Automatic Label Assignment for Object Detection

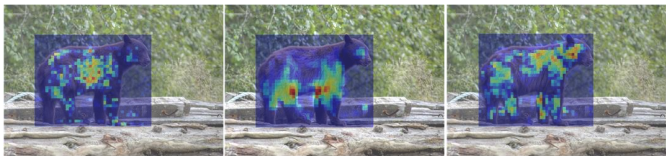
Hao Wang, Tong Jia, *Member, IEEE,* Qilong Wang, *Member, IEEE,* and Wangmeng Zuo, *Senior Member, IEEE*

***Abstract*—Label assignment, which aims to classify region proposals as positive or negative samples depending on the correlations between their classification and localization predictions with the corresponding ground truth, is recognized as an essential ingredient in object detection and strongly affects the detection performance. Recently, some dynamic label assignment methods have been proposed to overcome the limitations of the static methods and achieve promising performance improvement. Despite eliminating the restrictions of the human prior sampling knowledge in static methods, existing dynamic principles usually suffer from two weaknesses. First, most of them deploy mixture models or implicit branch in prediction head to coarsely estimate the spatial distribution of the positive samples for objects. They give little attention to the effect of appearance information of the objects. Furthermore, these methods still cannot perceive the quality distribution of the positive samples, and these low-quality samples lead to adverse effects on the detection performance. To address issues, this paper presents a novel automatic label assignment for object detection. Specifically, our method first introduces an instance property branch into object detection pipeline to distinguish the foreground from the background. Then, an objectness prediction module which is composed by the confidence and weight mechanisms is developed to generate the positive and negative weight maps for the objects. The instance property branch and objectness prediction module can provide a coarse-to-fine optimization framework to make our method realize the appearance of the objects. Finally, a positive sample selection strategy is proposed to explore the quality statistical distribution of the positive samples, which are trained by different designed label targets. We evaluate our method on the MS COCO dataset and we achieve 48.4%, 47.9%, 48.0% and 49.3% on ResNet-101, ResNeXt-101, DCN-ResNet-101 and DCN-ResNeXt-101 in terms of $AP_{0.5:0.95}$, respectively. We evaluate the timing complexity of ALA by calculating the inference speed and the frame per second (FPS) for these four backbones are 11.9, 10.4, 9.9 and 8.0, respectively. The experiment results demonstrate that we can obtain clear improvement over the competing methods with favorable performance compared to the state-of-the-arts.**

***Index Terms*—Object detection, appearance model, data sampling, labeling**

I. Introduction

OBJECT detection is served as a hot topic in computer vision community and attracts extensive attentions in recent years. It is well known that current object detection methods deploy a dense prediction pipeline and the extracted proposals are distinguished by specific targets (*i.e.*, positive or negative) according to the ground truth, which is also recognized as label assignment. As shown in Figure 1 (a), most of existing methods [1], [3], [4] implement a static strategy for label assignment. Anchor-based methods like Faster R-CNN [4] and RetinaNet [3] set a pre-defined threshold of Intersection-over-Union (IoU) between region proposals across different levels in Feature Pyramid Network [5] and ground truth as the criterion standard for positive and negative samples. While other anchor-free methods (*e.g.*, FCOS [1] and FoveaBox [6]) treat the central areas which are inside a fixed region around the center point of the ground truth as positive samples, and the rest areas are negative ones. However, these static strategy methods are heavily depended on human prior sampling knowledge, which are usually heuristically set by experience. Furthermore, they ignore the fact that objects vary with different sizes, shapes and angles, and the static strategy can not perceive the geometric differences between various objects, especially for some occlusion conditions.



(a) FCOS [1] (b) IQDet [2] (c) Ours

Fig. 1. Positive sample distributions of different label assignment methods, and brighter color indicates higher probability of the sample to be positive. (a) Fully Convolutional One-Stage Object Detection (FCOS) [1] deploys a static strategy and divides the locations in a fixed region (*i.e.*, inside the central of the objects) as positive samples. (b) Instance-wise Quality Distribution Detection (IQDet) [2] utilizes a Gaussian mixture model to estimate the shape of the objects and selects positive samples through a probabilistic manner. (c) We propose an automatic label assignment method, which takes the advantages of appearance of the objects for guiding the label assignment.

Recently, several works [2], [7], [8], [9], [10] have been proposed to alleviate the issues in static label assignment strategy. Instead of leveraging the pre-defined rules for label assignment, these methods adopt a dynamic strategy with little human prior knowledge. Adaptive Training Sample Selection (ATSS) [7] sets the threshold for the division of positive and negative samples depending on the statistical values of the IoU between proposals and ground truth. Probabilistic Anchor Assignment (PAA) [8] considers the learning status of the model and adaptively separates the anchors into positive and negative samples according to their probability distribution, which is fit by Gaussian mixture model. Optimal Transport Assignment (OTA) [9] provides a new perspective to solve the label assignment issue and proposes to formulate it as an

This work is supported in part by the National Key Research and Development Project of China under No.2022YFF0902401, the National Natural Science Foundation of China (NSFC) under No.s 62173083, 62206043, 62276186 and U22A2063, the China Postdoctoral Science Foundation under No.s 2023M730517 and 2024T170114, and the Liaoning Provincial "Selecting the Best Candidates by Opening Competition Mechanism" Science and Technology Program under NO.2023JH1/10400045.

H. Wang and J.Tong are with the College of Information Science and Engineering, Northeastern University, Shenyang, China. e-mail: ddsywh@yeah.net, jiatong@ise.neu.edu.cn.

Q. Wang is with the College of Intelligence and Computing, Tianjin University, Tianjin, China. e-mail: qlwang@tju.edu.cn.

W. Zuo is with the School of Computer Science and Technology, Harbin Institute of Technology, Harbin, China. e-mail: wmzuo@hit.edu.cn.

Corresponding author: Tong Jia

optimal operation problem in optimization theory. The best matching solution is to find out the minimum transportation cost between the anchors and ground truths based on their classification and regression losses. Obviously, PAA and OTA deal with traditional label assignment in a mathematical manner, and achieve performance improvement compared with static counterparts. Furthermore, Differentiable Label Assignment (AutoAssign) [10] and IQDet [2] consider the appearance of the objects with different weighting mechanisms in label assignment. Specifically, AutoAssign implements an additional implicit branch in prediction head to automatically simulate the shape of the instances and adjust their specific distribution. As shown in Figure 1 (b), IQDet evaluates the instance-wise quality distribution based on the Gaussian mixture model and automatically selects the training samples as positive among the spatial locations in a probabilistic manner. However, such methods heavily rely on the pre-defined Gaussian models and still fail to provide a satisfying solution to the human prior knowledge issue in label assignment.

Although aforementioned dynamic label assignment methods achieve promising improvement, they still suffer from two limitations. Firstly, current dynamic label assignment methods always deploy a coarse shape estimation for the distribution of the objects by Gaussian mixture model or implicit branch in prediction head, inevitably introducing noisy samples during the positive sample selection process. They ignore that the intrinsic appearance property of the objects can easily distinguish the positive or negative samples by estimating the foreground and background of the objects. Moreover, current methods cannot perceive the quality distribution of the positive samples, and some low-quality samples are classified as positive samples, introducing adverse effects to the object detection performance. For better addressing the limitations of current dynamic label assignment methods and further improving the object detection performance, we propose a novel automatic label assignment for object detection (ALA) in this paper. Specifically, we first build an instance property branch, which is utilized to obtain the foreground and background information of the instance, since foregrounds are more likely to be distinguished as positive samples and backgrounds are classified as negative samples. Then we develop an objectness prediction module which is constituted by confidence and weight mechanisms. Confidence mechanism obtains the probability of the positive and negative samples. While weight mechanism contains two parts, center and shape weights. The center weight can capture the the positive sample distribution for each category, and the shape weight can adapt to appearance of the object from the the scale and spatial dimensions. The instance property branch and objectness prediction module supply a coarse-to-fine optimization for the appearance of each instance, and the combination of them can realize the quality and probability information of the positive and negative samples. As elaborated in Figure 1 (c), our method can effectively and automatically adapt to the appearance of the objects. Finally, we design a positive sample selection strategy to evaluate the quality statistical distribution of the positive samples. These positive samples are optimized with different targets according to their quality scores. The proposed method can be trained in an end-to-end learning manner. To verify the effectiveness of ALA, we conduct experiments on MS COCO [11] and PASCAL VOC [12] datasets using various backbone models [13], [14], [15]. The contributions of this work can be summarized as follows:

1) First, we consider the appearance properties of the objects, and design an explicit mechanism, which is constituted of an instance property branch and an objectness prediction module. They conduct a coarse-to-fine estimation pipeline for the shapes of the objects, which is beneficial for making the module realize the foreground and background parts. Those information can be regarded as references and prior knowledge to clearly clarify the positive samples from the negative ones in the label assignment process.
2) Second, we design a positive sample selection strategy to calculate the quality statistical distribution of the positive samples. Instead of treating each positive sample equally, the proposed strategy benefits the optimization of the positive samples with different targets according to their corresponding quality scores.
3) At last, we evaluate our proposed methods on two different object detection benchmarks (*i.e.*, MS COCO [11] and PASCAL VOC [12]), and the experiment results show that the proposed method achieves clear performance improvement over current methods.

Current competitive counterparts usually deploy the static assignment manner, which suffers from the the restrictions of the human prior sampling knowledge. Despite some recently proposed ones start to explore dynamic label assignment framework, they only adopt some implicit styles to obtain a coarse estimation for the spatial shapes of the positive samples. Compared to those implicit manners, we implement an explicit module to take full advantage of the appearance information of the objects through a coarse-to-fine estimation pipeline, where the positive and negative samples can be better distinguished. Besides, our positive sample selection strategy gets rid of the limitations in current methods, where each positive sample is treated equally with the same optimization target. It calculates the quality distributions of the positive samples and classifies those samples into different parts with different optimization targets according to their qualities (*i.e.*, classification score and localization accuracy).

The remainders of this paper are organized as follows: Section II reviews the related works about different label assignment manners in object detection. In Section III, we give detailed descriptions of our proposed automatic label assignment for object detection, and show the experimental results including ablation studies and comparisons in Section IV. Finally, Section V demonstrates several conclusions for the proposed method.

