs. 5a, 5b, 5c, 5d, 5e, 5f, and 5g, the statistics of bounding box distributions achieve an impressive efficiency and a high accuracy simultaneously.



Detection Head of **Existing Work**

Detection Head of **GFocal**

Fig. 6. *The comparisons between conventional methods and our proposed GFocal on detection head.* We propose improved and efficient representations including Classification-IoU Joint Representation, General Distribution and Distribution-Guided Quality Predictor for dense object detectors, which are considerably lightweight and cost-free in practice.

3 Proposed Method

3.1 Improved Representations

The overall improved representations are illustrated in Fig. 6. We elaborate the details one by one as follows.

Classification → *Classification-IoU Joint Representation:* To solve the aforementioned inconsistency problem between training and test phases, we present a joint representation $\mathbf{J} = [J_1, J_2, \dots, J_m]$ (m indicates the total number of categories) of localization quality (i.e., IoU score) and classification score (termed "Classification-IoU Joint Representation"), where its supervision softens the standard one-hot category label and leads to a possible float vector $\mathbf{y} = [y_1, y_2, \dots, y_m]$. Given an object category label $c \in \{1, 2, \dots, m\}$, $\mathbf{y}$ satisfies

$$y_i = \begin{cases} \mathrm{IoU}(B_{pred}, B_{gt}), & \text{if } i = c; \\ 0, & \text{otherwise}, \end{cases} \tag{1}$$

where $\mathrm{IoU}(B_{pred}, B_{gt})$ denotes the IoU between the predict bounding box B_{pred} and the ground truth B_{gt}. Specifically, $y = 0$ denotes the negative samples with 0 quality score, and $0 < y \leq 1$ stands for the positive samples with target IoU score y. Following [28], [42], we adopt the multiple binary classification with sigmoid operators $\sigma(\cdot)$ for multi-class implementation. For simplicity, the output of sigmoid is marked as σ. Note that $\mathbf{J}$ is supervised by Quality Focal loss (QFL) (as later explained in next section) during training, and used directly as NMS score in inference, which substantially reduces the inconsistency of LQE and object classification between training and inference.

Dirac delta → *General Distribution Representation*.Following [42], [50], we adopt the relative offsets from the location to the four sides of a bounding box as the regression targets. Conventional operations of bounding box regression model the regressed label y as Dirac delta Distribution $\delta(x-y)$, where it satisfies $\int_{-\infty}^{+\infty} \delta(x-y)\mathrm{d}x = 1$ and is usually implemented through fully connected layers. More formally, the integral form to recover y is as follows:

$$y = \int_{-\infty}^{+\infty} \delta(x-y)x\mathrm{d}x. \tag{2}$$

According to the analysis in Introduction, instead of the Dirac delta [1], [15], [38], [42], [50] or Gaussian [7], [17] assumptions, we propose to directly learn the underlying General Distribution $P(x)$ without introducing any other priors. Given the range of label y with minimum y_0 and maximum y_n ($y_0 \leq y \leq y_n, n \in \mathbb{N}^+$), we can have the estimated value $\hat{y}$ from the model ($\hat{y}$ also meets $y_0 \leq \hat{y} \leq y_n$) by simply extending Eq. (2)

$$\hat{y} = \int_{-\infty}^{+\infty} P(x)x\mathrm{d}x = \int_{y_0}^{y_n} P(x)x\mathrm{d}x. \tag{3}$$

To be consistent with convolutional neural networks, we convert the integral over the continuous domain into a discrete representation, via discretizing the range $[y_0, y_n]$ into a set $\{y_0, y_1, \dots, y_i, y_{i+1}, \dots, y_{n-1}, y_n\}$ with even intervals Δ, $\Delta = y_{i+1} - y_i, \forall i \in [0, n-1]$ (we use $\Delta = 1$ for simplicity in later experiments). Consequently, given the discrete distribution property $\sum_{i=0}^{n} P(y_i) = 1$, the estimated regression value $\hat{y}$ can be presented as

$$\hat{y} = \sum_{i=0}^{n} P(y_i)y_i. \tag{4}$$

As a result, $P(x)$ can be easily implemented through a softmax $\mathcal{S}(\cdot)$ layer consisting of $n+1$ units. Note that $\hat{y}$ can be trained in an end-to-end fashion with any traditional loss objective like SmoothL1 [12], IoU Loss [43] or GIoU Loss [39]. To facilitate its efficient optimization, we further introduce the Distribution Focal Loss (DFL) as later explained in next section.

Convolution $\rightarrow \int$ *Distribution-Guided Quality Predictor*. Inspired by the possible strong correlation between the distribution statistics and LQE scores, we propose a very lightweight sub-network with only dozens of (e.g., 64) hidden units, on top of these distribution statistics to produce reliable LQE scores, instead of implicit convolution features. We term this lightweight sub-network as Distribution-Guided Quality Predictor (DGQP). It delivers the statistics of the learned General Distribution $\mathbf{P}$ into a tiny sub-network (see red dotted frame in Fig. 6) to obtain the predicted

IoU scalar I, which helps to generate high-quality Classification-IoU Joint Representation (Eq. (7)). For convenience, we mark the left, right, top and bottom sides as $\{l, r, t, b\}$, and define the discrete probabilities of the w side as $\mathbf{P}^w = [P^w(y_0), P^w(y_1), \ldots, P^w(y_n)]$, where $w \in \{l, r, t, b\}$.

As illustrated in Fig. 3, the flatness of the learned distribution is highly related to the quality of the final detected bounding box, and some relevant statistics can be used to reflect the flatness of the General Distribution. As a result, such statistical features have a very strong correlation with the localization quality, which will ease the training difficulty and improves the quality of estimation. Practically, we recommand to choose the Top-k values along with their mean value of each distribution vector $\mathbf{P}^w$, and concatenate them as the basic statistical feature $\mathbf{F} \in \mathbb{R}^{4(k+1)}$

$$\mathbf{F} = \mathrm{Concat}(\{\mathrm{Topkm}(\mathbf{P}^w) \mid w \in \{l, r, t, b\}\}), \tag{5}$$

where $\mathrm{Topkm}(\cdot)$ denotes the joint operation of calculating Top-k values and their mean value. $\mathrm{Concat}(\cdot)$ means the channel concatenation. Selecting Top-k values and their mean value as the input statistics have two benefits:

- Since the sum of $\mathbf{P}^w$ is fixed (i.e., $\sum_{i=0}^{n} P^w(y_i) = 1$), Top-$k$ values along with their mean value can basically reflect the flatness of the distribution: the larger, the sharper; the smaller, the flatter;
- Top-k and mean values can make the statistical feature insensitive to its relative offsets over the distribution domain (see Fig. 8), resulting in a robust representation which is not affected by object scales.

Given the statistical feature $\mathbf{F}$ of General Distribution as input, we design a very tiny sub-network $\mathcal{F}(\cdot)$ to predict the final IoU quality estimation. The sub-network has only two Fully-Connected (FC) layers, which are followed by ReLU [23] and sigmoid, respectively. Consequently, the IoU scalar $I \in [0, 1]$ can be calculated as

$$I = \mathcal{F}(\mathbf{F}) = \sigma(\mathbf{W}_2 \delta(\mathbf{W}_1 \mathbf{F})), \tag{6}$$

where δ and σ refer to the ReLU and sigmoid, respectively. $\mathbf{W}_1 \in \mathbb{R}^{d \times 4(k+1)}$ and $\mathbf{W}_2 \in \mathbb{R}^{1 \times d}$. k denotes the Top-k parameter and d is the channel dimension of the hidden layer (k = 4, d = 64 is a typical setting in our experiment). To facilitate the generation of Classification-IoU Joint Representation, we decompose $\mathbf{J}$ explicitly by leveraging information from both classification ($\mathbf{C}$) and regression (I) branches

$$\mathbf{J} = \mathbf{C} \times I, \tag{7}$$

where $\mathbf{C} = [C_1, C_2, \ldots, C_m], C_i \in [0, 1]$ denotes the original Classification Representation of total m categories.

3.2 Generalized Focal Loss

In this section, we first review the original Focal Loss [28] (FL) for learning dense classification scores of one-stage detectors. Next, based on the improved representations, i.e., Classification-IoU Joint Representation and General Distribution, we propose Quality Focal Loss (QFL) and Distribution Focal Loss (DFL) for their effective optimizations, respectively. Finally, we summarize the formulations of QFL and DFL into a unified perspective termed Generalized Focal Loss (GFocal), as a flexible extension of FL, to facilitate further promotion and general understanding in the future.

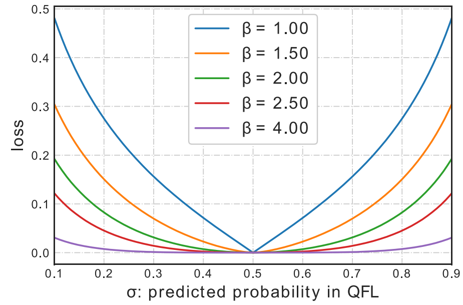


Fig. 7. *The illustration of QFL under quality label* $y = 0.5$. The gradients of values near ground-truth label tend to be small.

