Mutual Supervision for Dense Object Detection

Ziteng Gao Limin Wang[✉] Gangshan Wu
State Key Laboratory for Novel Software Technology, Nanjing University, China

Abstract

*The classification and regression head are both indispensable components to build up a dense object detector, which are usually supervised by the same training samples and thus expected to have consistency with each other for detecting objects accurately in the detection pipeline. In this paper, we break the convention of the same training samples for these two heads in dense detectors and explore a novel supervisory paradigm, termed as Mutual Supervision (*MuSu*), to respectively and mutually assign training samples for the classification and regression head to ensure this consistency. MuSu defines training samples for the regression head mainly based on classification predicting scores and in turn, defines samples for the classification head based on localization scores from the regression head. Experimental results show that the convergence of detectors trained by this mutual supervision is guaranteed and the effectiveness of the proposed method is verified on the challenging MS COCO benchmark. We also find that tiling more anchors at the same location benefits detectors and leads to further improvements under this training scheme. We hope this work can inspire further researches on the interaction of the classification and regression task in detection and the supervision paradigm for detectors, especially separately for these two heads.*

1. Introduction

Object detection has been drawing interest from researchers for decades as one of the fundamental visual tasks in the computer vision community, especially with the rise of convolutional neural networks (CNNs). The community has witnessed the fast evolution of both the methodology and the performance of detectors from region-based ones [8, 24, 10, 1, 4, 18], to one-stage dense ones [20, 23, 22, 29, 37, 32] and then to end-to-end transformer-based detectors [3, 41]. Among these methods, one-stage detectors, also known as dense detectors, are favored in terms of both the speed and accuracy, as well as the fast convergence due to their tiling anchors densely to cover objects of various scales and aspect ratios and directly predicting bounding boxes with labels with these anchors.

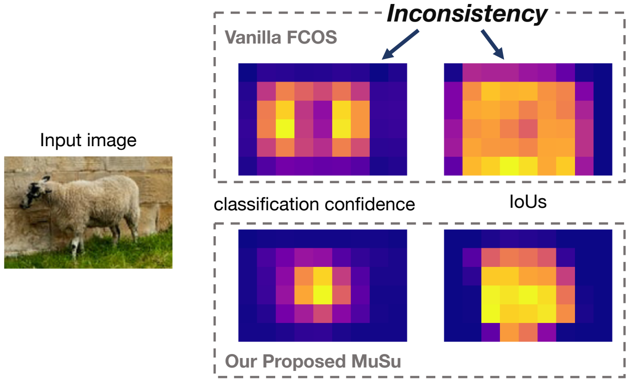

Figure 1. **Inconsistency from the classification head and regression head** between spatial distributions of classification confidence and IoUs with ground truth predicted in a converged FCOS detector and our MuSu-trained detector. The brighter the pixel looks, the higher the value stands for. The classification confidence is a product of the output of the classification head and centerness estimation as FCOS does in the NMS process. Note that this input image is a training image in MS COCO and the converged FCOS still suffers the inconsistency between classification and regression head. Our MuSu alleviates this inconsistency.

As detection task is about classifying and localizing simultaneously, object detectors are expected to produce bounding boxes with both correct classification labels and fine localization, and of course dense detectors are no exceptions. For a dense detector, these two tasks are usually done with specialized classification and regression heads. For the same input feature map from the backbone network, these two heads are expected to function differently: the classification head translates it into classification scores invariant with small shifts while the regression head transforms it to shift-equivalent localizing offsets from anchors to bounding boxes, which incurs intrinsic inconsistency between these two tasks.

An accurate dense object detector is supposed to produce high-quality bounding boxes with correct labels, which requires that these two heads of different functionalities coordinate at the same spatial location of final outputs. In other words, converged detectors should ensure spatial consis-

[✉]: Corresponding author (lmwang@nju.edu.cn).

tency on where the maximum classification and localizing scores appear for an object. However, even for a converged detector, this goal is hard to achieve and the maximum classification score and the most accurate localizing box for an object frequently appear at different locations for a training image as the input image depicted in Figure 1. This inconsistency hurts the performance of final models in the current detection pipeline, especially in the process of the common post-processing non-maximum suppression (NMS), which only keeps the box with the maximum classification score among overlapping ones without the consideration of localizing accuracy. As a result, bounding boxes with finer localization but lower classification scores are suppressed and such detectors lead to inferior performance.

To tackle this problem, previous work focuses on input features and network structures of these heads and disentangles the classification and regression heads from feature or structural perspectives. Different from those, we delve into this problem from the view of the supervision for these two heads, specifically, the definition of the training samples respectively for both them, and alleviate this inconsistency by proposing mutual supervision (MuSu) for dense detectors.