II. Related Work

Label assignment is a crucial factor to affect the performance of object detection and existing label assignment methods can be roughly categorized into two groups, *i.e.*,

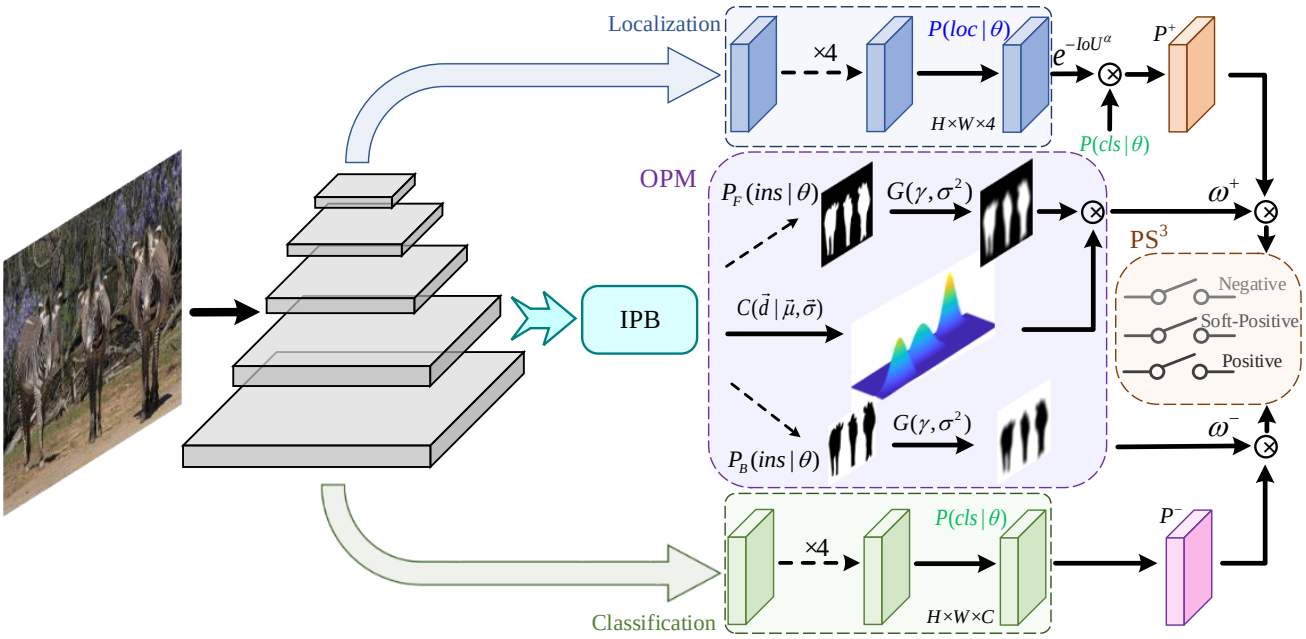


Fig. 2. Overview of our proposed automatic label assignment for object detection, whose core is an instance property branch (IPB), objectness prediction module (OPM) and positive sample selection strategy (PS^3). IPB takes the output feature from feature pyramid network and predicts the foreground and background of the image, which can be optimized by back-propagation during the training process. OPM contains confidence and weight mechanisms and calculates the corresponding confidences of positive and negative samples P^{+} and P^{-} and weights ω^{+} and ω^{-} based on the predictions of classification $P(cls|\theta)$, localization $P(loc|\theta)$ and instance property branch $P(int|\theta)$, respectively. PS^3 further distinguishes the positive samples into different sets (*i.e.*, positive, soft positive and negative) according to their quality scores and trains the positive samples with different target labels.

static and dynamic manners. Static manner such as Faster R-CNN [4] and its variants [16], [17] usually deploys IoU as a criterion to assign the positive and negative labels to the proposals by setting a fixed threshold (*e.g.*, 0.7 for positive and 0.3 for negative). While recently proposed anchor-free methods conduct a different static rule for label assignment. FCOS [1], Feature Alignment Object Detection (G-RFA) [18] and other point-based methods [19] directly assign the points in a certain region around the center of the object as positive samples. However, above static methods all require human prior sampling knowledge and the hand-crafted thresholds or regions ignore the diversity distribution of the shapes in objects. They are not always the optimal options for the assignment criterions.

For adaptively fitting the shape of the objects, many dynamic label assignment methods are proposed to get rid of the hand-crafted limitations of the static ones. Instead of using pre-defined fixed anchors, Guided Anchoring [20] automatically predicts the width, height and area of the anchors based on the distribution of the objects. MetaAchor [21] generates dynamic anchors from arbitrary customized prior boxes. FreeAnchor [22] firstly builds an anchor candidate for each ground truth based on the IoU of the anchors and then utilizes a detection-customized Mean-Max likelihood to implement positive and negative sample selection for the top-K anchors in each candidate set from the previous step. Soft Anchor-Point Object Detection (SAPD) [23] implements soft-weighted and soft-selected strategies to address the false attention issue within and across different pyramid levels. Pseudo-IoU [24] proposes a pseudo-IoU metric into detection pipeline to evaluate the centered pseudo box as positive or negative samples based on the pseudo-IoU threshold. Single-Shot Refinement Network (RefineDet++) [25] proposes an alignment convolution to improve the regression accuracy and multi-label assignment. Going beyond above aforementioned methods which focus on anchor designation in label assignment, ATSS [7] adaptively adjusts the threshold for positive sample selection through calculating the statistical characteristics of the whole samples. Similar as ATSS, PAA [8] solves the label assignment issue through a probabilistic manner by using the IoU as the measurement for the proposals. IQDet [2] and AutoAssign [10] both estimate the shape of the objects through Gaussian models. Specifically, IQDet deploys a Gaussian mixture model and quality sampling strategy to select positive samples in a probabilistic manner. Autoassign additionally attaches an implicit branch to adjust the category-specific prior distributions. OTA [9] provides a different insight for label assignment. It transfers label assignment as an optimal transport problem to find the best matching scheme based on the cost between each proposal and ground truth. Similar as OTA, Optimal Partition Assignment (OPA) [26] also formulates the label assignment as an optimal problem and selects an optimal divider line by mapping the classification and localization score to a two dimension coordinate. Though achieving promising performance gains, they provide a coarse and indirect estimation for the shape of the objects during positive sample selection. Besides, they can not perceive the distribution of the positive samples, which should be optimized by different targets. Balanced Sample Assignment (BSAODet) [27] introduces a quality-balanced

sample assignment strategy to dynamically collect the high-quality samples according to their performance and geometric constraints. Diag-IoU [28] designs a similarity measurement based on the diagonals of the boxes, and the proposed Diag-Iou can be utilized as a label assignment principle by considering the divergences between the predicted and target boxes. G-RFA [18] attempts to alleviate the label assignment issue from the aspect of feature alignment. It deploys an attention mechanism to remove the influence of the redundant contextual information for focusing on the foregrounds of the objects. Although these methods provide promising solutions for current label assignment issue, they take little consideration to the effect of the appearance information of the objects. Diag-IoU [28] and G-RFA [18] cannot filter out the low-quality negative samples, which bring adverse effect to the object detection performance. Besides, BSAODet [27] individually collects stable high-quality samples for each class, neglecting the union distributions for all the objects in the images.

Different from above methods which implement a many-to-one label assignment manner, some recently proposed methods attempt to conduct an one-to-one label assignment form, where one ground truth is only assigned by one proposal. Object Detection with Transformers (DETR) [29] and Deformable DETR [30] are two pioneering works which are built on Transformer. They realize an end-to-end detection pipeline without any post-processing step (*i.e.*, NMS) and prior knowledge (*e.g.*, constraint for pre-defined anchors). Sparse R-CNN [31] replaces the hand-designed dense object candidates with a fixed sparse set of learned object proposals and the final predictions are directly output without NMS post procedure. Additionally, Positive Sample Selector (PSS) [32] and Prediction Aware Label Assignment (POTO) [33] attach a positive sample selector branch and 3D max filter in the prediction head to choose the best positive sample for each ground truth. To present the generalization ability of our method, we also adopt ALA to one-to-one label assignment manner and show the results in Experiment section.

III. Method

In this section, we will introduce our proposed automatic label assignment for object detection which is presented in Figure 2. We firstly describe the detail of instance property branch. Then an objectness prediction module is developed to guide the label assignment in object detection. Finally, we present a positive sample selection strategy to apart positive samples depending on their corresponding quality scores and train them with different targets.

A. Instance Property Branch

Generally speaking, the foreground and background predictions can provide complementary and auxiliary guidance for the label assignment in object detection pipeline, since foreground locations of the instances are more inclined to correspond to positive samples, and background locations seem more like to be the negative samples. However, existing object detection methods neglect the properties inside the foreground and background, and use other coarse estimation methods (*e.g.*, Gaussian mixture model or implicit branch), bringing limited influence on the performance. For considering the intrinsic attributes of the foreground and background, we design an instance property branch in this paper, which is shown in Figure 3.

Specifically, the instance property branch is constructed based on the output features of the feature pyramid network [5]. Inspired by [34], we integrate multi level features in feature pyramid network for addressing the lack of discriminative information of single level feature. We incorporate high-level feature $P4$ and low-level feature $P2$ from feature pyramid network for both considering the global information from high level feature and local information from low level feature and obtain a better feature representation. Since features from different levels have different spatial resolutions, feature maps need to share the same size for the following concatenation operation. A bilinear interpolation upsampling operation is used to increase the size in later level (*i.e.*, $P4$) to the same spatial scale as $P2$. A 1×1 convolutional layer is implemented to align the low level feature (*i.e.*, $P2$) to a common representation. Then these two transformed feature maps from two levels are subsequently fused by concatenation operation. Moreover, we add an additional convolutional layer to calibrate the semantic gap between different layers and reduce the channel dimension to 2, as we only predict the foreground and background of the image. Finally, a *Sigmoid* layer is attached to output the final predictions. To this end, our instance property branch can be formulated as:

$$\begin{cases} F = \sigma(\mathbf{W}_C * concat(\widetilde{F}_{P2}, \widetilde{F}_{P4})), \\ \widetilde{F}_{P2} = \mathbf{W}_S * F_{P2}, \\ \widetilde{F}_{P4} = \textbf{Upsample}(F_{P4}), \end{cases} \tag{1}$$

where σ indicates *Sigmoid* function and **Upsample** is the bilinear interpolation operation. $*$ is a convolutional operation with 1×1 kernel $\mathbf{W}$. The sizes of $\mathbf{W}_c$ and $\mathbf{W}_s$ are $\mathbb{R}^{N\times C\times 1\times 1}$, where N and C denote the batch size and channel number, respectively. F_{P2} and F_{P4} are the feature maps from the second level $P2$ and fourth level $P4$ in feature pyramid network. The sizes of F_{P2} and $\widetilde{F}_{P2}$ are $\mathbb{R}^{N\times C\times \frac{W}{8}\times \frac{H}{8}}$, and the sizes of F_{P4} and $\widetilde{F}_{P4}$ are $\mathbb{R}^{N\times C\times \frac{W}{32}\times \frac{H}{32}}$, where W and H are the width and height of the input images. F is the output foreground and background predictions.