Focal Loss (FL). The original FL [28] is proposed to address the one-stage object detection scenario where an extreme imbalance between foreground and background classes often exists during training. A typical form of FL is as follows (we ignore α_t in original paper [28] for simplicity)

$$\mathbf{FL}(p) = -(1 - p_t)^{\gamma} \log(p_t), p_t = \begin{cases} p, & \text{when } y = 1 \\ 1 - p, & \text{when } y = 0 \end{cases} \tag{8}$$

where $y \in \{1, 0\}$ specifies the ground-truth class and $p \in [0, 1]$ denotes the estimated probability for the class with label $y = 1$. γ is the tunable focusing parameter. Specifically, FL consists of a standard cross entropy part $-\log(p_t)$ and a dynamically scaling factor part $(1 - p_t)^{\gamma}$, where the scaling factor $(1 - p_t)^{\gamma}$ automatically down-weights the contribution of easy examples during training and rapidly focuses the model on hard examples.

Quality Focal Loss (QFL). Since the proposed Classification-IoU Joint Representation requires dense supervisions over an entire image and the class imbalance problem still occurs, the idea of FL must be inherited. However, the current form of FL only supports $\{1, 0\}$ discrete labels, but our new label $\mathbf{y}$ contains decimals. Therefore, we propose to extend the two parts of FL for enabling the successful training under the case of Classification-IoU Joint Representation:

(1) The cross entropy part $-\log(p_t)$ is expanded into its complete version $-((1 - y)\log(1 - \sigma) + y\log(\sigma))$;

(2) The scaling factor part $(1 - p_t)^{\gamma}$ is generalized into the absolute distance between the estimation σ and its continuous label y, i.e., $|y - \sigma|^{\beta}$ ($\beta \geq 0$), here $|\cdot|$ guarantees the non-negativity.

Subsequently, we combine the above two extended parts to formulate the complete loss objective, which is termed as Quality Focal Loss (QFL)

$$\mathbf{QFL}(\sigma) = -\left|y - \sigma\right|^{\beta}((1 - y)\log(1 - \sigma) + y\log(\sigma)). \tag{9}$$

Note that $\sigma = y$ is the global minimum solution of QFL. QFL is visualized for several values of β in Fig. 7 under quality label $y = 0.5$. Similar to FL, the term $\left|y - \sigma\right|^{\beta}$ of QFL behaves as a modulating factor: when the quality estimation of an example is inaccurate and deviated away from label y, the modulating factor is relatively large, thus it pays more

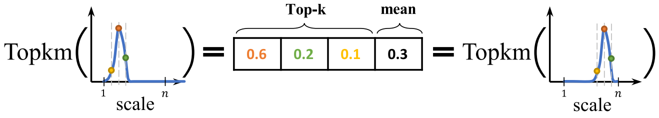


Fig. 8. $\mathrm{Topkm}(\cdot)$ *feature.* It is robust to object scales.

attention to learning this hard example. As the quality estimation becomes accurate, i.e., $\sigma \rightarrow y$, the factor goes to 0 and the loss for well-estimated examples is down-weighted, in which the parameter β controls the down-weighting rate smoothly ($\beta = 2$ works best for QFL in our experiments). Like FL, the modulating factor is sensitive to the quality of manual annotations as it may potentially force the network to focus on learning the mislabeled, hard samples.

Distribution Focal Loss (DFL). Although $\hat{y} = \sum_{i=0}^{n} P(y_i)y_i$ can already be trained in an end-to-end fashion with the traditional loss objective, e.g., GIoU Loss [39], there are infinite combinations of values for $P(x)$ that can make the final integral result being y, which may reduce the learning efficiency. Furthermore, as it is often the case that the most appropriate underlying location, if exists, would not be far away from the coarse manual label, it motivates us to optimize the shape of $P(x)$ via explicitly encouraging the high probabilities of values that are close to the target y. Therefore, we introduce the Distribution Focal Loss (DFL) which forces the network to rapidly focus on the values near label y, by explicitly enlarging the probabilities of y_i and y_{i+1} (the nearest two to y, $y_i \leq y \leq y_{i+1}$). As the learning of bounding boxes are only for positive samples without the risk of class imbalance problem, we simply apply the complete cross entropy part in QFL for the definition of DFL

$$\mathbf{DFL}(P_i, P_{i+1}) = -((y_{i+1} - y)\log(P_i) + (y - y_i)\log(P_{i+1})), \tag{10}$$

where $P(y_i)$ is denoted as P_i for simplicity. Intuitively, DFL aims to focus on enlarging the probabilities of the values around target y (i.e., y_i and y_{i+1}). The global minimum solution of DFL, i.e, $P_i = \frac{y_{i+1}-y}{y_{i+1}-y_i}, P_{i+1} = \frac{y-y_i}{y_{i+1}-y_i}$, can guarantee the estimated regression target $\hat{y}$ infinitely close to the corresponding label y, i.e., $\hat{y} = \sum_{j=0}^{n} P(y_j)y_j = P_i y_i + P_{i+1}y_{i+1} = \frac{y_{i+1}-y}{y_{i+1}-y_i} y_i + \frac{y-y_i}{y_{i+1}-y_i} y_{i+1} = y$, which also ensures its correctness as a loss function.

Generalized Focal Loss (GFocal). Note that QFL and DFL can be unified into a general form, which is called the Generalized Focal Loss (GFocal) in the paper. Assume that a model estimates probabilities for two variables $y_l, y_r (y_l < y_r)$ as p_{y_l}, p_{y_r} ($p_{y_l} \geq 0, p_{y_r} \geq 0, p_{y_l} + p_{y_r} = 1$), with a final prediction of their linear combination being $\hat{y} = y_l p_{y_l} + y_r p_{y_r} (y_l \leq \hat{y} \leq y_r)$. The corresponding continuous label y for the prediction $\hat{y}$ also satisfies $y_l \leq y \leq y_r$. Taking the absolute distance $|y - \hat{y}|^{\beta}$ ($\beta \geq 0$) as modulating factor, the specific formulation of GFocal can be written as

$$\begin{aligned}\mathbf{GFocal}(p_{y_l}, p_{y_r}) = -\left|y - (y_l p_{y_l} + y_r p_{y_r})\right|^{\beta}\big((y_r - y)\log(p_{y_l})\\ + (y - y_l)\log(p_{y_r})\big).\end{aligned} \tag{11}$$

It is easy to deduce that FL [28] and the proposed QFL and DFL are all *special cases* of GFocal:

- **FL**: Letting $\beta = \gamma, y_l = 0, y_r = 1, p_{y_r} = p, p_{y_l} = 1 - p$ and $y \in \{1, 0\}$ in GFocal

$$\begin{aligned}\mathbf{FL}(p) &= \mathbf{GFocal}(1-p, p)\\ &= -\left|y - p\right|^{\gamma}\big((1-y)\log(1-p) + y\log(p)\big), y \in \{1, 0\}\\ &= -(1-p_t)^{\gamma}\log(p_t), p_t = \begin{cases} p, & \text{when } y = 1\\ 1-p, & \text{when } y = 0\end{cases}\end{aligned} \tag{12}$$

- **QFL**: Having $y_l = 0, y_r = 1, p_{y_r} = \sigma$ and $p_{y_l} = 1 - \sigma$ in GFocal

$$\begin{aligned}\mathbf{QFL}(\sigma) &= \mathbf{GFocal}(1-\sigma, \sigma)\\ &= -\left|y - \sigma\right|^{\beta}((1-y)\log(1-\sigma) + y\log(\sigma)).\end{aligned} \tag{13}$$

- **DFL**: By substituting $\beta = 0, y_l = y_i, y_r = y_{i+1}, p_{y_l} = P(y_l) = P(y_i) = P_i, p_{y_r} = P(y_r) = P(y_{i+1}) = P_{i+1}$ in GFocal

$$\begin{aligned}&\mathbf{DFL}(P_i, P_{i+1}) = \mathbf{GFocal}(P_i, P_{i+1})\\ &= -((y_{i+1} - y)\log(P_i) + (y - y_i)\log(P_{i+1})).\end{aligned} \tag{14}$$

3.3 Training Dense Detectors With GFocal

Note that GFocal can be applied to any one-stage detectors. The modified detectors differ from the original detectors in two aspects. First, during inference, we directly feed the Classification-IoU Joint Representation as NMS scores without the need of multiplying any *individual* quality prediction if there exists (e.g., Centerness as in FCOS [42] and ATSS [50]). Second, the last layer of the regression branch for predicting each location of bounding boxes now has $n + 1$ outputs instead of 1 output, where the final position is obtained by a simple discrete integral over these $n + 1$ units, which brings *negligible* extra computing cost as later shown in Table 12.