MuSu separates the definition of training samples for classification and regression head and then makes them dependent on each other mutually. As illustrated in Figure 2, training samples are not shared between two heads. Training targets for classification are adaptively determined by IoU (Intersection over Union) scores between predicted boxes and ground-truth boxes from the regression head. Alike, training samples for the regression head are defined by classifying scores from the classification head. Next, MuSu translates scores of these training samples for these two heads to soft targets by associating weights to losses of each spatial location. By this means, MuSu aims to force the consistency between these two heads by the mutual assignment in the training phase. Under this mutual supervision scheme, MuSu also enjoys the advantage of the training samples adaptively emerging from the network itself, which are refrained from being hand-crafted by expert knowledge. Moreover, MuSu is exempt from any hand-crafted geometric prior and also get rids of subtle treatments to different pyramid levels. In this sense, MuSu makes a big step further to fully adaptive sample assignment and unleashes the power of a detector more comprehensively.

We carry out extensive ablation experiments on MS COCO dataset [21] to validate the effectiveness of our proposed MuSu method. In particular, MuSu boosts the FCOS detector with ResNet-50 backbone to a 40.6 AP in the COCO validation set under the common 90k training scheme without the sacrifice of the inference speed. Moreover, we investigate that tiling more anchors at the same location will benefit the detector under this mutual supervision scheme, pushing to 40.9 AP over the competitive one-anchor counterpart. We argue that our method of mutual supervision for the classification and regression head exploits multiple anchor settings more fully and thus boosts the performance higher, in contrast to [35]. We also utilize MuSu to train models with large backbones to compare state-of-the-art models and our models achieve promising results on COCO `test-dev` set.

2. Related Work

2.1. In the Context of Classification and Regression Heads

The classification and regression head as sibling heads serve as essential components for general object detectors, where input features from the backbone network are transformed into classification scores and predicted boxes, respectively. Regional CNN (R-CNN) detectors [8, 24, 1] commonly deployed the shared head (*2fc*) in the regional network to classify and do the finer localization based on the region of interest (RoI) which is pooled out of the feature map. The work [6, 30] proposed different heads for R-CNN detectors and disentangle them by the individual network to achieve consistency between the classification and regression output. TSD [26] argued that classification and regression heads need different spatial features and the shared RoI pooling operator is a cause to the misalignment. For dense object detectors, things are different and not so straightforward to deal with for there are not RoI operators and the feature into different heads is hard to disentangle. As a common practice in [20, 29], the classification and localization heads are respectively comprised of several convolutional layers with the hope for different functionalities where the input feature is the same.

Different from previous work on the feature or the structure, our proposed method tackles the problem of inconsistency from the perspective of designing training samples respectively for each head. Previous methods on the supervision [12, 28, 2, 34] involved solely the unidirectional supervision either from the regression to the classification or vice versa. In contrast, our proposed MuSu supervises each head by training samples defined by the counterpart head output and ensures the consistency in a bidirectional manner. The work most related to this paper [33] shares the same address with our method, which defined the customized IoU criteria with the consideration of the counterpart head output for sample definition. However, details in [33] are highly hand-crafted with the customized IoUs and the improvement is not validated on recent detectors while our MuSu brings improvements over strong baselines with the simple and adaptive supervision scheme design.

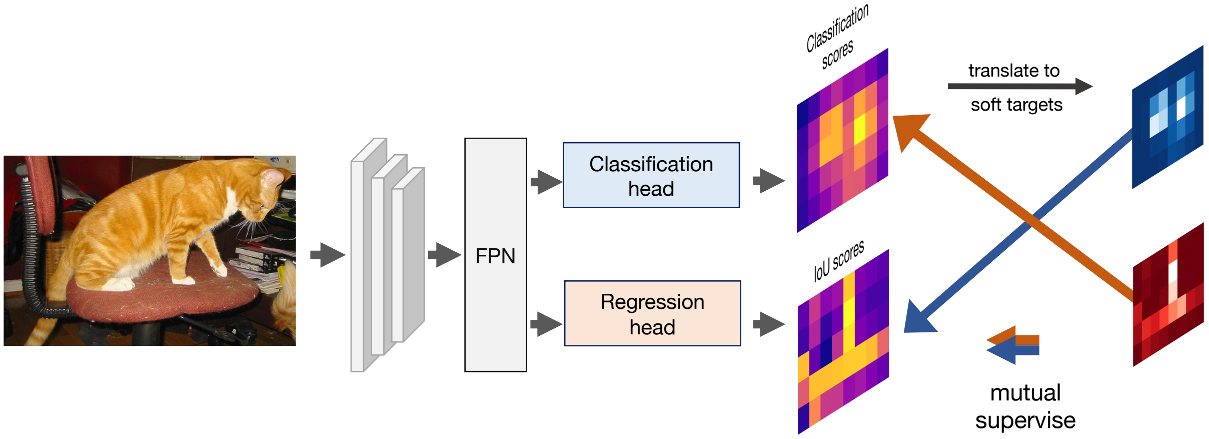

Figure 2. **Illustration of the mutual supervision (MuSu)** for the classification and regression head. We apply soft targets to supervise by weighting base loss at each location for these two heads respectively. The weights for the regression head are mainly decided by the classification score and the weights for the classification head are in turn mostly based on the localization scores (IoU scores). For more clarity, we do not show the construction of candidate bags and multi-level predictions.