Clearly, we design a simple and straightforward instance property branch to adaptively capture the appearance distribution for each instance. Compared to the implicit branch in AutoAssign [10], which only provides the foreground prediction, our proposed branch can acquire more diverse information for the appearance characteristic in each instance. It is optimized with the classification and localization branches and shares the same supervision with them, so no explicit labels are needed, where we will explain it in the next subsection.

B. Objectness Prediction Module

Label assignment acts a vital factor to influent the performance in object detection, and current static label assignment methods [1], [3], [4] usually require prior knowledge to determine the distribution of the anchor boxes, ignoring

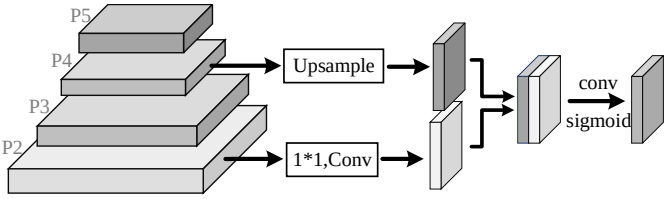


Fig. 3. Illustration of instance property branch. It incorporates $P4$ and $P2$ from feature pyramid network by upsampling and convolution operations, respectively. A sigmoid layer is attached at the end of the network to obtain the final prediction, which corresponds to the foreground and background of the input image.

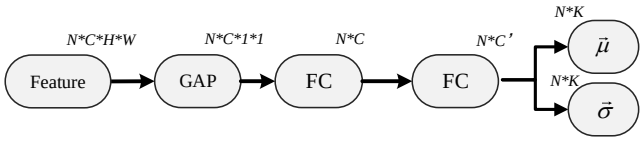


Fig. 4. Prediction network for center weighting parameters $\vec{\mu}$ and $\vec{\sigma}$. GAP and FC represent global average pooling and fully connected layer, respectively.

the variances of sizes and shapes between different objects. Despite some dynamic label assignment manners are proposed to alleviate the limitations of the static ones, they conduct an implicit way by deploying Gaussian mixture model, which only provides a coarse estimation to the shape of the objects. We take a further step in this paper and directly embed the appearance characteristic into object detection framework. To this end, we design an objectness prediction module (OPM), which can take full advantage of the appearance information (*e.g.*, shape, foreground and background) of the objects.

Generally, all locations across different feature pyramid network levels will be considered as positive or negative samples. For better distinguishing them, we design confidence and weighting mechanisms in OPM and calculate corresponding confidence and weighting values for these two kinds of samples. Specifically, confidence mechanism evaluates the quality of each sample, and weight mechanism assesses the probability of samples to be positive or negative. As suggested in [35], [36], classification and localization explore different properties between the predictions and ground truth. Because the original localization prediction outputs are the offsets of the boxes, we consider to convert the offsets to likelihood form:

$$P_i(loc|\theta) = e^{-IoU_i^{\alpha}}, \tag{2}$$

where α is the hyper-parameter, and it is decided through cross validation. IoU is the Intersection of Union between the prediction boxes and ground truth. i represents the location. As IoU represents the intersection-of-union between the predicted bounding box and its corresponding ground truth label, its value is between 0 and 1. We utilize α in Eqn. (2) to control the values of the localization predictions $P_i(loc|\theta)$, where the predictions from different locations are within smaller margins compared to using IoU alone. To this end, the positive confidence P_i^+ and negative confidence P_i^- are represented as:

$$\begin{cases} P_i^+ = P_i(cls|\theta)^{\beta} \cdot P_i(loc|\theta)^{1-\beta}, \\ P_i^- = P_i(cls|\theta), \end{cases} \tag{3}$$

where $P_i(cls|\theta)$ is the classification score and $*$ is the dot production operator. θ denotes the parameters of the network and β is the hyper-parameter to balance each component in P^+. Confidences for negative samples are only performed by classification score as they are outside the ground truth.

Meanwhile, we proposed a weight mechanism to measure the probability which distinguishes the samples to be positive or negative. Specifically, we first design a center weight to assign the locations near the center of the objects with higher weighs and provide relative low weights to the locations far from the center. Inspired by [10], we utilize the Gaussian function as the center weight:

$$\mathcal{C}(\vec{d}_i|\vec{\mu},\vec{\sigma}) = e^{\frac{-(\vec{d}_i-\vec{\mu})^2}{2\vec{\sigma}^2}}, \tag{4}$$

where $\vec{d}_i$ represents the offset of the location i to its corresponding center of the object along the $\mathbf{x}$ and $\mathbf{y}$ axis. $\vec{\mu}$ and $\vec{\sigma}$ are the category-wise learnable parameters to decide the shape of the Gaussian function. As shown in Figure 4, we take the features from feature pyramid network as input and implement global average pooling layer (GAP) and fully connected layer (FC) to obtain $\vec{\mu}$ and $\vec{\sigma}$ of the shape (N, K), where K is the number of the categories and N is the batch size. The initial values for $\vec{\mu}$ and $\vec{\sigma}$ are 1 and 0, respectively, and they are updated during the training process. Note that center weight is deployed in each layer in feature pyramid network and we exploit the down-scale factors to normalize the values in each center weight.

Besides, we further embed the instance property prediction into shape weight in each location i for better distinguishing the positive samples from the negative ones. Let $P_F(ins|\theta)$ and $P_B(ins|\theta)$ be the foreground and background property predictions from IPB, (*i.e.*, $\sigma(F) = (P_F(ins|\theta), P_B(ins|\theta))$), the shape weight can be formulated as:

$$\begin{cases} \mathcal{M}_i^+ = P_{F_i}(ins|\theta) * G(\gamma, \sigma^2), \\ \mathcal{M}_i^- = P_{B_i}(ins|\theta) * G(\gamma, \sigma^2), \end{cases} \tag{5}$$

where G represents Gaussian blur kernel with radius γ and variance σ , which are set to be 3 and 2, respectively. They are fixed during the training process. We conduct Gaussian blur operation to smooth the boundary of the instance property output $P_\star(ins|\theta)$. Finally, we combine Eqns. (4) and (5) to achieve the positive and negative weights ω^+ and ω^- in the weight mechanism:

$$\begin{cases} \omega_i^+ = C(\vec{d}_i|\vec{\mu},\vec{\sigma}) \cdot \mathcal{M}_i^+, \\ \omega_i^- = \mathcal{M}_i^-. \end{cases} \tag{6}$$

By combing Eqns. (3) and (6) together, we obtain the quality scores for positive and negative samples:

$$\begin{cases} \mathcal{S}_i^+ = P_i^+ \cdot \omega_i^+, \\ \mathcal{S}_i^- = P_i^- \cdot \omega_i^-. \end{cases} \tag{7}$$

Through designing the confidence and weigh mechanisms, we can obtain quality scores which are applied to automatically and dynamically assign the targets to the samples.

Algorithm 1 Positive Sample Selection Strategy

Input: Positive sets R_i^+ and R_i^-, which are inside and outside the i-th ground truth bounding box at all the levels in feature pyramid network, respectively, and the corresponding score set $\mathcal{S}_i$ based on Eqn. (7).

Output: Hard positive sample sets R_i^H, soft positive sample set R_i^S and negative sample set R_i^-;

Calculate the average value m_i for set $\mathcal{S}_i$: $m_i = Mean(\mathcal{S}_i)$;
Calculate the standard deviation v_i for set $\mathcal{S}_i$: $v_i = std(\mathcal{S}_i)$;
for $j \in R_i^+$ and $s_{i_j} \in \mathcal{S}_i$ **do**
  if $p_{i_j} \geq m_i + v_i$ **then**
    $R_i^H = R_i^H \cup j$; `#Hard Positive Sample`
  else if $p_{i_j} < m_i + v_i$ and $p_{i_j} \geq m_i - v_i$ **then**
    $R_i^S = R_i^S \cup j$; `#Soft Positive Sample`
  else
    $R_i^- = R_i^- \cup j$; `#Negative Sample`
  end if
end for

C. Positive Sample Selection Strategy

Intuitively, the locations which are inside the ground truth bounding box are treated as positive samples and locations which are outside ground truth are divided into negative samples on the contrary. However, some locations may correspond to the background even though they are inside the ground truth. Besides, current label assignment methods [1], [4], [10] usually assign hard labels (*i.e.*, 1) to the whole positive samples. They ignore that not all positive samples fit for the ground truth very well and training those ambiguous positive ones which stay across the line between positive and negative with hard label may bring slight effect to the detection performance.

Based on above observations, we propose a positive sample selection strategy, which can perceive the quality statistical distribution of the samples and further meticulously distinguish the positive samples into different parts. As shown in Alg. 1, we compute average and standard deviation values for the positive sample set R^+. Then the positive samples are divided into different sets depending on their quality scores and trained by different labels. Specifically, the positive samples whose quality values are higher than the summation of mean m and standard deviation v are considered as certain samples and categorized into hard positive sample set R^H. These positive samples are trained with hard labels:

$$\ell_{R^H} = -\sum_{n=1}^{N} \sum_{i \in R_n^H} log(s_i), \tag{8}$$

where N denotes the numbers of ground truth. Apart from the certain positive samples, we classify the positive samples where their quality values are between $m - v$ and $m + v$ as uncertain and ambiguous positive samples which are assigned to soft positive sample set R^S. We exploit a soft label ξ_i to evaluate their positive or negative degree since they may be across the boundary of positive and negative:

$$\ell_{R^S} = -\sum_{n=1}^{N} \sum_{i \in R_n^S} (\xi_i \times log(s_i) + (1 - \xi_i) \times log(1 - s_i)). \tag{9}$$

In this paper, ξ_i is adaptively adjusted during the training process to keep the network hold strong feature representation ability:

$$\xi_i = \frac{1}{2} \frac{s_i}{\max_{i \in R^S}(s_i)} * (1 + \cos(\frac{it_c}{it_T}\pi)), \tag{10}$$

where it_c and it_T denote the current iteration and total iterations. We use the ratio between s_i and $\max(s_i)$ to normalize the scores of different samples into the same scale and set the soft label to be related to the quality scores for considering that the samples with higher prediction scores should be more like to be categorized as positive. The value of ξ is enlarged at early training stage to introduce more positive supervision, and it is gradually decreased since the network has more power to distinguish the ambiguous positive samples.