We define training loss $\mathcal{L}$ with GFocal

$$\mathcal{L} = \frac{1}{N_{pos}}\sum_{z}\mathcal{L}_{\mathcal{Q}} + \frac{1}{N_{pos}}\sum_{z}\mathbf{1}_{\{c_z^* > 0\}}(\lambda_0\mathcal{L}_{\mathcal{B}} + \lambda_1\mathcal{L}_{\mathcal{D}}), \tag{15}$$

where $\mathcal{L}_{\mathcal{Q}}$ is QFL and $\mathcal{L}_{\mathcal{D}}$ is DFL. Typically, $\mathcal{L}_{\mathcal{B}}$ denotes the GIoU Loss as in [42], [50]. N_{pos} stands for the number of positive samples. λ_0 (typically 2 as default, similarly in [5]) and λ_1 (practically $\frac{1}{4}$, averaged over four directions) are the balance weights for $\mathcal{L}_{\mathcal{Q}}$ and $\mathcal{L}_{\mathcal{D}}$, respectively. The summation is calculated over all locations z on the pyramid feature maps [27]. $\mathbf{1}_{\{c_z^* > 0\}}$ is the indicator function, being 1 if $c_z^* > 0$ and 0 otherwise. Following the common practices in the official codes [5], [24], [42], [50], we also utilize the quality scores to weight $\mathcal{L}_{\mathcal{B}}$ and $\mathcal{L}_{\mathcal{D}}$ during training.

3.4 Analyses

Discussion About the Distributions. We list several key comparisons about these distributions in Table 1. It can be observed that the loss objective of the Gaussian assumption is actually a dynamically weighted L2 Loss, where its training weight is related to the predicted variance σ. It is somehow similar to that of Dirac delta (standard L2 Loss) when optimized at the edge level. Moreover, it is not clear how to integrate the Gaussian assumption into the IoU-based Loss formulations, since it

TABLE 1
Comparisons Between Three Distributions

Distribution Type	Probability Density Function	Inference Target	Uncertainty Modelling	Supported Optimization Level	
				edge	box
Dirac delta Distribution [42], [50]	$\delta(x-y)$	x	×	✓, $\frac{(x-y)^2}{2}$	✓, IoU-based Loss [39]
Gaussian Distribution [7], [17]	$N(x,\sigma^2)$	x	✓	✓, $\frac{(x-y)^2}{2\sigma^2}+\frac{1}{2}\log(\sigma^2)$	×
General Distribution **(ours)**	$P(x)$	$\int P(x)x\mathrm{d}x$	✓	✓, $\frac{\left(\int P(x)x\mathrm{d}x-y\right)^2}{2}$	✓, IoU-based Loss [39]

"edge" level denotes optimization over four respective directions, whilst "box" level means IoU-based Losses [39] that consider the bounding box as a whole. The proposed General Distribution can flexibly support uncertainty modelling, both edge and box level optimization, which are superior to other distribution types.

heavily couples the expression of the target representation with its edge level optimization objective. Therefore, it can not enjoy the benefits of the IoU-based optimization, as it is proved to be very effective in practice [39]. In contrast, our proposed General Distribution decouples the representation and loss objective, making it feasible for any type of optimizations, including both edge and box level.

Complexity. The overall architecture of GFocal is illustrated in Fig. 6. It is worth noting that the General Distribution and DGQP module are both very lightweight. First, they only bring thousands of additional parameters, which are negligible compared to the number of parameters of the entire detection model that easily exceed *10 millions*. For example, for the model with ResNet-50 [16] and FPN [27], the extra parameters of the General Distribution and DGQP module only account for ~0.04%. Second, the computational overhead of them is also very small due to its extremely light structure. As shown in Tables 12 and 13, the use of the General Distribution and DGQP module hardly reduces the training and inference speed of the original detector in practice.

4 Experiment and Analysis

Our experiments are mainly conducted on COCO benchmark [29], where `trainval35k` (115K images) is utilized for training and we use `minival` (5K images) as validation for our ablation study. The main results are reported on `test-dev` (20K images) which can be obtained from the evaluation server. For fair comparisons, all results are produced under mmdetection [5], where the default hyper-parameters are adopted.

Training Details. The ImageNet pretrained models [16] with FPN [27] are utilized as the backbones. During training, the input images are resized to keep their shorter side being 800 and their longer side less or equal to 1333. In ablation study, unless otherwise stated, the networks are trained using the Stochastic Gradient Descent (SGD) algorithm for 12 epochs (denoted as 1x schedule) with 0.9 momentum, 0.0001 weight decay and 16 batch size. The initial learning rate is set as 0.01 and decayed by 0.1 at epoch 8 and 11, respectively.

Inference Details. During inference, the input image is resized in the same way as in the training phase, and then passed through the whole network to output the predicted bounding boxes with a predicted class. Then we use the threshold 0.05 to filter out a variety of backgrounds, and output top 1000 candidate detections per feature pyramid. Finally, NMS is applied under the IoU threshold 0.6 per class to produce the final top 100 detections per image as results.

4.1 Classification-IoU Joint Representation

Comparisons to Additional Quality Branch and Quality Weighting Methods. In Table 2, we first compare the proposed joint representation with its separate or implicit counterparts, i.e., the additional quality branch and quality weighting approaches. Two alternatives for representing localization quality: IoU [20], [47] and Centerness [42], [50] are also adopted in the experiments. In general, we construct 4 variants that use separate or implicit representation, as illustrated in Fig. 9. According to the results from Table 2, we observe that the joint representations (optimized by QFL) consistently achieve better performance than all the counterparts, whilst IoU always performs better than Centerness as a measurement of localization quality. From Table 2, it can be observed that the joint representation largely improves the AP_{50} (by a gain of ~1 point), but achieves comparable performance to other best counterparts under larger IoU threshold (i.e., AP_{75}). It is reasonable that our joint representation can make *more top-ranked* instances roughly (under loose IoU constraint) close to the ground-truths by unifying the representation of classification and LQE, but the performance under strict IoU threshold (i.e., ≥ 0.75) is mainly limited by the bounding box regression branch, rather the classification one.

One-Stage Detectors With Joint Representations. Beyond FCOS and ATSS, we apply the joint representations to a

TABLE 2
Comparisons to Additional Quality Branch and Quality Weighting Methods

Method	Type	FCOS [42]			ATSS [50]		
		AP	AP_{50}	AP_{75}	AP	AP_{50}	AP_{75}
w/o quality (Baseline)		37.8	56.2	40.8	38.0	56.5	40.7
Addition Branch	ctr.	38.5	56.8	41.6	39.2	57.4	42.2
	IoU	38.7	56.7	**42.0**	39.6	57.6	**43.0**
Quality Weight	ctr.	37.9	56.7	40.7	38.2	56.2	41.0
	IoU	38.2	57.0	41.1	38.9	57.4	41.8
joint rep. **(ours)**		**39.0**	**57.8**	41.9	**39.9**	**58.5**	**43.0**

"ctr." denotes Centerness. The baselines are FCOS [42] and ATSS [50] detectors without their quality estimation branch. The proposed joint representation performs best among all variants.

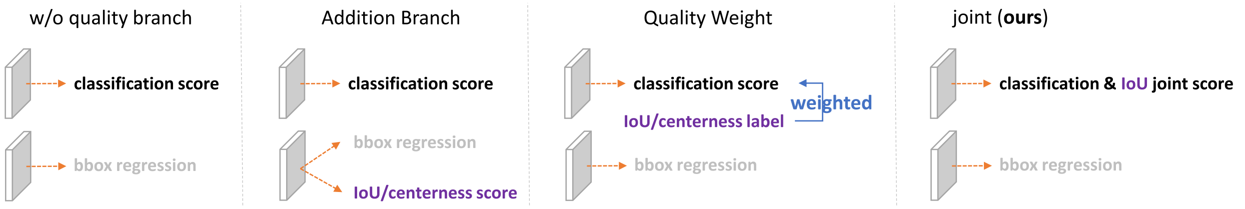


Fig. 9. *Illustrations of modified versions for separate/implicit and joint representation.* The baseline (i.e., the leftmost figure) is built upon FCOS [42]/ ATSS [50] detector without their quality prediction branch applied.

series of one-stage object detectors. Note that the proposed Classification-IoU Joint Representation does not require any additional parameter nor network modification, which is completely cost-free. From Table 3, the joint representations can consistently boost the performance of one-stage detectors by 0.6~0.8 AP gains.

Comparisons to Other Type of Classical Loss Functions. As described in Introduction, the supervisions for Classification-IoU Joint Representation contains continuous values which is beyond the scope of original Focal Loss (FL). Consequently, we propose Quality Focal Loss (QFL) to solve the problem. However, we wonder whether the classical L_1 and L_2 Loss can already fix such an issue. In Table 4, we apply different type of loss objectives to optimize the proposed Classification-IoU Joint Representation, and discover that our proposed QFL is necessary to achieve the competitive performance, as it extends from FL by seriously considering the class imbalance problem.