2.2. In the Context of Training Sample Selection

The most popular strategy to select training samples is to use IoU as a criterion between an anchor and a ground-truth box, dating back to [9, 24]. Recently, various training sample selection strategies are proposed based on either the geometric relation, the classification score, the IoU or jointly them, to determine which object an candidate anchor belongs to in the training phase, and exploit the potential of a detector more further. FreeAnchor [36] was the first to adaptive training samples based on the customized likelihood by classification scores and IoUs. The literature [13, 16, 38, 27] proposed to explicitly select training samples by the joint criteria of the classification and the regression. ATSS [35] utilized the statistics of IoUs with regard to anchors with an object to determine positive samples. PAA [14] introduced a probabilistic process to the training sample selection and determine samples by the expectation-maximization algorithm. All these work cast improvements over the performance and has indicated the significance of designing better training samples.

Our method follows this research line but differs from these methods above. We step further in this line of adaptive training samples by assigning different samples to different heads and our proposed method automatically mines classification samples from the IoU and regression samples from the classification score. Fortunately, with this mutual supervision, our method MuSu also gets rid of the geometric prior and subtle treatment for each pyramid level in these adaptive approaches and in that sense, our proposed MuSu method is the neatest way to adaptively assign training samples by far while achieving promising results.

3. Proposed Method

To make accurate detections, a dense detector is expected to have alignment between the classification and regression heads since that the post-processing NMS only keeps detections with the maximum classification confidence when there are multiple overlapping ones. In detectors like RetinaNet [20], the classification head is trained by the supervision signal where the overlapping of predicting and ground-truth box is higher over a certain value, without the further consideration of how well the ground-truth is localized. Indeed, the current pipeline expects that the classification confidence represents not just how well the detector classify but also how well the detector localizes, as argued by [17, 14, 12, 29]. Therefore, the spatial distribution of the supervision for the classification head should rely on the localizing performance of the regression head, that is, where the IoU score is larger, where the classification supervision is stronger. In turn, the supervision for the regression should be also imposed more on the positions with higher classification scores, forcing well-classified ones to regress accurately as well. Figure 2 depicts this dependency between them and mutual supervision.

We introduce the *Mutual Supervision* (MuSu) algorithm for dense object detectors as a simple instantiation of this mutual philosophy. Specifically, MuSu ensures the consistency between the classification and regression in the training procedure by assigning training samples mutually and reciprocally from and for these two heads. MuSu treats the training sample in the soft target form by weighting losses of anchors in a ranking mechanism. MuSu can be described as three steps: **i)** construct adaptive candidate bags jointly by the classification and regression head to select the candidate anchors most belonging to an object; **ii)** compute candidate rankings from the perspectives of the classification and regression respectively inside the candidate bags; **iii)** translate these rankings to weights to sum out losses of each position and supervise the classification and regression head. The MuSu algorithm is depicted in Procedure 1.

3.1. Adaptive Candidate Bag Construction

We first construct adaptive candidate bag for every object as a preliminary step to filter out plenty of unsuitable anchors to better perform the following mutual supervision. The proposed candidate bag adaptively keeps out the false candidates to an object jointly by the classification and regression head and prevent anchors which obviously belong to the background or other objects from feeding into the next procedure. Otherwise, the mutual scheme will confuse detectors since classification scores are instance-agnostic and tend to be noisy at the initial stage.

More formally, considering for an object j, given an anchor i as well as its classification score p_i and IoU criterion IoU_i w.r.t. object j, we define a joint likelihood of weighting how much an anchor i is a candidate to object j, that is

$$P_i = p_i q_i, \tag{1}$$

where $q_i = \text{IoU}_i^{\theta}$ and θ is the weighting coefficient that rescales IoU in the exponential way to approximate the range of classification scores. θ is set to 4 our experiments. We calculate the threshold of candidate bags by

$$t = b \cdot \max_i P_i, \tag{2}$$

where b is the thresholding parameter less than 1. Any anchor in the ground-truth box with the joint likelihood higher than t becomes a candidate in the candidate bag $\mathcal{C}_j$ for an object j. As this procedure only chooses candidates loosely and filters out obviously unsuitable anchors, the parameter b is preferred to a low value, *e.g.*, 0.1. For multiple objects situation, we only keep the object j with the highest IoU in an anchor i involved in this computation and leave other ground-truth boxes out of consideration. This also makes the candidate bag mutually exclusive with regard to the object. By this means, we assign the ground-truth box j to each candidate $i \in \mathcal{C}_j$ without conflicts.

The candidate bag is also adaptive in its size. When the misalignment between the classification and regression head occurs, the threshold of a candidate bag becomes lower as the maximum of the joint likelihood for an object is also lower. More candidates are selected into the bag under this situation and this mechanism enables us to mines hard objects concerning the inconsistency between the classification and regression by enlarging their sample number and focus on them during training.