Even though the rest of positive samples are inside the ground truth bounding box, they may correspond to the background of the object with lower quality score. So we select them as negative sample set R^- and the training loss for R^- is conducted as:

$$\ell_{R^-} = -\sum_{n=1}^{N} \sum_{i \in R_n^-} \mathbf{FL}(s_i, 0), \tag{11}$$

where $\mathbf{FL}$ indicates the Focal Loss [3]. To this end, the whole network can be trained in a differentiable manner. We integrate Eqns. (8), (9) and (11) together, and the overall loss for ALA can be represented as:

$$\mathcal{L} = \ell_{R^H} + \ell_{R^S} + \ell_{R^-}. \tag{12}$$

We analyse the complexity of the Alg. 1. Since we can simultaneously handle those positive samples belonging to the same ground truth bounding box, the computational complexity of this algorithm is $\mathcal{O}(N)$, where N denotes the numbers of the ground truth bounding boxes in the images. OTA [9] deploys an optimal transportation pipeline for label assignment, and the optimization process contains many matrix multiplications, which should be implemented on GPU devices for accelerating. IQDet [2] introduces the Gaussian mixture model to capture the appearance distributions of the objects, and the added RoI-Alignment layer [37] occupies additional computation cost, where it also needs to be implemented on GPU. Compared to these state-of-the-art methods, our proposed algorithm introduces negligible computation complexity, and it can be run in parallel without GPU device.

IV. Experiment

In this section, we evaluate our ALA on MS COCO [11] and PASCAL VOC [12] datasets, where FCOS [1] is mainly deployed to build our ALA. We firstly provide ablation studies on the key components of ALA, including instance property branch, objectness prediction module and positive sample selection strategy. Then, we give comprehensive comparisons

TABLE I
RESULTS OF ALA WITH DIFFERENT INSTANCE PROPERTY BRANCHES ON THE VALIDATION SET OF MS COCO.

Method	$AP_{0.5:0.95}$	$AP_{0.5}$	$AP_{0.75}$
Baseline	37.0	55.8	39.7
Center-ness [1]	39.8	60.4	43.2
$ImpObj$ [10]	40.1	60.5	43.8
ALA w/o IPB	39.5	59.9	43.5
ALA	**40.9**	**61.3**	**44.6**

TABLE II
SEPARATION RESULTS (%) OF ALA WITH OR WITHOUT CENTER WEIGHTING ($\mathcal{C}$) AND SHAPE WEIGHTING (G) ON THE VALIDATION SET OF MS COCO.

method	IoU_{FG}	IoU_{BG}
w/o C and G	67.8	74.7
with C and G	88.4	91.5

with counterparts and state-of-the-arts methods on several backbone models. Finally, we show the generalization ability of our ALA by extending it to one-to-one label assignment manner.

A. Implementation Details

We employ various backbone architectures (*e.g.*, ResNet [14], ResNeXt [13] and DCN [15]) pre-trained on ImageNet [38] as backbone models. Following the common training configurations, we exploit the same parameter setting and protocol as the competing counterpart [1]. We train ALA on the training set of MS COCO, validate the parameters in the key components on the validation set and represent the results on the $test-dev$ set for comparing with state-of-the-arts. The mean Average Precision (mAP) under $IoU = [0.5 : 0.95]$ is used as evaluation metric. We implement our experiments on a sever equipped with two NVIDIA RTX 4090 GPUs with Memory 128G and CPU Core i9-13900K. While the software environment is built based on Linux system (Ubuntu 20.04.6). We utilize MMDetection benchmark [39] with version 3.1.0 as our code base. The official PyTorch version is 2.0.0 with CUDA 11.1 and Python 3.8.

B. Ablation Study

In this subsection, we make ablation studies on our method, including instance property branch, objectness prediction module and positive sample selection strategy. We implement ALA on FCOS [1] based on ResNet-50 [14]. We follow the standard $1\times$ training configuration [39] with single training and testing strategy on MS COCO training set. The training epoch, batch size, momentum and weight_decay are set to be 12, 4, 0.9 and 0.0001, respectively. Stochastic Gradient Descent (SGD) is selected as optimizer following the default configuration in MMDetection [39]. The initial learning rate is 0.01 and it will decay to 0.001 and 0.0001 when the epoches become 8 and 11 during training process, respectively. The performance is evaluated on the validation set of MS COCO.

TABLE III
RESULTS OF ALA WITH CONFIDENCE ($P_i^{\star|\star\in(+,-)}$) AND WEIGHT ($\omega_i^{\star|\star\in(+,-)}$) MECHANISMS IN OBJECTNESS PREDICTION MODULE ON THE VALIDATION SET OF MS COCO.

Confidence	Weight	$AP_{0.5:0.95}$	$AP_{0.5}$	$AP_{0.75}$
Baseline		37.0	55.8	39.7
✓	✗	38.9	57.9	41.8
✗	✓	10.5	28.4	19.7
✓	✓	**40.9**	**61.3**	**44.6**

TABLE IV
RESULTS OF ALA WITH DIFFERENT CLASSIFICATION ($P(cls)$) AND LOCALIZATION ($P(loc)$) CONFIGURATIONS IN CONFIDENCE MECHANISM OF OBJECTNESS PREDICTION MODULE ON THE VALIDATION SET OF MS COCO.

$P(cls)$	$P(loc)$	$AP_{0.5:0.95}$	$AP_{0.5}$	$AP_{0.75}$
Baseline		37.0	55.8	39.7
✓	✗	27.5	44.8	30.1
✗	✓	30.4	47.5	33.6
✓	✓	**40.9**	**61.3**	**44.6**

1) Instance Property Branch: For guiding the label assignment in object detection, we embed an instance property branch for taking advantages of the appearance information in an end-to-end learning pipeline. As shown in Table I, we evaluate the contribution of our IPB and compare it with another two counterparts, Center-ness branch in FCOS [1] and $ImpObj$ in AutoAssign [10]. Center-ness helps suppress the low-quality detected bounding boxes and underline the high-quality ones by calculating the relative distances between the location for each sample and the center point of the object. $ImpObj$ is deployed for emphasizing the proper positive samples and filtering out the noise candidates during the label assignment. Despite instance property branch shares similar designation insight with $ImpObj$, it provides an explicit prediction to clarify the foreground from the background and can be adaptively adjusted according to the shape and scale across different instances. Clearly, Our IPB can outperform its two counterparts over 1.1% and 0.8% gains, and achieve 40.9% in terms of $IoU = [0.5 : 0.95]$. Besides, the performance declines to 39.5% when we discard IPB in our method. Above improvement verifies that the instance property branch can provide a prior distribution for the appearance of the objects by separating the foreground from the background.

Furthermore, we conduct additional experiments to verify the concept of this module. Since the purpose of instance property branch is to separate the foreground samples from the background ones, the separation process can be treated as a two-class segmentation task, which contains foreground and background. We deploy the standard evaluation metric IoU in segmentation task, which represents the ratio of the intersection and the union between the prediction set and label one, to evaluate the separation accuracy of the foreground and background. The labels for the foregrounds and backgrounds are obtained from the annotations in the dataset. As shown in Table II, we represent the results under two different conditions, which are before introducing $C(\vec{d}_i|\vec{\mu},\vec{\sigma})$ and $G(\gamma,\sigma^2)$ (*i.e.*,

TABLE V
RESULTS (MAP UNDER $IoU = [0.5 : 0.95]$) OF ALA WITH DIFFERENT VALUES OF α AND β IN CONFIDENCE MECHANISM OF OBJECTNESS PREDICTION MODULE ON THE VALIDATION SET OF MS COCO.

Method	β	α 0.2	0.4	0.6	0.8	1	1.2	1.4
Baseline	–				37.0			
MUL	0	10.5	10.5	10.5	10.5	10.5	10.5	10.5
	0.2	38.9	39.4	39.6	40.1	40.0	39.9	39.5
	0.4	39.9	40.1	40.2	40.4	40.4	40.1	39.8
	0.6	40.2	40.5	40.6	**40.9**	40.8	40.2	39.5
	0.8	39.5	39.8	40.1	40.2	40.3	40.0	39.8
	1	38.9	39.0	39.4	39.5	39.1	38.8	38.7
ADD	0	9.8	9.8	9.8	9.8	9.8	9.8	9.8
	0.2	38.5	39.0	39.9	40.0	39.8	39.5	39.4
	0.4	39.8	39.9	40.1	40.1	40.2	40.0	39.7
	0.6	40.0	40.2	40.5	40.5	40.6	40.2	39.9
	0.8	39.8	40.0	40.3	40.4	40.4	40.3	40.0
	1	39.5	39.8	40.1	40.2	40.1	39.8	39.5

w/o C and G) and after introducing $C(\vec{d}_i|\vec{\mu},\vec{\sigma})$ and $G(\gamma,\sigma^2)$ (*i.e.*, with C and G). Clearly, after introducing C and G, the separation results of the foreground and background can achieve ∼90%. The experiment results verify the concept of the proposed instance property branch, which can effectively distinguish the foreground samples from the background ones. After combining with C and G, it takes full advantages of the appearances of each instance and is beneficial for the label assignment in object detection task.

2) Objectness Prediction Module: In this subsection, we assess the effectiveness of the proposed objectness prediction module, which contains confidence and weight mechanisms. Specifically, confidence mechanism holds the quality information of the samples and weight mechanism excavates the probability of the positive or negative. As illustrated in Table III, we only obtain 10.5% when only weight mechanism is considered in objectness prediction module, as the classification and localization predictions are not applied in the quality score of the samples in Eqn. (7) and these two branches will not be optimized during the training process. When only exploiting the confidence mechanism, the performance is 38.9%, indicating that the prior probability is critical for guiding the assignment. After confidence and weight mechanisms are both conducted, we can obtain 40.9%, outperforming the baseline over 3.9% gains. The improvement suggests that our objectness prediction module can take full advantage of the appearance and shape information for guiding the label assignment in object detection, where such prior distribution of the objects can be fully adopted in our method.

We further explore the different configurations in center and weight mechanisms and assess the corresponding performance. We firstly evaluate the influence of classification confidence $P(cls)$ and localization accuracy $P(loc)$ in the confidence mechanism in Eqn. (3). As elaborated in Table IV, using one of these two prediction results can not obtain satisfactory results and the performance in the first and second rows are even lower than the baseline (27.5/30.4 vs 37.0). After combining them together, we achieve the highest performance. The results show that both of the classification and localization should be jointly considered in the confidence mechanism for measuring the quality of the samples.