The Choice of Hyper-Parameter β. Table 5 shows the effect of β in QFL based on ATSS, where $\beta = 2$ is the best practical setting.

4.2 General Distribution

Comparisons to Other Prior Distributions. Based on FCOS, we apply different prior distribution for bounding box regression to conduct the experiments. For Gaussian Distribution which does not support IoU-based loss objectives, we simply adopt its applicable KL loss for each edge following [7], [17]. The results are listed in Table 6, showing that the General Distribution achieves superior results, whilst DFL can further boost its performance.

The Choice of Hyper-Parameter y_n and Δ. Based on ATSS model with Classification-IoU Joint Representation and General Distribution applied, we conduct a series of experiments to see the effect of y_n and Δ for the discrete representation of General Distribution. To quickly select a reasonable value of y_n, we first illustrate the distribution of the regression targets over all training samples on COCO `trainval35k` in Fig. 10. It is observed that most of the values locates at the range $[0, 15]$, thus we search y_n in the list of $\{12, 14, 16, 18\}$ by fixing $\Delta = 1$ in Table 7. Given $y_n = 16$ as a typical optimal choice, we further vary Δ, and discover that $\Delta = 1$ can already perform favorably good for practice.

Qualitative Results. We illustrate the learned General Distribution of our model based on ResNet-50 backbone in Figs. 11 and 12. As demonstrated in Fig. 11, we show several cases with boundary ambiguities, e.g., does the slim and almost invisible backpack strap belong to the box of the bag (left top)? In Fig. 12, more examples with clear boundaries and sharp General Distributions are shown, where our method is very confident to generate accurate bounding boxes, e.g., the bottom parts of the skiing woman. We also notice that in some cases, our models even produce more reasonable coordinates of bounding boxes than the ground-truth ones.

TABLE 3
The Effect of Joint Representations

Method	AP	AP_{50}	AP_{75}
FoveaBox [22]	36.4	**55.8**	38.8
FoveaBox [22] + joint rep. **(ours)**	**37.0**	55.7	**39.6**
RetinaNet [28]	35.6	55.5	38.1
RetinaNet [28] + joint rep. **(ours)**	**36.4**	**56.3**	**39.1**
SSD512 [31]	29.4	49.1	30.6
SSD512 [31] + joint rep. **(ours)**	**30.2**	**50.3**	**31.7**

The proposed joint representations improves popular dense object detectors by 0.6~0.8 AP without any additional computation cost.

TABLE 4
Comparisons Between Different Loss Objectives to Optimize the Joint Representation

Loss Type	FCOS [42]			ATSS [50]		
	AP	AP_{50}	AP_{75}	AP	AP_{50}	AP_{75}
L_1		NaN			NaN	
L_2	19.2	27.4	21.0	20.4	28.6	22.1
QFL **(ours)**	**39.0**	**57.8**	41.9	**39.9**	**58.5**	**43.0**

"NaN" indicates that the training is unstable due to the numerical overflow.

TABLE 5
Varying β for QFL Based on ATSS: $\beta = 2$ Performs Best

β (QFL)	AP	AP_{50}	AP_{75}
0	37.6	55.4	40.3
1	39.0	58.1	41.7
2	**39.9**	**58.5**	**43.0**
2.5	39.7	58.1	42.7
4	38.2	55.4	41.6

TABLE 6
Performances Under Different Data Representation of Bounding Box Based on FCOS: The Proposed General Distribution Supervised by DFL Improves Favorably Over The Competitive Baselines

Prior	AP	AP_{50}	AP_{75}
Dirac delta Distribution [42], [50]	38.5	56.8	41.6
Gaussian Distribution [7], [17]	38.6	56.5	41.6
General Distribution **(ours)**	38.8	56.6	42.0
General Distribution w/ DFL **(ours)**	**39.0**	**57.0**	**42.3**

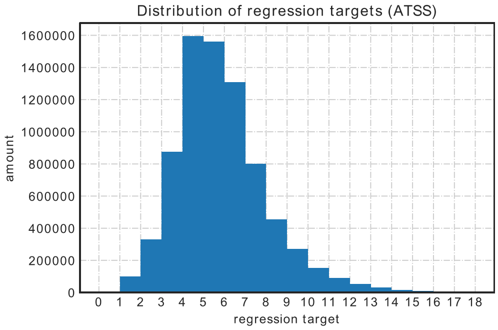


Fig. 10. *The histogram of bounding box regression targets of ATSS.* It is calculated over all training samples on COCO `trainval35k`.

TABLE 7
Performances of Various y_n and Δ Based on ATSS With GFocal

y_n	Δ	AP	AP$_{50}$	AP$_{75}$	AP$_S$	AP$_M$	AP$_L$
12	1	40.1	58.4	43.1	23.1	43.8	52.5
14		**40.2**	58.3	**43.6**	**23.3**	44.2	52.2
16		**40.2**	**58.6**	43.4	23.0	**44.3**	**53.0**
18		40.1	58.1	43.1	22.6	43.9	52.6
16	0.5	**40.2**	58.4	43.0	22.3	43.8	**53.1**
	1	**40.2**	**58.6**	**43.4**	**23.0**	**44.3**	53.0
	2	39.9	58.3	42.9	22.5	43.8	51.8
	4	39.8	58.5	42.8	22.8	43.4	52.3

The performance is robust to y_n and $\Delta = 1$ is good enough for practice.

4.3 Distribution-Guided Quality Predictor

Structure of DGQP (i.e., k, d). We examine the impact of different parameters of k, d in DGQP on the detection performance. Specifically, we report the effect of k and d by fixing one and varying another in Table 8. It is observed that $k = 4, d = 64$ steadily achieves the optimal accuracy among various combinations.

Comparisons of Guidance for Quality Prediction. To the best of our knowledge, the proposed DGQP is the first to use the statistics of learned distributions of bounding boxes for the generation of better LQE scores in the literature. Since the input (distribution statistics) and the output (LQE scores) are highly correlated potentially, we speculate that it can be more effective or efficient than ordinary convolutional guidance proposed in existing methods. Therefore, we fix the hidden layer dimension of DGQP (i.e., $d = 64$) and compare our statistical input with most existing possible types of convolutional inputs, from point (a), region (b), border (c)-(e), regular points (f), and deformable points (g), respectively (Fig. 5). Table 9 shows that our distribution statistics perform best in overall AP, also fastest in inference, compared against various convolutional features.

DGQP Improves LQE and Eases the Learning Difficulty. To assess whether DGQP is able to benefit the estimation of localization quality, we first obtain the predicted IoUs (given by four representative models with IoU as the quality estimation targets) and their corresponding real IoUs over all the positive samples on COCO `minival`. Then we calculate their Pearson Correlation Coefficient (PCC) in Table 10. It demonstrates that DGQP in GFocal indeed improves the linear correlation between the estimated IoUs and the ground-truth ones by a considerable margin (+0.26) against the one without DGQP, which eventually leads to an absolute 0.9 AP gain. Fig. 15 provides the visualization of the training losses on LQE scores, where DGQP in GFocal successfully accelerates the training process and converges to lower losses.

4.4 GFocal

Ablation Study on Each Component. In general, GFocal consists of Classification-IoU Joint Representation optimized by QFL, General Distribution optimized by DFL, and Distribution-Guided Quality Predictor (DGQP) for reliable localization quality estimation. In Table 11, we show the individual performance gain of each component based on ATSS [50], where QFL (+0.7) and DGQP (+0.9) contribute most to the overall improvement.