Procedure 1 : Mutual **Su**pervision algorithm

```
Input: $\mathcal{G}, \mathcal{A}$
    $\mathcal{G}$ is a set of ground-truth boxes
    $\mathcal{A}$ is a set of all anchors across all pyramid levels
Output: $\mathcal{L}^{cls}, \mathcal{L}^{reg}$
    $\mathcal{L}^{cls}$ is the total loss for the classification head
    $\mathcal{L}^{reg}$ is the total loss for the regression head
initialize $w_i^{cls}$ and $w_i^{reg}$ to zeros for all $i \in \mathcal{A}$;
for every ground-truth box $j \in \mathcal{G}$ do
    // construct candidate bag $\mathcal{C}_j$ for box $j$
    $\mathcal{A}_j \leftarrow \{i \in \mathcal{A}$ : anchor $i$ lies in the box $j$ and has maximum IoU with $j$ across $\mathcal{G}\}$;
    compute $P_i$ in Equ 1 for every anchor $i \in \mathcal{A}_j$;
    $\mathcal{C}_j \leftarrow \{i \in \mathcal{A}_j : P_i > b \cdot \max_{k \in \mathcal{A}_j} P_k\}$;
    // compute candidate rankings in candidate bags $\mathcal{C}_j$
    compute $v_i^{cls}, v_i^{reg}$ for $i \in \mathcal{C}_j$ according to Equ 3;
    sort $v_i^{cls}$ and $v_i^{reg}$ in descending order and obtain ranking $R_i^{cls}$ and $R_i^{reg}$ (starting from 0);
    // compute base losses and transform rankings to weights
    assign box $j$ to $i$ in $\mathcal{C}_j$ as detection target and compute base losses for all $i$ in $\mathcal{C}_j$;
    translate $R_i^{cls}, R_i^{reg}$ to $w_i^{cls}$ and $w_i^{reg}$ according to Equ 4;
end for
caculate $\mathcal{L}^{cls}$ and $\mathcal{L}^{reg}$ according to Equ 5;
```

3.2. Mutual Supervision

As we complete constructing candidate bags to filter out background anchors, we are ready to apply our proposed mutual supervision for classification and regression heads. For each head, MuSu assigns each candidate a ranking in descending order by evaluating the accuracy between the counterpart head prediction and the ground-truth object. Then MuSu translates the ranking to the weight for each candidate. We reuse the classification score p_i and the scaled IoU q_i in Equation 1 as the evaluation criteria for the classification and regression head. A natural choice is to use p_i for computing rankings for candidates of the regression head, R_i^{reg}, and use q_i for rankings of the classification head, R_i^{cls}. However, we found that this straightforward way of mutual supervision performed inferior in our experiments. In contrast, MuSu utilizes the regularized criteria values to compute rankings for candidates in the symmetric form,

$$\begin{cases} v_i^{cls} = q_i \cdot p_i^{\alpha}, \\ v_i^{reg} = p_i \cdot q_i^{\alpha}, \end{cases} \tag{3}$$

where α acts like a hyper-parameter, varying from 0 to 1, which regularizes the mutual scheme by also considering the output of the head itself. Our mutual supervision scheme can be a generalized training sample framework, where $\alpha = 1$ gives recently studied training sample strategies based on joint likelihood by these two heads [38, 13, 14] and $\alpha = 0$ gives the straightforward mutual supervision without the regularization of the supervised head itself.

3.3. Loss Weighting Paradigm

As we obtain regularized criteria v_i^{cls} and v_i^{reg}, MuSu sorts these values in a descending order within each candidate bag separately for both the classification and regression head to acquire the ranking R_i^{cls} and R_i^{reg} (starting from 0, increasing by step size 1, *i.e*., $0, 1, 2, \cdots$). MuSu supervises these two heads in the soft target form by weighting losses for each candidate and summing these weighted losses into a total loss. The weights w_i^{cls} and w_i^{reg} for candidates are decided by the ranking of each candidate separately for two heads and MuSu adopts a negative exponential way to translate rankings to weights:

$$\begin{cases} w_i^{cls} = \exp(-R_i^{cls}/\tau^{cls}), \\ w_i^{reg} = \exp(-R_i^{reg}/\tau^{reg}), \end{cases} \tag{4}$$

where τ^{cls} and τ^{reg} are temperature coefficients for the classification and regression head, indicating how many weights are assigned for samples to an object. As the ranking $R_i^{(\cdot)}$ increases ($v_i^{(\cdot)}$ becomes smaller), the weights are exponentially decreasing at a speed related to the temperature $\tau^{(\cdot)}$. Thanks to the mutual supervision scheme, we can control the number of positive training samples respectively for each head and we found that the performance would be better if we assign less weights to the regression head.

The total loss in an image for each head can be formulated in general as:

$$\mathcal{L} = \frac{1}{N} \sum_i w_i \ell_i, \tag{5}$$

where the normalized term $N = \sum_i w_i$ and ℓ_i is the loss function with regard to the prediction and ground-truth assigned for each anchor i. ℓ can be arbitrary loss functions for each head, *e.g*., the focal loss [20] for classification and GIoU [25] loss for regression. Details with not-assigned classes for the focal loss are discussed in Section 4.1.