TABLE VI
RESULTS OF ALA WITH CENTER WEIGHTING ($\mathcal{C}$) AND SHAPE WEIGHTING (G) IN WEIGHT MECHANISM OF OBJECTNESS PREDICTION MODULE ON THE VALIDATION SET OF MS COCO.

$\mathcal{C}$	G	$AP_{0.5:0.95}$	$AP_{0.5}$	$AP_{0.75}$
Baseline		37.0	55.8	39.7
✗	✗	39.5	59.8	43.0
✓	✗	40.3	60.8	44.0
✗	✓	40.0	60.3	43.6
✓	✓	**40.9**	**61.3**	**44.6**

Besides, we implement the cross validation for determining the hyper-parameters α and β in Eqns. (2) and (3). Based on Eqn. (3) (denoted as 'MUL'), we also design another confidence definition as $P_i^+ = \beta * P_i(cls|\theta) + (1-\beta)P_i(loc|\theta)$, which is simplified as 'ADD'. The comparison results are shown in Table V. When the value of β is set to be 0 or 1, it represents that only classification confidence or localization accuracy is taken into consideration during calculating the confidence of the samples, respectively. Clearly, using both of the classification confidence and localization accuracy information outperforms the counterparts which only consider one of these predictions during calculating the confidence. We obtain the best performance when α and β equal to 0.8 and 0.6, respectively, where this configuration is utilized in the following experiments.

Furthermore, we evaluate the contributions of center and shape weighting in the weight mechanism. Center weighting focuses on the regions which are around the center of the object and provides more weights for these regions, because they have more probability to be positive samples. Gaussian blur weighting is deployed for smoothing the boundary of the instance property information. We demonstrate the experimental results in Table VI. Each of these weighting methods can improve the performance and the combination can obtain 40.9%. Moreover, it is superior over the one which utilizes neither of the weighting methods over 1.4% (39.5% vs 40.9%).

At last, we analyse different value setting configurations for $\vec{\mu}$ and $\vec{\sigma}$ of the center weighting in Eqn. (6). As shown in Table VII, 'Fixed' means that the values are not changed during the training process. 'Shared' represents that all the categories in the dataset share the same value of $\vec{\mu}$ and $\vec{\sigma}$, which are obtained through the network in Figure 4 and the output dimension changes to $(N, 1)$. Clearly, our method obtains 0.4% and 0.3% gain compared with 'Fixed' and 'Shared' methods, respectively. The improvement attributes to the specific ability of our method to wisely fit for the center regions of different categories.

3) Positive Sample Selection Strategy: Finally, we delve into the effectiveness of the positive sample selection strategy, which can perceive the statistical information of the positive samples and carefully distinguish and optimize the positive samples with different labels. We firstly compare the results with or without PS^3 in our ALA and illustrate the performance in Table VIII. Obviously, PS^3 can obtain extra 1.0% gains

TABLE VII
RESULTS OF ALA WITH DIFFERENT $\vec{\mu}$ AND $\vec{\sigma}$ SETTINGS IN CENTER WEIGHTING OF OBJECTNESS PREDICTION MODULE ON THE VALIDATION SET OF MS COCO.

Method	$AP_{0.5:0.95}$	$AP_{0.5}$	$AP_{0.75}$
Baseline	37.0	55.8	39.7
Fixed	40.5	60.8	44.0
Shared	40.6	61.0	44.1
Ours	**40.9**	**61.3**	**44.6**

TABLE VIII
RESULTS OF ALA WITH OR WITHOUT POSITIVE SAMPLE SELECTION STRATEGY (PS^3) ON THE VALIDATION SET OF MS COCO.

Method	$AP_{0.5:0.95}$	$AP_{0.5}$	$AP_{0.75}$
Baseline	37.0	55.8	39.7
ALA w/o PS^3	39.9	59.9	43.8
ALA with PS^3	**40.9**	**61.3**	**44.6**

TABLE IX
RESULTS OF ALA WITH DIFFERENT POSITIVE SAMPLE THRESHOLD METHODS IN PS^3 ON THE VALIDATION SET OF MS COCO.

Method		$AP_{0.5:0.95}$	$AP_{0.5}$	$AP_{0.75}$
Baseline		37.0	55.8	39.7
Fixed	0.3	40.0	60.2	43.8
	0.4	40.2	60.5	44.2
	0.5	40.3	60.7	44.3
	0.6	40.5	61.0	44.4
	0.7	40.2	60.4	44.3
Ours		**40.9**	**61.3**	**44.6**

TABLE X
RESULTS OF ALA WITH DIFFERENT SOFT LABEL ξ METHODS IN PS^3 ON THE VALIDATION SET OF MS COCO.

Method		$AP_{0.5:0.95}$	$AP_{0.5}$	$AP_{0.75}$
Baseline		37.0	55.8	39.7
Fixed	0.3	38.9	58.0	41.9
	0.4	39.5	59.9	43.1
	0.5	39.9	60.5	43.5
	0.6	40.4	60.9	43.9
	0.7	40.5	61.1	44.1
	0.8	40.1	60.3	44.0
	0.9	40.0	60.3	43.9
	1.0	39.7	60.1	43.2
Ours		**40.9**	**61.3**	**44.6**

TABLE XI
RESULTS (%) OF ALA WITH DIFFERENT LOSSES ON THE VALIDATION SET OF MS COCO.

ℓ_{R^H}	ℓ_{R^S}	ℓ_{R^-}	$AP_{0.5:0.95}$	$AP_{0.5}$	$AP_{0.75}$
Baseline			37.0	55.8	39.7
✓	✓	✗	23.5	39.7	25.5
✓	✗	✓	39.9	59.9	39.7
✗	✓	✓	18.7	33.8	19.8
✓	✓	✓	**40.9**	**61.3**	**44.6**

over its counterpart, and show its potential ability to improve the object detection performance by further considering the quality and intrinsical information of the positive samples. Then, we explore the impact of the positive sample threshold

TABLE XII
RESULTS (%) OF ALA WITH DIFFERENT SOFT LABEL STRATEGIES ON THE VALIDATION SET OF MS COCO.

methods	$AP_{0.5:0.95}$	$AP_{0.5}$	$AP_{0.75}$
Baseline	37.0	55.8	39.7
PNLP [40]	40.2	60.5	43.8
Hu. *et.al* [41]	40.4	60.8	43.9
PS^3	**40.9**	**61.3**	**44.6**

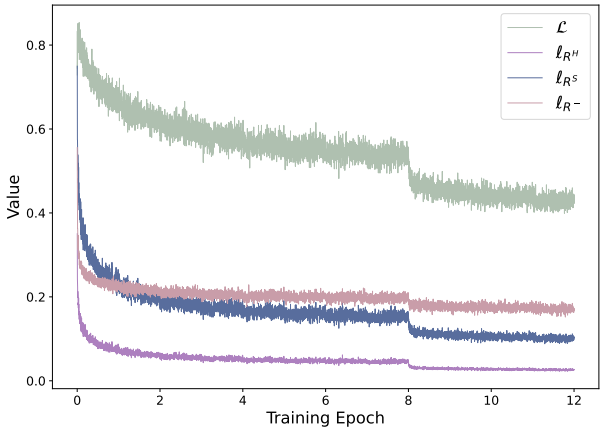


Fig. 5. Visualization of the training progress of ALA including $\mathcal{L}$, ℓ_{R^H}, ℓ_{R^S} and ℓ_{R^-}.

in Alg. 1. Our PS^3 conducts an adjusted method which decides the threshold depending on the average and standard deviation values in the positive sample set. As shown in Table IX, we achieve the best performance compared to the fixed positive sample threshold setting strategy. Furthermore, we also make a comparison for the soft label setting in Eqn. (10) with the fixed one. Similar to Table IX, we achieve the best performance among different values of fixed strategy in Table X. Above results verify the effectiveness of our positive sample selection strategy.

We also give the ablation study to explore the impacts of these three losses following the same configuration in Table VIII and show the results in Table XI. The results indicate that each loss can contribute to the improvement of the performance. It is worthy noting that if we do not deploy ℓ_{R^S} in the framework, it will change to the traditional sample selection manner, where all the samples are classified into only two parts, positive and negative.

For further evaluating the effectiveness of the proposed positive sample selection strategy, we compare it to the another two methods [40], [41], which both handle the label assignment issue using soft labels based on the graph structure. Specifically, PNLP [40] reformulates the label propagation framework and adds additional constraint information during the negative sample propagation process, where the i-th sample dose not have the k-th label. Besides, it can also address the issue where the negative samples are not inherent from the data structure. Hu. *et.al* [41] introduce soft label to the human activities recognition task, allowing the annotators to assign multiple and weighted labels to the sequential data. The latent variables are utilized to consider the relationship

TABLE XIII
COMPARISON OF ALA USING RESNET-101, RESNEXT-101, DCN-RESNET-101 AND DCN-RESNEXT-101 WITH STATE-OF-THE-ART METHODS ON THE *test-dev* SET OF MS COCO. ***Static*** AND ***Dynamic*** REPRESENT STATIC AND DYNAMIC LABEL ASSIGNMENT STRATEGIES, RESPECTIVELY.