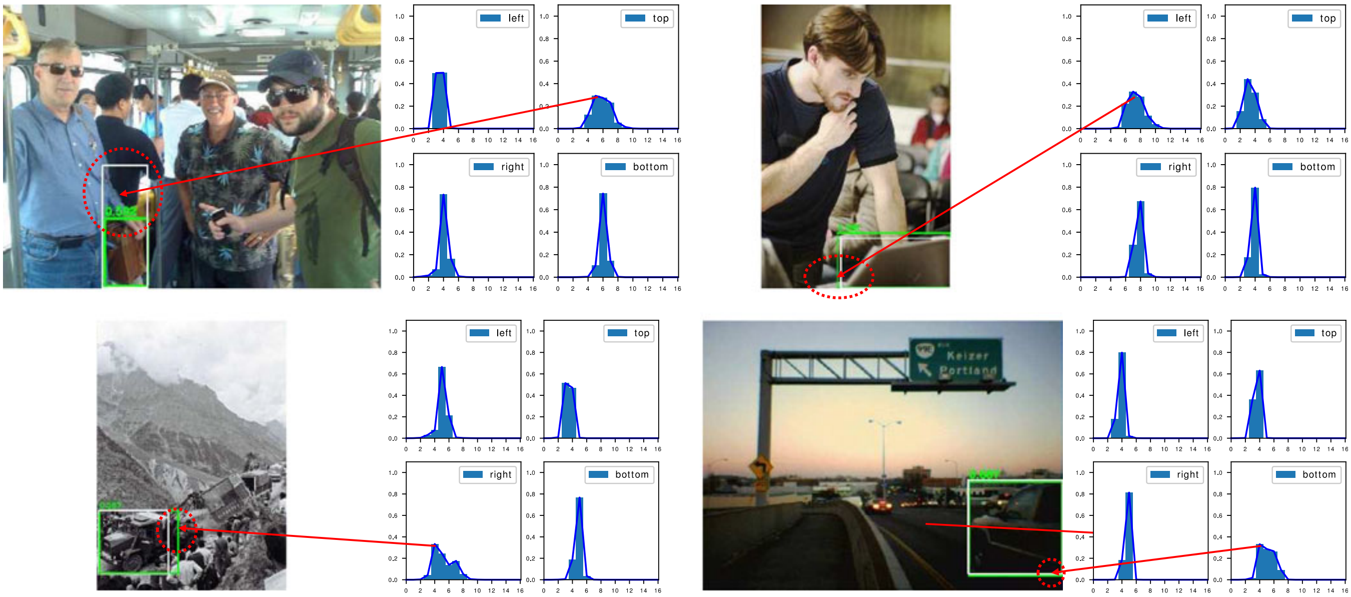


Fig. 11. *Examples with huge boundary ambiguities and uncertainties*. The learned General Distributions tend to be flatten on at least one side. Predictions are marked green in images, whilst ground-truth boxes are white.

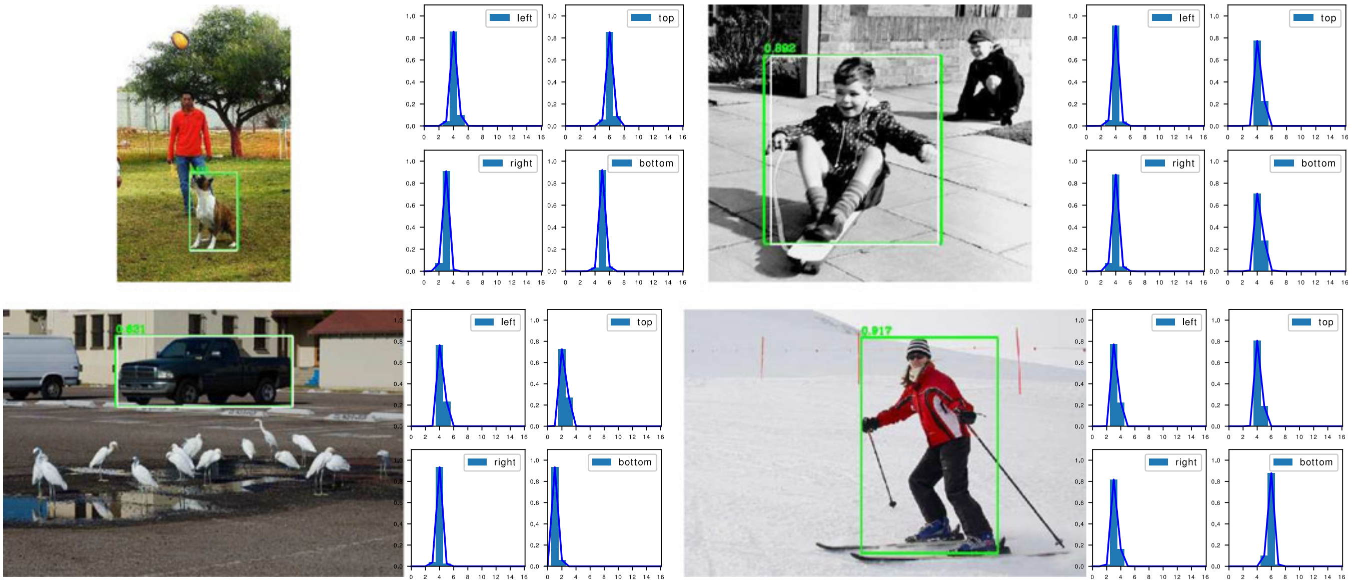


Fig. 12. *Examples with extremely clear boundaries.* The learned General Distributions are relatively sharp whilst producing very accurate box estimations. Predictions are marked green in images, whilst ground-truth boxes are white.

Compatibility for Dense Detectors. Since GFocal is very lightweight and can be adapted to various types of dense detectors, we employ it to a series of recent popular detection methods.

TABLE 8
Performances of Various k, d in DGQP Based on ATSS With GFocal

k	d	AP	AP_{50}	AP_{75}	AP_S	AP_M	AP_L
0	–	40.2	58.6	43.4	23.0	44.3	53.0
1	64	40.2	58.3	44.0	23.4	44.1	52.1
2		40.9	58.5	44.6	23.3	44.8	**53.5**
3		40.9	58.5	44.6	**24.3**	**44.9**	52.3
4		**41.1**	**58.8**	**44.9**	23.5	**44.9**	53.3
8		41.0	58.6	44.5	23.5	44.5	53.4
16		40.8	58.5	44.4	23.4	44.2	53.1
4	8	40.9	58.4	44.5	23.1	44.5	52.6
	16	40.8	58.3	44.1	23.3	44.6	52.0
	32	40.9	58.7	44.3	23.1	44.6	53.2
	64	**41.1**	**58.8**	**44.9**	**23.5**	**44.9**	**53.3**
	128	40.9	58.3	44.6	23.2	44.4	52.7
	256	40.7	58.3	44.4	23.4	44.3	52.9

$k = 0$ denotes the baseline version without the usage of DGQP.

TABLE 9
Comparisons Among Different Input Guidance by Fixing the Hidden Layer Dimension of DGQP

Input Guidance		AP	AP_{50}	AP_{75}	FPS
Baseline (ATSS [50] w/ QFL)		39.9	58.5	43.0	**19.4**
Convolutional Features	(a)	40.2	58.6	43.7	19.3
	(b)	40.5	59.0	44.0	14.0
	(c)	40.5	58.7	44.1	16.2
	(d)	40.6	59.0	44.0	18.3
	(e)	40.6	58.9	44.1	17.8
	(f)	40.7	59.0	44.1	17.9
	(g)	40.8	58.9	44.6	18.4
Distribution Statistics **(ours)**		**41.1**	58.8	44.9	**19.4**

The distribution statistics are proved to be the most efficient.

TABLE 10
Pearson Correlation Coefficients (PCC) for Representative Dense Object Detectors

Method	AP	FPS	PCC ↑
FCOS* [42]	39.1	19.4	0.624
ATSS* [50]	39.9	19.4	0.631
GFocal w/o DGQP	40.2	19.4	0.634
GFocal w/ DGQP	**41.1 (+0.9)**	19.4	**0.660 (+0.26)**

* *denotes the application of Classification-IoU Joint Representation, instead of additional Centerness branch.*

TABLE 11
Ablation Study on Each Component of GFocal

QFL	DFL	DGQP	FPS	AP	AP_{50}	AP_{75}
ATSS [50] (Baseline)			19.4	39.2	57.4	42.2
✓			19.4	39.9 (+0.7)	58.5	43.0
✓	✓		19.4	40.2 (+1.0)	58.6	43.4
✓	✓	✓	19.4	41.1 (+1.9)	58.8	44.9

"QFL" denotes the Classification-IoU Joint Representation optimized by QFL; "DFL" denotes the General Distribution optimized by DFL; "DGQP" refers to the usage of Distribution-Guided Quality Predictor.

TABLE 12
Integrating GFocal Into Various Popular Dense Object Detectors

Method	GFocal	AP	AP_{50}	AP_{75}	FPS
RetinaNet [28]		36.5	55.5	38.7	19.0
RetinaNet [28]	✓	**38.6 (+2.1)**	**56.2**	**41.7**	19.0
FoveaNet [22]		36.4	55.8	38.8	20.0
FoveaNet [22]	✓	**38.5 (+2.1)**	**56.8**	**41.6**	20.0
FCOS [42]		38.5	56.9	41.4	19.4
FCOS [42]	✓	**40.6 (+2.1)**	**58.2**	**43.9**	19.4
ATSS [50]		39.2	57.4	42.2	19.4
ATSS [50]	✓	**41.1 (+1.9)**	**58.8**	**44.9**	19.4
RepPointsV2 [6]		41.0	59.6	43.8	13.5
RepPointsV2 [6]	✓	**42.0 (+1.0)**	**60.4**	**44.8**	13.5

A consistent 1$\sim$2 AP gain is observed without loss of inference speed, which validates its efficient representation learning.