It is notable that MuSu is not a specific loss function for either classification or regression and *de facto* acts as a hyper-loss formulation built upon these base losses. Actually, the focus of MuSu is to discuss the sample assignment for each head in two aspects: first, the ground-truth assignment for position i, stands for which object is the target of position i to supervise; second, weights for assigned training samples, w_i, indicating how much the position i should be supervised. As a plus, we separate the assignment strategies from underlying loss function choices and put attention on the relative rankings of anchors inside candidate bags, guaranteeing that the absolute amplitude of losses has no effect on assignments.

We summarize our proposed mutual supervision approach MuSu as several key points: *first*, MuSu exploits the spatial distribution of scores originating from the counterpart head to adaptively determine training samples for the classification and regression heads in a respective manner. This paradigm circumvents either any hand-crafted training sample assignments or geometric clues[1]. Thus, MuSu emerges as a simple and general training sample selection approach; *second*, MuSu enables detectors to align classification scores to IoUs scores, making detectors friendly to the NMS procedure and the final detection evaluation; *third*, MuSu disentangles the training sample assignment and the choices of base loss functions in the mutual supervision since it utilizes the relative ranking to determine loss weights associated with these anchors, which is extensible to any loss function improvement in the future; *finally*, MuSu alleviates the regressing difficulty of the regression head by assigning fewer positive samples for it and focusing on positions with higher classification scores. Experiments shows that MuSu-trained detectors lead to the consistent better performance.

4. Experiments

To validate the effectiveness of our proposed mutual supervision scheme for classification and regression head, we conduct experiments on MS COCO detection dataset [21] in this section. Following the common practice of previous work, we use `trainval35k` subset consisting up of 115K images to train our models and use `minival` subset of 5K images as the validation set. We report our ablation study results on `minival` subset. We also submit our final model results on the `test-dev` subset, whose labels are not publicly visible, to the MS COCO evaluation server to compare with state-of-the-art models. We implement our MuSu method in mmdetection codebase [5].

4.1. Implementation Details

Network structure. Theoretically, our mutual supervision method is universal for dense object detectors. In this paper, we adopt the recently-proposed dense detector FCOS [29] as our network architecture. The FCOS architecture serves as a strong baseline for dense detectors by utilizing Group Normalization [31] to both the classification and regression detection head, adding trainable scalars for each pyramid level on FPN [19] and using the centerness layer from the last feature map of the regression head to filter out a number of inaccurate detections. As our proposed method selects training samples adaptively and does not depend on the fixed centerness estimation, following previous work [38], we redirect the output of the centerness layer

[1] Except for the inner-box restriction, which is necessary as the FCOS-like architecture uses non-negative distance to predict offsets to bounding box borders.

in the FCOS architecture to the output of the classification head as so-called implicit objectness and merge them by multiplication to get final classification scores.

Initializations. All backbones of detectors throughout our experiments are initialized from the pre-trained models on the ImageNet dataset [7]. For stabilization during early training, we initialize weights of the last convolutional layer in the regression head to zeros. We also set a constant stride factor on each feature pyramid level of FPN [19] to scale regression boxes, starting from stride $s = 8$ at the finest pyramid level P_3 to $s = 128$ at the level P_7. These settings make boxes predicting from the regression branch for each position initialized to the same size $2s \times 2s$ for a FPN level, serving as a geometric prior in early iterations for more stable mutual supervision.

Mutual supervision instantiation. We set the temperature τ^{cls}, which controls the number of positive samples assigned to an object, to the squared root of the candidate bag size, and then set the temperature for the regression branch to the half of the temperature for the classification ($\tau^{cls} : \tau^{reg} = 2 : 1$) as our default. That is,

$$\begin{cases} \tau^{cls} = \sqrt{|\mathcal{C}_j|}, \\ \tau^{reg} = 0.5\tau^{cls} = 0.5\sqrt{|\mathcal{C}_j|}. \end{cases} \tag{6}$$

The temperature τ^{cls} and τ^{reg} are specific to a candidate bag of a ground-truth object j. The squared root operator makes the temperature vary moderately across different objects when the candidate bag size varys a lot and leads to more stable training. We set the thresholding coefficient b in Equation 2 to 0.1 as our default.

We adopt the focal loss [20] as our base loss for the classification and the GIoU loss [25] for the regression. The focal loss tackles the classification task in detection as the multi-class binary classification problem. For an anchor, there exists the negative classification of non-target classes along with the positive classification of the target class. In addition, the negative classification should be also applied to the assigned class with soft targets. Thus, we treat it carefully and separate the focal loss into three parts: the positive term for the assigned class label, the negative penalty term for the assigned class label, and the background term for all other not-assigned class labels. We extend the loss form in Equation 5:

$$\mathcal{L}_{cls} = \frac{1}{N}\sum_i [w_i^{cls} \cdot \ell_i^+ + (1 - w_i^{cls})^\beta \cdot \ell_i^- + \ell_i^{bg}], \tag{7}$$