Method	Publication	Backbone	$AP_{0.5:0.95}$	$AP_{0.5}$	$AP_{0.75}$	AP_S	AP_M	AP_L	FPS	Mem(GB)
Static										
FASF [42]	CVPR2020		40.9	61.5	44.0	24.0	44.2	51.3	11.8	5.1
MuSu [43]	ICCV2021		44.8	63.2	49.1	26.2	47.9	56.4	11.5	5.6
DIAG-IOU [28]	TCSVT2023		39.4	58.6	42.2	22.7	43.6	51.1	12.5	**4.6**
G-RFA [18]	TCSVT2023		46.8	65.4	50.8	27.5	50.0	**58.8**	10.7	6.4
Cross-Split [44]	TCSVT2023		48.3	**65.6**	53.3	33.2	51.4	56.2	11.0	5.8
Dynamic										
FreeAnchor [22]	NIPS2019		43.1	62.2	46.4	24.5	46.1	54.8	**12.9**	6.8
ATSS [7]	CVPR2020		43.6	62.1	47.4	26.1	47.0	53.6	12.3	5.6
PAA [8]	ECCV2020	ResNet-101	44.8	63.3	48.7	26.5	48.8	56.3	6.2	9.8
SAPD [23]	ECCV2020		43.5	63.6	46.5	24.9	46.8	54.6	6.4	9.7
AutoAssign [10]	ArXiv2020		44.5	64.3	48.4	25.9	47.4	55.0	11.2	5.5
OTA [9]	CVPR2021		45.3	63.5	49.3	26.9	48.8	56.1	6.4	11.7
IQDet [2]	CVPR2021		45.1	63.4	49.3	26.7	48.5	56.6	8.7	7.4
Pseudo-IoU [24]	CVPR2021		41.5	61.0	44.5	24.1	44.6	51.9	10.9	5.7
ObjectBox [45]	ECCV2022		46.1	65.0	48.3	26.0	48.7	57.3	10.7	6.1
OPA [26]	TMM2022		44.3	61.6	48.3	–	–	–	10.8	6.0
O2F [46]	CVPR2023		40.9	–	–	–	–	–	11.4	5.8
ALA			**48.4**	**65.6**	**53.8**	**33.4**	**52.0**	57.9	11.9	4.7
Static										
FASF [42]	CVPR2020		42.9	63.8	46.3	26.6	46.2	52.7	9.6	9.4
VFNet [47]	CVPR2021		46.7	65.2	50.8	28.3	50.1	57.3	9.7	8.9
G-RFA [18]	TCSVT2023		47.2	**66.7**	51.4	**29.4**	50.4	58.2	9.5	7.8
Dynamic										
ATSS [7]	CVPR2020	ResNeXt-101	45.1	63.9	49.1	27.9	48.2	54.6	**10.4**	6.9
AutoAssign [10]	ArXiv2020		46.5	66.5	50.7	28.3	49.7	56.6	10.1	7.1
SAPD [23]	ECCV2020		45.4	65.6	48.9	27.3	48.7	56.8	5.7	10.5
IQDet [2]	CVPR2021		47.0	65.7	51.1	29.1	50.5	57.9	7.7	8.5
OTA [9]	CVPR2021		47.0	65.8	51.1	29.2	50.4	57.9	5.1	13.1
Pseudo-IoU [24]	CVPR2021		44.0	63.8	47.3	28.0	47.5	58.1	10.0	7.4
ALA			**47.9**	66.1	**51.9**	28.8	**50.9**	**58.9**	**10.4**	**6.8**
Static										
FPN [5]	ICCV2017		40.8	63.2	44.6	–	–	–	7.9	11.5
FCOS [1]	ICCV2019		45.6	64.6	49.8	28.2	49.7	58.8	**10.1**	8.8
BorderDet [48]	ECCV2020		47.2	**66.1**	51.0	28.5	50.2	59.9	8.4	11.7
Dynamic R-CNN [49]	ECCV2020		44.9	63.8	49.0	25.8	47.5	57.5	7.4	12.0
MuSu [43]	ICCV2021	DCN-ResNet-101	47.4	65.0	**51.8**	27.8	50.5	60.0	9.8	9.1
Dynamic										
ATSS [7]	CVPR2020		46.3	64.7	50.4	27.7	49.8	58.4	9.7	7.4
SAPD [23]	ECCV2020		46.0	65.9	49.6	26.3	49.2	59.6	5.8	10.7
PAA [8]	ECCV2020		47.4	65.7	51.6	27.9	**51.3**	**60.6**	4.7	11.4
Pseudo-IoU [24]	CVPR2021		43.4	63.2	46.8	25.8	47.8	56.7	9.7	7.4
ALA			**48.0**	66.0	**51.8**	**28.9**	51.2	60.1	9.9	**7.1**
FCOS [1]	ICCV2019		46.4	65.7	50.1	**30.5**	49.4	60.4	7.7	12.5
AutoAssign [10]	ArXiv2020		48.3	67.4	52.7	29.2	51.0	60.3	**8.0**	11.4
ATSS [7]	CVPR2020		47.7	66.6	52.1	29.3	50.8	59.7	7.9	11.8
SAPD [23]	ECCV2020		47.4	67.4	51.1	28.1	50.3	61.5	4.0	14.4
PAA [8]	ECCV2020		49.0	67.8	53.3	30.2	52.8	62.2	3.8	15.1
OTA [9]	CVPR2021	DCN-ResNeXt-101	49.2	67.6	53.5	30.0	52.5	**62.3**	4.0	15.7
IQDet [2]	CVPR2021		49.0	67.5	53.1	30.0	52.3	62.0	5.9	12.7
VFNet [47]	CVPR2021		49.2	67.8	**53.6**	30.0	52.6	62.1	7.7	12.2
PSS [32]	TMM2023		47.5	–	–	–	–	–	7.8	11.7
ALA			**49.3**	**67.9**	53.5	29.8	**52.9**	62.2	**8.0**	**11.2**

between data and improve the expressiveness of the model. We use the sample selection strategies in [40], [41] to replace the proposed PS^3 and report the comparison results in Table XII, which follow the same training configuration in Table XI. Our proposed method achieves better competitive performance over another two counterparts. The reasons may owe to that the constraint information introduced in PNLP [40] performs well in multi label assignment process, and the label assignment in ALA only has two categories (*i.e.*, foreground and background). While [41] mainly focuses on sequential data, and it is not suitable for discrete label assignment in object detection.

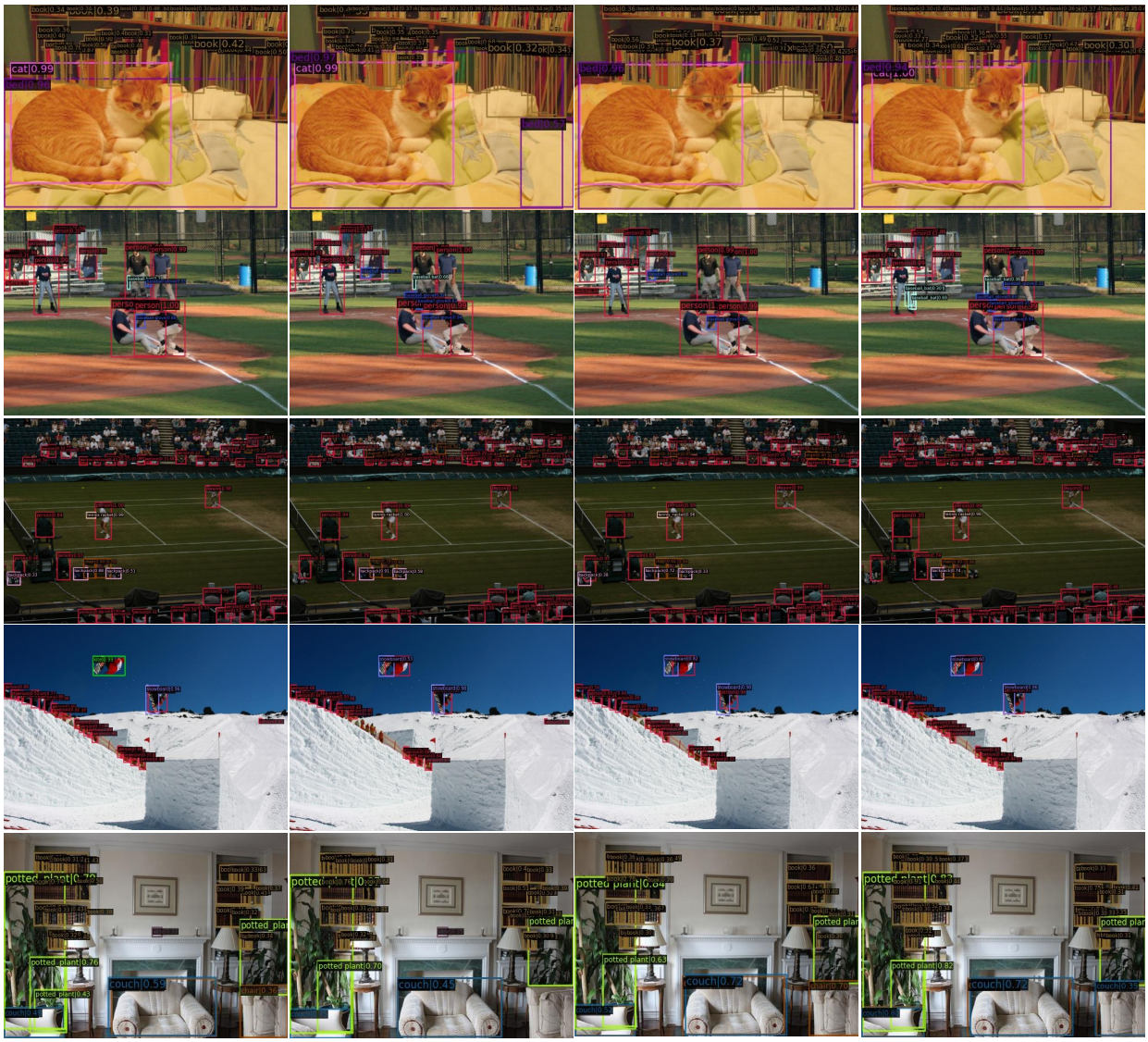
Furthermore, we plot the training progress including the training losses $\mathcal{L}$, ℓ_{R^H}, ℓ_{R^S} and ℓ_{R^-} of our method in Figure 5.

C. Comparison with State-of-the-Arts

In this section, we implement ALA on one-stage detector FCOS [1] using various backbone models (*e.g.*, ResNet [14], ResNeXt [13] and DCN [15]). All comparison methods are trained under 2× configuration [39] with multi-scale training strategy. The batch size, momentum, weight_decay and optimizer are the same as the ones in ablation study. The training epoch for standard 2× training configuration is 24. The initial learning rate is 0.01 and it will decay to 0.001 and 0.0001 when the epoches become 16 and 22 during training process, respectively. The performance is evaluated on the $test-dev$ set of MS COCO [11]. As exhibited in Table XIII and XIV, we respectively compare our ALA with its counterparts. For ResNet-101, we obtain the best performance and outperform other dynamic label assignment

TABLE XIV
COMPARISON OF ALA USING MULTI-SCALE TESTING STRATEGY WITH STATE-OF-THE-ART METHODS ON THE *test-dev* SET OF MS COCO.