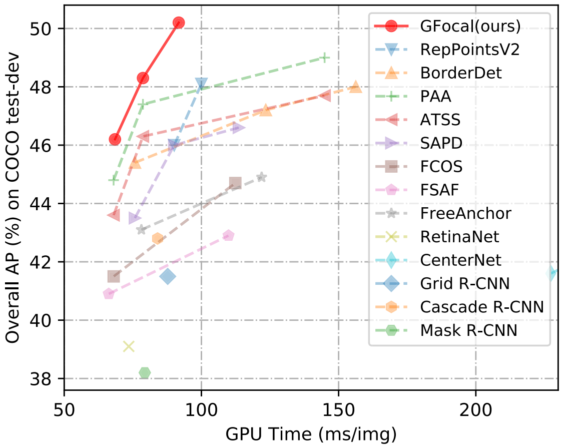


Fig. 13. *Single-model single-scale speed (ms) versus accuracy (AP) on COCO* `test-dev` among state-of-the-art approaches. GFocal achieves better speed-accuracy trade-off than its competitive counterparts.

TABLE 13
Comparisons of Training and Inference Efficiency Based on ResNet-50 Backbone

Method	AP	Training Hours ↓	Inference FPS ↑
ATSS* [50]	39.9	8.2	19.4
PAA [21]	40.4	12.5 (+52%)	19.4
RepPointsV2 [6]	41.0	14.4 (+65%)	13.5 (-30%)
BorderDet [34]	41.4	10.0 (+22%)	16.7 (-14%)
GFocal **(ours)**	41.1	8.2	19.4

*"Training Hours" is evaluated on 8 GeForce RTX 2080Ti GPUs under standard 1x schedule (12 epochs). * denotes the application of Classification-IoU Joint Representation.*

For those detectors that do not support the distributed representation of bounding boxes, we make the minimal and necessary modifications to enable it to generate distributions for each edge of a bounding box. According to the analysis in [50], we simplify the RetinaNet baseline by utilizing only one single anchor and keep its original accuracy. Based on the results in Table 12, GFocal can consistently improve 1~2 AP in popular dense detectors, without loss of inference speed.

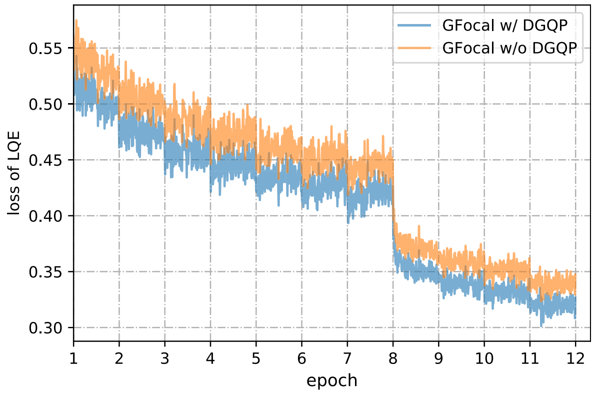


Fig. 15. *Comparisons of losses on LQE with and without DGQP.* DGQP helps to ease the learning difficulty with lower losses.

TABLE 14
Comparisons Among Popular Dense Object Detectors on CrowdHuman Dataset

Method	MR ↓	AP ↑	JI ↑
RetinaNet [28]	52.90	86.11	75.44
FCOS [42]	51.04	86.81	75.84
PAA [21]	48.89	87.56	77.29
VFNet [49]	48.54	88.20	77.81
GFocal **(ours)**	**46.77**	**88.85**	**78.06**

GFocal outperforms the competitive counterparts in all metrics.

Comparisons With Stage-of-the-Arts. In this section, we compare GFocal with state-of-the-art approaches on COCO `test-dev` in Fig. 13. Following previous works [28], [42], the multi-scale training strategy and 2x learning schedule (24 epochs) are adopted during training. For a fair comparison, the results of single-model single-scale testing for all methods using various backbones are illustrated, including the inference efficiency (ms per image). It is observed that GFocal pushes the envelope of accuracy-speed boundary to a new level.

Training/Inference efficiency. We also compare the training and inference efficiency among recent state-of-the-art dense detectors in Table 13. Note that PAA [21], RepPointsV2 [6] and

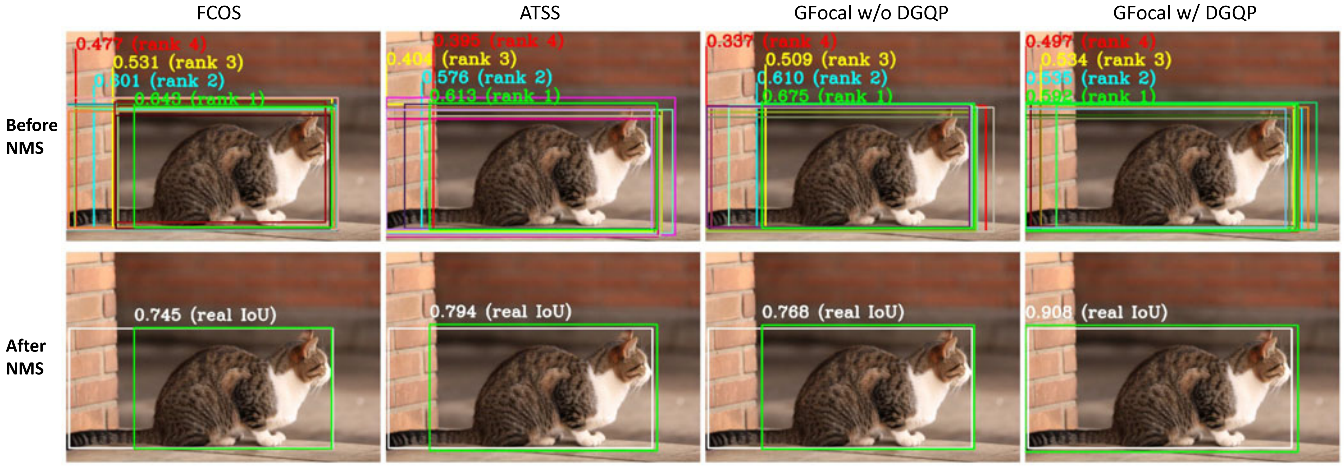


Fig. 14. *Visualization of predicted bounding boxes before and after NMS.* Their corresponding predicted LQE scores (only Top-4 scores) are plotted for a better view. For many existing approaches [42], [50], they fail to produce the highest LQE scores for the best candidates. In contrast, our GFocal (w/ DGQP) reliably assigns larger quality scores for those real high-quality ones, thus reducing the risk of mistaken suppression in NMS processing. White: ground-truth bounding boxes; Other colors: predicted bounding boxes.

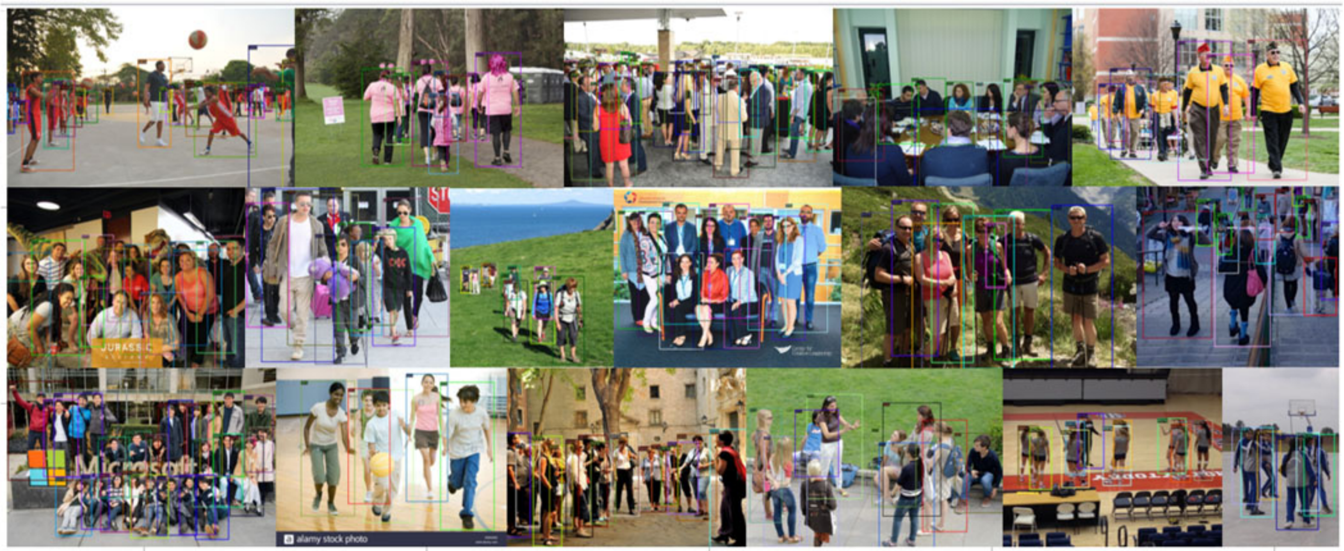


Fig. 16. *Qualitative results of GFocal on the CrowdHuman val set with the ResNet-50 backbone.*

BorderDet [34] bring an inevitable time overhead (52%, 65%, and 22% respectively) during training, and the latter two also sacrifice inference speed by 30% and 14%, respectively. In contrast, our proposed GFocal can achieve top performance (∼41 AP) while still maintaining the training and inference efficiency.