method	AP	AP$_{50}$	AP$_{75}$	AP$_S$	AP$_M$	AP$_L$
FCOS	36.5	55.7	38.3	21.2	40.3	48.1
FCOS++	38.6	57.2	41.7	22.4	42.4	50.2
MuSu with						
$v_i^{cls} = v_i^{reg} = p_i$	38.3	**60.0**	40.5	22.8	41.5	49.3
$v_i^{cls} = v_i^{reg} = q_i$	31.8	49.9	33.7	15.0	34.9	45.2
MuSu under v_i^{cls} and v_i^{reg} in Equ 3 with						
$\alpha = 1.0$	40.4	59.6	43.9	**23.5**	43.5	53.7
$\alpha = 1/2$	40.3	59.1	43.8	23.1	43.7	53.5
$\alpha = 1/3$	**40.6**	58.9	**44.3**	23.0	44.0	**54.2**
$\alpha = 1/4$	40.5	58.9	43.8	23.4	**44.2**	53.6
$\alpha = 1/6$	40.4	59.0	43.6	22.5	**44.2**	53.9
$\alpha = 0$	38.5	57.5	41.4	20.9	42.9	52.4

Table 1. **MuSu criteria value settings for each head** on COCO `minival` set with ResNet-50 backbone (same in tables below, unless otherwise specified). FCOS and FCOS++ are baselines. FCOS++ denotes the improved FCOS with tricks (e.g., center sampling). The term α refers to the regularizing factor in Equation 3. The settings $v^{(\cdot)} = p$ and $v^{(\cdot)} = q$ stand for that the method assigns the same weights for each head, totally according to the classification or localization score of an anchor inside the bag.

where the ℓ_i^+ is the focal loss to the positive classification of the assigned label while ℓ_i^- is the focal loss for the negative classification of the assigned label as the penalty for the position with insufficient w_i^{cls}. These two loss terms function as the soft target in a unified manner. Background loss term ℓ_i^{bg} is the sum of the focal loss for negative classification of all other classes which are not assigned to anchor i. The focusing and balance parameter for the focal loss follows the default settings in [29] and the penalty decay term β is set to 4 following [15, 37]. The total loss for detection is simply the sum of the classification and regression loss,

$$\mathcal{L}_{det} = \mathcal{L}_{cls} + \mathcal{L}_{reg}. \tag{8}$$

Optimization and inference. We use SGD with the learning rate 0.01, momentum factor 0.9, and weight decay 0.0001 to optimize our models throughout experiments. A total batch of 16 images, 2 images per GPU, are used in the training. The statistics and affine parameters of batch normalization layers in the backbone are frozen as in [29]. For ablation studies, we train models with the ResNet-50 backbone [11] in 90K iterations with the learning rate warmup during the first 500 iterations. The learning rate is divided by a factor of 10 at the 60K and 80K iteration, respectively. All images in the 90K training scheme are resized to their shorter size being 800 and their longer size not greater than 1333 and are randomly flipped horizontally as the only data augmentation. At the inference stage, we resize the input image to the same size in the training procedure without random flipping. The threshold of the classification score is set to 0.05 and the NMS threshold is set to 0.6 in the detection pipeline, also following recent common practice. The optimization and inference details are kept the same throughout our experiments unless otherwise stated.

4.2. Training with Mutual Supervision

Study on the mutual supervision. We start our experiments from the vanilla FCOS detector as our baseline. The vanilla FCOS is supervised by dense signals and serves

b	AP	AP$_{50}$	AP$_{75}$	AP$_S$	AP$_M$	AP$_L$
0.20	40.3	58.8	43.7	22.8	43.4	53.4
0.10	**40.6**	**58.9**	**44.3**	23.0	**44.0**	**54.2**
0.05	40.4	**58.9**	43.7	**23.5**	43.8	54.0

Table 2. **Different threshold coefficients** b in Equation 2.

(τ^{cls}, τ^{reg})	AP	AP$_{50}$	AP$_{75}$	AP$_S$	AP$_M$	AP$_L$
$(10.0, 5.0)$	40.1	58.5	43.3	22.2	43.4	53.7
$(5.0, 2.5)$	39.9	58.5	43.4	22.1	43.5	53.2
$(\sqrt{\lvert\mathcal{C}_j\rvert}, 0.5\sqrt{\lvert\mathcal{C}_j\rvert})$	**40.6**	**58.9**	**44.3**	**23.0**	**44.0**	**54.2**

Table 3. **Adaptive temperatures** τ^{cls} **and** τ^{reg} **benefit**. First two rows act as counterparts with fixed temperatures.

as a competitive baseline for dense detectors, which got 36.5 AP in Table 1. The FCOS++ model in Table 1 denotes the improved architecture and more importantly, the refined highly hand-crafted training sample, which only assigns positive samples within the center area of objects. In contrast, our MuSu emerges as an adaptive training sample assignment approach for dense object detectors and the key component in MuSu is the criteria value in Equation 3 as it decides which training sample selection strategy for each head our method adopts. In Table 1, we carry out experiments of various settings for criteria values in Equation 3. The settings whose weights for two heads are totally decided by the single head output (the classification p_i or the regression q_i) and assign the same criteria value for both two heads, which in other words enables the sole unidirectional supervision without the mutual scheme, leads to inferior results of 38.3 and 31.8 AP respectively.