Method	Publication	Backbone	$AP_{0.5:0.95}$	$AP_{0.5}$	$AP_{0.75}$	AP_S	AP_M	AP_L	FPS	Mem(GB)
AutoAssign [10]	ArXiv2020	DCN-ResNeXt-101	52.1	69.6	**58.0**	33.9	54.0	64.0	8.0	11.4
ATSS [7]	CVPR2020		50.6	68.6	56.1	33.6	52.9	62.2	7.9	11.8
PAA [8]	ECCV2020		51.3	68.8	56.6	34.3	53.5	63.6	3.8	15.1
IQDet [2]	CVPR2021		51.6	68.7	57.0	**34.5**	53.6	64.5	5.9	13.7
OTA [9]	CVPR2021		51.5	68.6	57.1	34.1	53.7	64.1	4.0	15.7
TOOD [50]	ICCV2021		51.1	69.4	55.5	31.9	54.1	63.7	6.4	12.1
ALA		ResNet-101	48.9	66.6	56.4	31.1	52.2	61.0	**11.9**	**4.7**
ALA		DCN-ResNeXt-101	**52.3**	**70.1**	57.9	33.4	**54.8**	**64.8**	8.0	11.2



(a) FCOS (b) ATSS (c) AutoAssign (d) Ours

Fig. 6. Qualitative comparisons of our ALA with FCOS [1], ATSS [7] and AutoAssign [10]. Our method shows better detection performance in those small and overlapped objects.

methods AutoAssign [10] and IQDet [2] over 1.7% (44.5% *vs* 46.2%) and 1.1% (45.1% *vs* 46.2%) gains, respectively. It is worthy noting that our ALA obtains additional 1.6% and 0.1% gains over the recently proposed static label assignment methods G-RFA [18] and Cross-Split [44]. For ResNeXt-101 and DCN-ResNet-101, ALA also achieves consistent improvement over its counterparts. Specifically, ALA is clearly superior over other dynamic methods, *e.g.*, OTA [9], PAA [8] and SAPD [23]. Compared to static methods G-RFA [18], ALA achieves 0.7% gain on ResNeXt-101. Our method acquires

TABLE XV
STATISTICAL RESULTS (AVERAGE SCORE μ AND VARIANCE σ) FOR ALA USING DIFFERENT BACKBONE MODELS UNDER 10 TIMES WITH DIFFERENT SEEDS. $\star$ INDICATES USING MULTI-SCALE TESTING STRATEGY.

Backbone	Statistics	$AP_{0.5:0.95}$	$AP_{0.5}$	$AP_{0.75}$	AP_S	AP_M	AP_L
ResNet-101	μ	48.4	65.6	53.8	33.4	52.0	57.9
	σ	0.011	0.005	0.015	0	0.011	0.003
ResNeXt-101	μ	48.9	67.1	52.9	33.8	51.9	59.9
	σ	0.012	0.008	0.009	0.007	0.005	0.007
DCN-ResNet-101	μ	48.0	66.0	51.8	28.9	51.2	60.1
	σ	0.009	0.004	0.008	0.012	0.008	0.005
DCN-ResNeXt-101	μ	49.3	67.9	53.5	29.8	52.9	62.2
	σ	0.010	0.011	0.007	0.011	0.007	0.009
ResNet-101*	μ	48.9	66.6	56.4	31.1	52.2	61.0
	σ	0.005	0.004	0.005	0.008	0.011	0.007
DCN-ResNeXt-101*	μ	52.3	70.1	57.9	33.4	54.8	64.8
	σ	0	0.011	0.004	0.011	0.008	0.0012

TABLE XVI
COMPARISON OF ALA WITH STATE-OF-THE-ART METHODS USING RESNET-50 ON THE PASCAL VOC 2007 AND 2012 DATASETS.

Method		FCOS [1]	ATSS [7]	AutoAssign [10]	OTA [9]	OPA [26]	PSS [32]	O2F [46]	ALA
mAP	PASCAL VOC 2007	87.1	87.6	88.7	88.6	88.6	89.1	89.4	**90.1**
	PASCAL VOC 2012	83.4	84.1	85.2	85.0	85.1	85.5	85.9	**87.4**

similar performance gains on DCN-ResNeXt-101, and it is competitive to state-of-the-arts, *e.g.*, PSS [32]. When multi-scale testing strategy is applied, we obtain 48.9% and 52.3% on ResNet-101 and DCN-ResNeXt-101, respectively, and it is superior over state-of-the-arts methods. Above gains obviously confirm the effectiveness of our ALA, which has the ability to improve the detection performance and can be well adopted to different backbone models. We deploy all the experiments on official MMDetection benchmark [39] and utilize the standard $1\times$ training configuration for ablation study and $2\times$ training configuration for comparison with state-of-the-arts methods. As we follow these two default training configurations in the official code base, where these configurations have been verified as the optimal optimization solution for training, we do not meet with overfitting issue in our experiments.

To demonstrate the evidence of the statistically significant between methods, we run the proposed method 10 times by setting different seeds in Table XIII and XIV, and calculate the statistical results (*i.e.*, average score μ and variance σ) for each evaluation metric. The results are reported in Table XV. We achieve stable results in each evaluation metric and the variances σ of different backbones are $\sim$0.01.

$AP_{0.5:0.95}$ evaluation metric, which calculates the average value across all 10 IoU thresholds and all 80 categories, is the primary challenging metric in MS COCO dataset [11]. It is considered as the single most important metric for determining the challenging winner[1]. As shown in Table XIII and XIV, our proposed method ALA achieves the best performance in this evaluation metric based on different backbone models (*e.g.*, ResNet-101, ResNeXt-101 and DCN-ResNet-101) compared to the static approaches. The results indicate that our proposed dynamic label assignment method ALA can address the limitations in those static ones. Besides, in other evaluation metrics, ALA also obtains very competitive performance. For example, for ResNet-101, we achieve the best performance in $AP_{0.5}$, $AP_{0.75}$, AP_S and AP_M, and gain the second best performance in AP_L. ALA also performs similar gains over other static methods in $AP_{0.75}$, AP_M and AP_L on ResNeXt-101 and in $AP_{0.75}$ and AP_S on DCN-ResNet-101 backbones. Above results demonstrate that our proposed method can take full advantages of the dynamic label assignment process and is beneficial for the performance improvement in object detection task.

To further validate the effectiveness of our ALA, we qualitatively evaluate our proposed ALA on MS COCO dataset and compare the results with FCOS [1], ATSS [7] and AutoAssign [10]. The visualization comparison results are shown in Figure 6. Obviously, ALA demonstrates better detection ability for those small size and overlapped objects (*e.g.*, book and person).

We further evaluate our proposed method on another two benchmarks PASCAL VOC 2007 and PASCAL VOC 2012 [12]. For PASCAL VOC 2007, we follow the same training configuration in [1], where the networks are trained on the union sets of training/validation in PASCAL VOC 2007 and PASCAL VOC 2012, and the results are reported on the test set of PASCAL VOC 2007 for comparison. For PASCAL VOC 2012, we train the networks on the training and validation sets of PASCAL VOC 2007 and PASCAL VOC 2012 with additional test set of PASCAL VOC 2007, and evaluate the performance on the test set of PASCAL VOC 2012. We utilize the mean Average Precision (mAP), which is the standard evaluation metric in PASCAL VOC to measure different detectors. We deploy ResNet-50 [14] as the backbone model and show the experiment results in Table XVI. Similar as MS COCO, our proposed ALA achieves the best performance over its counterparts on these two detection datasets, and the improvement verifies again the effectiveness of our method.

[1]https://cocodataset.org/#detection-eval

TABLE XVII
COMPARISON OF ALA WITH STATE-OF-THE-ART ONE-TO-ONE LABEL ASSIGNMENT METHODS USING RESNET-50 AND RESNET-101 ON THE *test-dev* SET OF MS COCO.

Method	Backbone	$AP_{0.5:0.95}$	$AP_{0.5}$	$AP_{0.75}$	FPS	Mem(GB)
DETR [29]	ResNet-50	42.0	62.4	44.2	8.7	7.9
Sparse R-CNN [31]		42.3	61.2	**45.7**	14.2	5.3
POTO [33]		42.3	–	–	14.9	4.2
PSS [32]		44.0	–	–	14.5	4.9
ALA		**44.4**	**66.5**	44.3	**15.4**	**3.8**
DETR [29]	ResNet-101	43.5	63.8	46.4	5.3	10.2
Sparse R-CNN [31]		43.5	62.1	**47.2**	11.4	7.1
POTO [33]		42.9	–	–	11.8	5.2
PSS [32]		44.2	–	–	11.5	6.7
ALA		**45.4**	**66.8**	46.7	**11.9**	**4.7**

D. Generalization to One-to-One Label Assignment

ALA provides a many-to-one label assignment form, where serveral positive samples are assigned to the same ground truth. Recently, the one-to-one label assignment manner becomes an interesting direction in this topic, in which only one positive sample corresponds to the ground truth. One-to-one label assignment can eliminate the heuristic post-processing step None-Maximum-Suppression in inference process, making detectors end-to-end. For verifying the generalization ability of our ALA to different label assignment types, we extend ALA to one-to-one form. Concretely, we select the sample which has the highest quality score p in positive set $\mathcal{S}$ as the only one positive sample and regard all the rest samples as negative ones in Alg. 1. Following the same training and inference configuration in [33] for a fair comparison, we build ALA on FCOS and represent the results in Table XVII based on different backbone models. Surprisingly, we obtain very competitive performance with other methods on ResNet-50 and ResNet-101, which are carefully designed for accelerating the training speed (*e.g.*, removing redundant positive samples in POTO [33]). The results express that the generalization ability of our ALA, which can be effectively adopt to one-to-one label assignment style.

V. CONCLUSION

In this paper, we propose a novel automatic label assignment for object detection (ALA). We tackle the instance property and objectness prediction module, which can supply a coarse-to-fine description for the appearance of the objects. By combining them together, we can obtain the distributions for the positive and negative samples. Besides, they can be optimized a differentiable manner with classification and localization branches. Furthermore, a positive sample selection strategy is deployed to further optimize the positive samples with different labels for considering their quality score distribution. Extensive experiments demonstrate that our ALA can effectively improve the performance on various backbone models on MS COCO dataset, verifying the effectiveness of our proposed label assignment principle. Since the current designation of instance property branch is based on the feature pyramid network through convolution and concatenation, its architecture can be further improved by referring to the insights from segmentation task for better guiding the label assignment procedure. Besides, we will consider to excavate other more specific and accurate models to obtain the center weight in the future work.