Qualitative Results. In Fig. 14, we qualitatively demonstrate the mechanism how GFocal with DGQP makes use of its more reliable IoU quality estimations to maintain accurate predictions during NMS. Unfortunately for other detectors, high-quality candidates are wrongly suppressed due to their relatively lower localization confidences, which eventually leads to a performance degradation.

4.5 GFocal on CrowdHuman

To test the generalization of GFocal on the other scenarios, we conduct experiments on highly crowded dataset CrowdHuman [40]. CrowdHuman contains 15K images for training, 4K for validation and 5K for test. Following [8], [40], the AP, log-average Miss Rate on False Positive Per Image (FPPI) in $[10^{-2}, 10^{0}]$ (MR) [10] and Jaccard Index (JI) are utilized as evaluation metrics. We compare GFocal with representative state-of-the-art dense object detectors, using ResNet-50 [16] with FPN [27] as the backbone network. The models are trained in batch size 4 (on 2 GPUs) for 20 epochs, with learning rate decay of 0.1 after epoch 16. For fair comparisons, all methods are implemented under the same mmdetection [5] framework. During test, the images are resized so that the short edge is 800 pixels while the long edge is smaller than 1400 pixels.

The quantitative results are listed in Table 14. It is observed that GFocal outperforms the representative dense object detectors in all of the three metrics. For MR which is more suitable for evaluation in this task, GFocal achieves a significant improvement of 4.3 points compared to FCOS baseline. The superior performance of GFocal demonstrates that the proposed improved representations are not restricted to a typical dataset, but can benefit in more challenging scenarios, e.g., datasets with crowded instances. The qualitative results are demonstrated in Fig. 16, where the detected bounding boxes look favorably good even under heavily occluded cases.

4.6 GFocal under Mobile Setting

Based on GFocal, we construct a lightweight dense object detector termed "NanoDet" which is especially suitable for end/embedded devices under. very limited computation resources (e.g., the mobile with ARM CPU platforms).

For real-time speed, we first adopt smaller input resolution like 320 × 320 and 416 × 416 following [44]. The small input size makes the higher feature levels P_6 and P_7 less important, thus we simply remove them to further reduce the inference time. As the detection head is considerably heavy in its original design, we utilize four strategies to reduce its complexity: 1) replacing ordinary convolutions by depth-wise separable convolutions; 2) halving the number of stacked convolutions from 4 to 2; 3) compressing the dimension 256 to 96; and 4) making classification branch and regression branch share the same convolutional towers. The classification branch models the Classification-IoU Joint Representation and the regression branch estimates the General Distribution of bounding boxes. The proposed GFocal loss is used to optimize these representations, respectively. For detection neck, we adopt a simplified Path Aggregation Network (PAN) [30] instead of traditional FPN, where the up/down-sampling operators are all based on feature interpolation for optimal speed. Further considering the parameter efficiency, ShuffleNetV2 [32] is applied for detection backbone.

During training, we employ a more aggressive training strategy. The models are trained using SGD optimizer with momentum 0.9 and weight decay 1e-4. Learning rate is linearly increased from 0.014 to 0.14 in the first 300 training steps and decayed by 0.1 at epoch 240, 260 and 275. We use horizontal flipping, random scaling (between 0.6 to 1.4), random shifting, and color jittering as data augmentation. Each model is trained for 280 epochs with total batch size 192 on a single GPU.

We compare NanoDet with popular lightweight detectors from YOLO series in Table 15. The performance is measured on Kirin 980 (4xA76+4xA55) ARM CPU based on ncnn[1]. It is observed that under the same input resolution 416×416, NanoDet is 1.8 AP higher, 2x faster and 6x smaller

[1] https://github.com/Tencent/ncnn

TABLE 15
The Comparisons Between Tiny Models on Kirin 980 (4xA76+4xA55) ARM CPU Based on Ncnn

Model	Input	AP	Latency	Params	Model Size
YOLOV3 tiny [37]	416^2	16.6	37.60ms	8.86M	33.7MB
YOLOV4 tiny [44]	416^2	21.7	32.81ms	6.06M	23.0MB
NanoDet **(ours)**	320^2	20.6	**10.23ms**	**0.95M**	**3.7MB**
NanoDet **(ours)**	416^2	**23.5**	16.44ms	**0.95M**	**3.7MB**

AP is validated on COCO `minival` *dataset with no testing time augmentation.*

than YOLOV4 [44], which demonstrates its wide application in end/embedded devices.

5 Conclusion

In this work, to effectively learn qualified and distributed bounding boxes for dense object detectors, we propose Generalized Focal Loss (GFocal) that generalizes the original Focal Loss from {1,0} discrete formulation to the continuous version. GFocal can be specialized into Quality Focal loss (QFL) and Distribution Focal Loss (DFL), where QFL encourages to learn a better joint representation of classification and localization quality, and DFL provides more informative and precise bounding box estimations by modeling their locations as General Distributions. Based on the close relation between the General Distributions and their localization quality, we also propose to learn reliable localization quality estimation, through the guidance of statistics of the learned General Distributions, which is an entirely new and completely different perspective in the literature. Extensive experiments and analyses validate its effectiveness, compatibility and efficiency. To further enlarge the impact of GFocal, we have provided a significantly lightweight version of it, termed NanoDet, which has state-of-the-art performance and inference speed. Given its effectiveness and efficiency, we hope that GFocal can serve as a strong and simple baseline for the community.

Acknowledgments

The authors would like to thank the editor and the anonymous reviewers for their critical and constructive comments and suggestions.

References

[1] Z. Cai and N. Vasconcelos, "Cascade R-CNN: Delving into high quality object detection," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2018, pp. 6154–6162.

[2] J. Cao, Y. Pang, J. Han, and X. Li, "Hierarchical shot detector," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2019, pp. 9704–9713.

[3] Y. Cao, K. Chen, C. C. Loy, and D. Lin, "Prime sample attention in object detection," 2019, *arXiv:1904.04821*.

[4] J. Chang, Z. Lan, C. Cheng, and Y. Wei, "Data uncertainty learning in face recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2020, pp. 5709–5718.

[5] K. Chen *et al.*, "Mmdetection: Open mmlab detection toolbox and benchmark," 2019, *arXiv:1906.07155*.

[6] Y. Chen, Z. Zhang, Y. Cao, L. Wang, S. Lin, and H. Hu, "RepPoints v2: Verification meets regression for object detection," in *Proc. Adv. Neural Inf. Process. Syst.*, 2020, pp. 5621–5631.

[7] J. Choi, D. Chun, H. Kim, and H.-J. Lee, "Gaussian YOLOv3: An accurate and fast object detector using localization uncertainty for autonomous driving," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2019, pp. 502–511.

[8] X. Chu, A. Zheng, X. Zhang, and J. Sun, "Detection in crowded scenes: One proposal, multiple predictions," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2020, pp. 12211–12220.

[9] J. Dai *et al.*, "Deformable convolutional networks," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2017, pp. 764–773.

[10] P. Dollar, C. Wojek, B. Schiele, and P. Perona, "Pedestrian detection: An evaluation of the state of the art," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 34, no. 4, pp. 743–761, Apr. 2011.

[11] L. Fan, X. Xiong, F. Wang, N. Wang, and Z. Zhang, "RangeDet: In defense of range view for LiDAR-based 3D object detection," in *Proc. IEEE/CVF Conf. Comput. Vis.*, 2021, pp. 2898–2907.

[12] R. Girshick, "Fast R-CNN," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2015, pp. 1440–1448.

[13] R. Girshick, J. Donahue, T. Darrell, and J. Malik, "Rich feature hierarchies for accurate object detection and semantic segmentation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2014, pp. 580–587.

[14] P. Hall, "On kullback-leibler loss and density estimation," *Ann. Statist.*, vol. 15, no. 4, pp. 1491–1519, 1987.

[15] K. He, G. Gkioxari, P. Dollár, and R. Girshick, "Mask R-CNN," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2017, pp. 2980–2988.

[16] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2016, pp. 770–778.

[17] Y. He, C. Zhu, J. Wang, M. Savvides, and X. Zhang, "Bounding box regression with uncertainty for accurate object detection," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2019, pp. 2883–2892.

[18] Z. Huang, L. Huang, Y. Gong, C. Huang, and X. Wang, "Mask scoring R-CNN," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2019, pp. 6402–6411.

[19] S. Ioffe and C. Szegedy, "Batch normalization: Accelerating deep network training by reducing internal covariate shift," in *Proc. Int. Conf. Mach. Learn.*, 2015, pp. 448–456.

[20] B. Jiang, R. Luo, J. Mao, T. Xiao, and Y. Jiang, "Acquisition of localization confidence for accurate object detection," in *Proc. Eur. Conf. Comput. Vis.*, 2018, pp. 816–832.