The naive mutual supervision without the regularized term (setting $\alpha = 0.0$) achieves 38.5 AP, comparable to the highly hand-crafted FCOS++ model. When we add the regularized factor, even from $\alpha = 1/6$, the performance of models significantly improves to 40.4 AP. With the regularized factor $\alpha = 1/3$, the performance of models trained with the mutual supervision leads to a best 40.6 AP, 2.0 AP higher than the FCOS++ model. We argue that the regularizing term is necessary for the assignment because it is also aware of *how well each head itself learned* in the training procedure and makes the best of the use of each head prediction to avoid the assignment fluctuation. It is notable that the MuSu with $\alpha = 1.0$, which assigns the same criteria value for training sample selection for two heads, is also a case of mutual supervision where the supervision of a head is also aware of the prediction of the counterpart head. In this sense, we include training samples based on the joint likelihood explored by the recent approach [38, 13, 14] in our proposed MuSu approach. However, this same criteria strategy ($\alpha = 1$) suffers from stagnant or even degenerate performance when tiling more anchors while the MuSu with $\alpha = 1/3$ benefits from more anchors as discussed below.

τ^{cls}: τ^{reg}	AP	AP$_{50}$	AP$_{75}$
1 : 1	40.4	58.6	44.0
1.5 : 1	**40.6**	58.9	44.1
2 : 1	**40.6**	58.9	**44.3**
3 : 1	40.2	**59.0**	43.7

Table 4. **Varying the ratio of temperature coefficients** of the classification to the regression τ^{cls}: τ^{reg}.

α	#A	AP	AP$_{50}$	AP$_{75}$	AP$_S$	AP$_M$	AP$_L$
1.0	1	40.4	**59.6**	43.9	23.5	43.5	53.7
	2	40.4	**59.6**	44.1	23.7	43.4	53.6
	3	39.9	59.4	43.3	**23.9**	43.2	51.9
1/3	1	40.6	58.9	44.3	23.0	44.0	54.2
	2	40.6	58.8	44.4	23.1	44.0	54.3
	3	**40.9**	59.0	44.3	23.3	44.3	54.2
	4	40.8	58.9	**44.6**	23.6	**44.5**	**54.4**
	5	40.3	58.6	44.3	22.8	43.9	53.0

Table 5. **Tiling more anchors** when the regularizing term $\alpha = 1.0$ and $\alpha = 1/3$ in MuSu.

Study on adaptive candidate bags and temperature. As we discuss in Section 3.1, the candidate bag is designed for filtering out the plenty of background anchors adaptively by the joint likelihood of the classification and regression. The candidate bag only serves as a preliminary procedure to keep obviously unsuitable anchors from the next mutual assignment procedure, so the threshold coefficient b is preferred to a relatively low value. In Table 2, we vary the coefficient to see its impact. The coefficient $b = 0.10$ gives the best result.

A candidate bag is also adaptive with regard to its size. On account of that, MuSu can put more focuses on objects with strong inconsistency between classification and regression by assigning more positive samples to them in the relation depicted in Equation 4 and 6. We validate the effectiveness of adaptive candidate bag on final detectors by disabling the adaptive temperature w.r.t the bag size and setting τ^{cls} and τ^{reg} to a fixed number. Borrowing the average temperature τ^{cls} and τ^{reg} when using adaptive candidate bags, the temperature for the classification τ^{cls} is set to the fixed constant 5.0 across objects and keep $\tau^{cls} : \tau^{reg} = 2 : 1$. For more ablation, we add the situation $\tau^{cls} = 10.0$ in Table 3. We find that the adaptive temperature as the function of the bag size benefits our method by adaptively mining hard objects with regard to the inconsistency. We also present results of applying different ratios of temperatures for assigning samples to each head (τ^{cls}:τ^{reg}) under the setting of adaptive temperature to anchor bag sizes in Table 4, which indicates that moderately reducing samples for regression is favourable for the fine localization and overall performance.

Soft versus hard targets. As our method defines training samples for both heads as soft targets by weighting losses of them, one natural question is whether we can use hard targets instead of soft targets to achieve a similar performance. We trained the model with hard targets by modifying Equation 4 to $w_i^{(\cdot)} = \mathbb{I}[R_i^{(\cdot)} < \tau^{(\cdot)}]$, where $\mathbb{I}[\cdot]$ is the indicating function and this resulted in a 40.0 AP model, which is 0.6 AP behind the soft target scheme. This com-