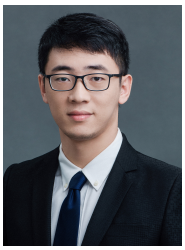
REFERENCES

[1] Z. Tian, C. Shen, H. Chen, and T. He, "FCOS: Fully convolutional one-stage object detection," in *ICCV*, 2019.
[2] Y. Ma, S. Liu, Z. Li, and J. Sun, "IQDet: Instance-wise quality distribution sampling for object detection," in *CVPR*, 2021.
[3] T.-Y. Lin, P. Goyal, R. Girshick, K. He, and P. Dollár, "Focal loss for dense object detection," in *ICCV*, 2017.
[4] S. Ren, K. He, R. Girshick, and J. Sun, "Faster R-CNN: Towards real-time object detection with region proposal networks," in *NIPS*, 2015.
[5] T.-Y. Lin, P. Dollár, R. B. Girshick, K. He, B. Hariharan, and S. J. Belongie, "Feature pyramid networks for object detection," in *CVPR*, 2017.
[6] T. Kong, F. Sun, H. Liu, Y. Jiang, L. Li, and J. Shi, "Foveabox: Beyound anchor-based object detection," *IEEE T-IP*, 2020.
[7] 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 *CVPR*, 2020.
[8] K. Kim and H. S. Lee, "Probabilistic anchor assignment with iou prediction for object detection," in *ECCV*, 2020.
[9] Z. Ge, S. Liu, Z. Li, O. Yoshie, and J. Sun, "OTA: Optimal transport assignment for object detection," in *CVPR*, 2021.
[10] B. Zhu, J. Wang, Z. Jiang, F. Zong, S. Liu, Z. Li, and J. Sun, "Autoassign: Differentiable label assignment for dense object detection," *arXiv preprint arXiv:2007.03496*, 2020.
[11] T.-Y. Lin, M. Maire, S. Belongie, J. Hays, P. Perona, D. Ramanan, P. Dollár, and C. L. Zitnick, "Microsoft COCO: Common objects in context," in *ECCV*, 2014.
[12] M. Everingham, S. A. Eslami, L. Van Gool, C. K. Williams, J. Winn, and A. Zisserman, "The pascal visual object classes challenge: A retrospective," *IJCV*, 2015.
[13] S. Xie, R. Girshick, P. Dollár, Z. Tu, and K. He, "Aggregated residual transformations for deep neural networks," in *CVPR*, 2017.
[14] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *CVPR*, 2016.
[15] J. Dai, H. Qi, Y. Xiong, Y. Li, G. Zhang, H. Hu, and Y. Wei, "Deformable convolutional networks," in *ICCV*, 2017.
[16] J. Nie, Y. Pang, S. Zhao, J. Han, and X. Li, "Efficient selective context network for accurate object detection," *IEEE T-CSVT*, 2020.
[17] J. Cao, Y. Pang, S. Zhao, and X. Li, "High-level semantic networks for multi-scale object detection," *IEEE T-CSVT*, 2019.
[18] F. Gao, Y. Cai, F. Deng, C. Yu, and J. Chen, "Feature alignment in anchor-free object detection," *IEEE T-CSVT*, 2023.
[19] K. Duan, S. Bai, L. Xie, H. Qi, Q. Huang, and Q. Tian, "Centernet: Keypoint triplets for object detection," in *ICCV*, 2019.
[20] J. Wang, K. Chen, S. Yang, C. C. Loy, and D. Lin, "Region proposal by guided anchoring," in *CVPR*, 2019.
[21] T. Yang, X. Zhang, Z. Li, W. Zhang, and J. Sun, "Metaanchor: Learning to detect objects with customized anchors," in *NIPS*, 2018.
[22] X. Zhang, F. Wan, C. Liu, R. Ji, and Q. Ye, "Freeanchor: Learning to match anchors for visual object detection," in *NIPS*, 2019.
[23] C. Zhu, F. Chen, Z. Shen, and M. Savvides, "Soft anchor-point object detection," in *ECCV*, 2020.

[24] J. Li, B. Cheng, R. Feris, J. Xiong, T. S. Huang, W.-M. Hwu, and H. Shi, "Pseudo-iou: Improving label assignment in anchor-free object detection," in *CVPR*, 2021.
[25] S. Zhang, L. Wen, Z. Lei, and S. Z. Li, "Refinedet++: Single-shot refinement neural network for object detection," *IEEE T-CSVT*, 2020.
[26] Y. Yang, X. Sun, W. Diao, X. Rong, S. Yan, D. Yin, and X. Li, "Optimal partition assignment for universal object detection," *IEEE T-MM*, 2022.
[27] W. Xiao, Y. Peng, C. Liu, J. Gao, Y. Wu, and X. Li, "Balanced sample assignment and objective for single-model multi-class 3d object detection," *IEEE T-CSVT*, 2023.
[28] S. Zhang, C. Li, Z. Jia, L. Liu, Z. Zhang, and L. Wang, "Diag-iou loss for object detection," *IEEE T-CSVT*, 2023.
[29] N. Carion, F. Massa, G. Synnaeve, N. Usunier, A. Kirillov, and S. Zagoruyko, "End-to-end object detection with transformers," in *ECCV*, 2020.
[30] X. Zhu, W. Su, L. Lu, B. Li, X. Wang, and J. Dai, "Deformable detr: Deformable transformers for end-to-end object detection," *arXiv preprint arXiv:2010.04159*, 2020.
[31] P. Sun, R. Zhang, Y. Jiang, T. Kong, C. Xu, W. Zhan, M. Tomizuka, L. Li, Z. Yuan, C. Wang *et al.*, "Sparse R-CNN: End-to-end object detection with learnable proposals," in *CVPR*, 2021.
[32] Q. Zhou and C. Yu, "Object detection made simpler by eliminating heuristic nms," *IEEE T-MM*, 2023.
[33] J. Wang, L. Song, Z. Li, H. Sun, J. Sun, and N. Zheng, "End-to-end object detection with fully convolutional network," in *CVPR*, 2021.
[34] K. Chen, J. Pang, J. Wang, Y. Xiong, X. Li, S. Sun, W. Feng, Z. Liu, J. Shi, W. Ouyang *et al.*, "Hybrid task cascade for instance segmentation," in *CVPR*, 2019.
[35] H. Wang, Q. Wang, H. Zhang, Q. Hu, and W. Zuo, "Crabnet: Fully task-specific feature learning for one-stage object detection," *IEEE T-IP*, 2022.
[36] H. Wang, T. Jia, B. Ma, Q. Wang, and W. Zuo, "Fully cascade consistency learning for one-stage object detection," *IEEE T-CSVT*, 2023.
[37] K. He, G. Gkioxari, P. Dollár, and R. Girshick, "Mask R-CNN," in *ICCV*, 2017.
[38] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei, "Imagenet: A large-scale hierarchical image database," in *CVPR*, 2009.
[39] K. Chen, J. Wang, J. Pang, Y. Cao, Y. Xiong, X. Li, S. Sun, W. Feng, Z. Liu, J. Xu *et al.*, "Mmdetection: Open mmlab detection toolbox and benchmark," *arXiv preprint arXiv:1906.07155*, 2019.
[40] O. Zoidi, A. Tefas, N. Nikolaidis, and I. Pitas, "Positive and negative label propagations," *IEEE T-CSVT*, 2016.
[41] N. Hu, G. Englebienne, Z. Lou, and B. Kröse, "Learning to recognize human activities using soft labels," *IEEE T-PAMI*.
[42] C. Zhu, Y. He, and M. Savvides, "Feature selective anchor-free module for single-shot object detection," in *CVPR*, 2019.
[43] Z. Gao, L. Wang, and G. Wu, "Mutual supervision for dense object detection," in *ICCV*, 2021.
[44] S.-Y. Wang, Z. Qu, and C.-J. Li, "A dense-aware cross-splitnet for object detection and recognition," *IEEE T-CSVT*, 2022.
[45] M. Zand, A. Etemad, and M. Greenspan, "Objectbox: From centers to boxes for anchor-free object detection," in *ECCV,*.
[46] S. Li, M. Li, R. Li, C. He, and L. Zhang, "One-to-few label assignment for end-to-end dense detection," in *CVPR*, 2023.
[47] H. Zhang, Y. Wang, F. Dayoub, and N. Sunderhauf, "Varifocalnet: An iou-aware dense object detector," in *CVPR*, 2021.
[48] H. Qiu, Y. Ma, Z. Li, S. Liu, and J. Sun, "Borderdet: Border feature for dense object detection," in *ECCV*, 2020.
[49] H. Zhang, H. Chang, B. Ma, N. Wang, and X. Chen, "Dynamic R-CNN: Towards high quality object detection via dynamic training," in *ECCV*, 2020.
[50] C. Feng, Y. Zhong, Y. Gao, M. R. Scott, and W. Huang, "Tood: Task-aligned one-stage object detection," in *ICCV*, 2021.

Hao Wang received the Ph.D. degree in the School of Computer Science and Technology, Harbin Institute of Technology, Harbin, China, in 2022. He currently joins the Northeastern University, Shenyang, China as a Lecturer with the College of Information Science and Engineering. His research interests include object detection, object segmentation and related problems.

Tong Jia received the B.E. degree in computer science and the Ph.D. degree in pattern identification and intelligent system from Northeastern University, China, in 1998 and 2008, respectively. From 2012 to 2013, he was an International Visiting Scholar with the Department of Electronic Engineering, Michigan State University, USA. He is a Chang Jiang Scholar Awarded Professor since 2023, and currently a Professor with the College of Information Science and Engineering, Northeastern University, Shenyang, China. His research interests include computer/machine vision, image processing, and pattern identification.



Qilong Wang (M'18) received the Ph.D. degree in the School of Information and Communication Engineering with Dalian University of Technology, China, in 2018. He is currently an Associate Professor with the College of Intelligence and Computing, Tianjin University. His research interests include computer vision and pattern recognition, particularly image/video classification, object detection, and deep probability distribution modeling. He has published more than 40 academic papers in top-tier conferences and referred journal including ICCV/CVPR/NIPS/ECCV/IJCAI and IEEE TPAMI/TIP/TCSVT.

Wangmeng Zuo (M'09-SM'14) received the Ph.D. degree in computer application technology from the Harbin Institute of Technology, Harbin, China, in 2007. From 2004 to 2006, he was a Research Assistant with the Department of Computing, The Hong Kong Polytechnic University, Hong Kong. From 2009 to 2010, he was a Visiting Professor with Microsoft Research Asia. He is currently a Professor with the School of Computer Science and Technology, Harbin Institute of Technology. He has published over 100 papers in top-tier academic journals and conferences. His current research interests include image enhancement and restoration, image/video generation, and object detection. He has served as a Tutorial Organizer in ECCV 2016, an Associate Editor of the IEEE Trans. Pattern Analysis and Machine Intelligence and a Senior Editor of Journal of Electronic Imaging.