[21] K. Kim and H. S. Lee, "Probabilistic anchor assignment with iou prediction for object detection," in *Proc. Eur. Conf. Comput. Vis.*, 2020, pp. 355–371.

[22] T. Kong, F. Sun, H. Liu, Y. Jiang, and J. Shi, "FoveaBox: Beyond anchor-based object detector," 2019, *arXiv:1904.03797*.

[23] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," *Adv. Neural Inf. Process. Syst.*, vol. 25, pp. 1097–1105, 2012.

[24] H. Li, Z. Wu, C. Zhu, C. Xiong, R. Socher, and L. S. Davis, "Learning from noisy anchors for one-stage object detection," 2019, *arXiv:1912.05086*.

[25] X. Li, W. Wang, X. Hu, J. Li, J. Tang, and J. Yang, "Generalized focal loss V2: Learning reliable localization quality estimation for dense object detection," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2021, pp. 11627–11636.

[26] X. Li *et al.*, "Generalized focal loss: Learning qualified and distributed bounding boxes for dense object detection," in *Proc. 34th Adv. Neural Inf. Process. Syst.*, 2020, pp. 21002–21012.

[27] T.-Y. Lin, P. Dollár, R. Girshick, K. He, B. Hariharan, and S. Belongie, "Feature pyramid networks for object detection," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2017, pp. 936–944.

[28] T.-Y. Lin, P. Goyal, R. Girshick, K. He, and P. Dollár, "Focal loss for dense object detection," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2017, pp. 2999–3007.

[29] T.-Y. Lin *et al.*, "Microsoft COCO: Common objects in context," in *Proc. Eur. Conf. Comput. Vis.*, 2014, pp. 740–755.

[30] S. Liu, L. Qi, H. Qin, J. Shi, and J. Jia, "Path aggregation network for instance segmentation," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2018, pp. 8759–8768.

[31] W. Liu *et al.*, "SSD: Single shot multibox detector," in *Proc. Eur. Conf. Comput. Vis.*, 2016, pp. 21–37.

[32] N. Ma, X. Zhang, H.-T. Zheng, and J. Sun, "ShuffleNet V2: Practical guidelines for efficient CNN architecture design," in *Proc. Eur. Conf. Comput. Vis.*, 2018, pp. 122–138.

[33] H. Qiu, H. Li, Q. Wu, and H. Shi, "Offset bin classification network for accurate object detection," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2020, pp. 13185–13194.

[34] H. Qiu, Y. Ma, Z. Li, S. Liu, and J. Sun, "BorderDet: Border feature for dense object detection," in *Proc. Eur. Conf. Comput. Vis.*, 2020, pp. 549–564.

[35] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, "You only look once: Unified, real-time object detection," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2016, pp. 779–788.

[36] J. Redmon and A. Farhadi, "YOLO9000: Better, faster, stronger," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2017, pp. 6517–6525.

[37] J. Redmon and A. Farhadi, "Yolov3: An incremental improvement," 2018, *arXiv:1804.02767*.

[38] S. Ren, K. He, R. Girshick, and J. Sun, "Faster R-CNN: Towards real-time object detection with region proposal networks," in *Proc. Adv. Neural Inf. Process. Syst.*, 2015, pp. 91–99.

[39] H. Rezatofighi, N. Tsoi, J. Gwak, A. Sadeghian, I. Reid, and S. Savarese, "Generalized intersection over union: A metric and a loss for bounding box regression," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2019, pp. 658–666.

[40] S. Shao *et al.*, "Crowdhuman: A benchmark for detecting human in a crowd," 2018, *arXiv:1805.00123*.

[41] Y. Shi and A. K. Jain, "Probabilistic face embeddings," in *Proc. IEEE/CVF Int. Conf. Comput. Vis.*, 2019, pp. 6901–6910.

[42] Z. Tian, C. Shen, H. Chen, and T. He, "FCOS: Fully convolutional one-stage object detection," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2019, pp. 9626–9635.

[43] L. Tychsen-Smith and L. Petersson, "Improving object localization with fitness NMS and bounded IoU loss," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2018, pp. 6877–6885.

[44] C.-Y. Wang, A. Bochkovskiy, and H.-Y. M. Liao, "Scaled-yolov4: Scaling cross stage partial network," 2020, *arXiv:2011.08036*.

[45] J. Wang *et al.*, "Side-aware boundary localization for more precise object detection," 2019, *arXiv:1912.04260*.

[46] S. Wu and X. Li, "Iou-balanced loss functions for single-stage object detection," 2019, *arXiv:1908.05641*.

[47] S. Wu, X. Li, and X. Wang, "IoU-aware single-stage object detector for accurate localization," *Image Vis. Comput.*, vol. 97, 2020, Art. no. 103911.

[48] Z. Yang, S. Liu, H. Hu, L. Wang, and S. Lin, "RepPoints: Point set representation for object detection," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2019, pp. 9656–9665.

[49] H. Zhang, Y. Wang, F. Dayoub, and N. Sünderhauf, "VarifocalNet: An IoU-aware dense object detector," 2020, *arXiv:2008.13367*.

[50] S. Zhang, C. Chi, Y. Yao, Z. Lei, and S. Z. Li, "Bridging the gap between anchor-based and anchor-free detection via adaptive training sample selection," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2020, pp. 9756–9765.

[51] X. Zhang, F. Wan, C. Liu, R. Ji, and Q. Ye, "Freeanchor: Learning to match anchors for visual object detection," in *Proc. Adv. Neural Inf. Process. Syst.*, 2019, pp. 147–155.

[52] Z. Zheng, R. Ye, P. Wang, J. Wang, D. Ren, and W. Zuo, "Localization distillation for object detection," 2021, *arXiv:2102.12252*.

[53] X. Zhou, D. Wang, and P. Krähenbühl, "Objects as points," 2019, *arXiv:1904.07850*.

[54] B. Zhu *et al.*, "Autoassign: Differentiable label assignment for dense object detection," 2020, *arXiv:2007.03496*.

[55] L. Zhu, Z. Xie, L. Liu, B. Tao, and W. Tao, "Iou-uniform R-CNN: Breaking through the limitations of RPN," 2019, *arXiv:1912.05190*.

[56] X. Zhu, H. Hu, S. Lin, and J. Dai, "Deformable ConvNets V2: More deformable, better results," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2019, pp. 9300–9308.

Xiang Li received the PhD degree from the Nanjing University of Science and Technology, Jiangsu, China, in 2020. He is currently an associate professor with the College of Computer Science, Nankai University. His research interests include CNN/Transformer backbone, object detection, knowledge distillation and self-supervised learning. He has published more than 20 papers in top journals and conferences such as *IEEE Transactions on Pattern Analysis and Machine Intelligence*, CVPR, NeurIPS, etc.

Chengqi Lv received the BS degree from Southeast University, China, in 2018. He is currently a researcher with OpenMMLab, Shanghai AI Laboratory. His research interests include object detection and instance segmentation.

Wenhai Wang received the PhD degree from the Department of Computer Science, Nanjing University, in 2021. He is currently a research scientist with Shanghai AI Laboratory. His main research interests include CNN/Transformer backbone, object detection, semantic/instance/panoptic segmentation, vision-language model, autonomous driving perception, and optical character recognition. He has published more than 20 papers in vision journals and conferences, such as *IEEE Transactions on Pattern Analysis and Machine Intelligence*, CVPR, ICCV, ECCV, etc.

Gang Li received the BS degree from Shenzhen University, China, in 2018. He is currently working toward the PhD degree with the School of Computer Science and Engineering, Nanjing University of Science and Technology. His research interests include object detection, pedestrian detection and instance segmentation.

Lingfeng Yang received the BS degrees from the Nanjing University of Science and Technology, China in 2020. He is currently working toward the PhD degree with the Department of Computer Science and Engineering, Nanjing University of Science and Technology. His research interests include object detection, defect detection, and fine-grained visual categorization.

Jian Yang received the PhD degree from the Nanjing University of Science and Technology (NUST), on the subject of pattern recognition and intelligence systems, in 2002. In 2003, he was a postdoctoral researcher with the University of Zaragoza. From 2004 to 2006, he was a postdoctoral fellow with Biometrics Centre of Hong Kong Polytechnic University. From 2006 to 2007, he was a postdoctoral fellow with the Department of Computer Science of New Jersey Institute of Technology. From 2006 to 2007, he was a chang-Jiang professor with the School of Computer Science and Engineering of NUST. He is currently a distinguished professor with the College of Computer Science of Nankai University. He is the author of more than 200 scientific papers in pattern recognition and computer vision. His papers have been cited more than 30000 times in the Scholar Google. His research interests include pattern recognition, computer vision and machine learning. Currently, he is an associate editor of *Pattern Recognition*, *Pattern Recognition Letters*, *IEEE Trans. Neural Networks and Learning Systems*, and *Neurocomputing*. He is also a fellow of the IAPR.