method	backbone	AP	AP$_{50}$	AP$_{75}$	AP$_S$	AP$_M$	AP$_L$
FCOS [29]	ResNet-101	41.5	60.7	45.0	24.4	44.8	51.6
FreeAnchor [36]	ResNet-101	43.1	62.2	46.4	24.5	46.1	54.8
SAPD [39]	ResNet-101	43.5	63.6	46.5	24.9	46.8	54.6
MAL [13]	ResNet-101	43.6	61.8	47.1	25.0	46.9	55.8
ATSS [35]	ResNet-101	43.6	62.1	47.4	26.1	47.0	53.6
AutoAssign [38]	ResNet-101	44.5	64.3	48.4	25.9	47.4	55.0
PAA [14]	ResNet-101	**44.8**	**63.3**	48.7	**26.5**	**48.8**	56.3
MuSu (ours)	ResNet-101	**44.8**	63.2	**49.1**	26.2	47.9	**56.4**
SPAD [39]	ResNet-101-DCN	46.0	65.9	49.6	26.3	49.2	59.6
ATSS [35]	ResNet-101-DCN	46.3	64.7	50.4	27.7	49.8	58.4
PAA [14]	ResNet-101-DCN	**47.4**	**65.7**	51.6	**27.9**	**51.3**	**60.6**
MuSu (ours)	ResNet-101-DCN	**47.4**	65.0	**51.8**	27.8	50.5	60.0

Table 6. **Comparison on COCO `test-dev` set** by different training sample selection methods with ResNet-101 and ResNet-101-DCN.

parison supports the conclusion of the literature [17, 39, 34] and soft targets in our method share the same idea in the flexible classification to align regression scores.

Tiling more anchors. Placing multiple anchors at each spatial position of output detection maps is a common way to cover image boxes of different scales and aspect ratios as many as possible in dense object detectors. This strategy is popular among both one-stage detectors [20] or proposal networks of two-stage detectors [24] for achieving better performance. However, recent work [29, 35] challenges this necessity of tiling more anchors by changing sample assignment strategies and shows that there are no performance gains by placing more anchors under their settings.

To discuss multiple anchor situations, we set the initial scale and aspect ratio of anchors by initializing the bias parameter of the last convolutional layer to produce bounding boxes. The scale factor of an anchor and the aspect ratio are drawn uniformly at random from the interval $[1, 2]$ and $[\frac{1}{2}, 2]$, respectively.

Surprisingly, we find that tiling more anchors has a boost on the detection performance over competitive results under our mutual supervision scheme, even without well-crafted settings of scales and aspect ratios. As shown in Table 5, these results show that MuSu enables the detector to fully exploit the setting of more anchors. The performance of a detector can increase to about 40.9 AP when adding anchor per location to 3 or 4. In contrast, the counterpart results, which assigns the same criteria values for two heads with $\alpha = 1.0$, will not be better when adding more anchors and even suffer from that. The final MuSu model is 2.3 AP higher than FCOS++ model, 4.1 AP higher than the vanilla FCOS model, and 0.5 AP higher than our competitive baseline with $\alpha = 1.0$ and $\#A = 1$. This empirical evidence validates the effectiveness of our MuSu method beyond the single anchor situation.

4.3. Comparison to the State of the Art

To compare with other state-of-the-art methods of training sample selection for detection, we use deeper backbones and deformable one [40] to train with our MuSu. To align with previous work and compare fairly, we extend the training schedule to the 180K iterations and reduce the learning rate at the 120K and 160K iteration by a factor of 0.1. For an input image, we resize the shorter side to a scale randomly chosen value of $[640, 800]$. We train our MuSu detectors with 3 anchors per location ($\#A = 3$). For the DCN variant, we also apply deformable convolutional layer to the last layer on each head following [35, 14]. As shown in Table 6, both the ResNet-101 detector and the DCN variant trained by the MuSu surpass previous competitive models in the overall AP while achieving the new state-of-the-art AP$_{75}$ without bells and whistles at the inference stage. Further, MuSu-trained models are on par with PAA models that are with the score voting as the improvement at inference stage.

It is worth noting that our MuSu offers as a simple instantiation of our proposed mutual supervision and this scheme, in general, is also compatible to specific training sample selection methods, such as the PAA algorithm [14] for each head to expect better results.

5. Conclusion

In this paper, we have presented the mutual supervision (MuSu) scheme for training accurate dense object detectors in which we break the convention of the same training samples for the classification and regression heads and then these two heads are supervised based on the output of each other in the soft target way. MuSu makes a big step further to fully adaptive training sample selection by means of assigning different samples to these two heads in a mutual manner without the subtle geometric designing. Moreover, we discuss multiple anchor settings under our proposed mutual supervision and find that is beneficial to our method. Experimental results on the challenging MS COCO benchmark validate the effectiveness of our proposed MuSu training scheme on detectors.

Acknowledgements. This work is supported by National Natural Science Foundation of China (No. 62076119, No. 61921006), Program for Innovative Talents and Entrepreneur in Jiangsu Province, and Collaborative Innovation Center of Novel Software Technology and Industrialization.

References

[1] Zhaowei Cai and Nuno Vasconcelos. Cascade R-CNN: delving into high quality object detection. In *CVPR*, 2018. 1, 2

[2] Yuhang Cao, Kai Chen, Chen Change Loy, and Dahua Lin. Prime sample attention in object detection. In *CVPR*, 2020. 2

[3] Nicolas Carion, Francisco Massa, Gabriel Synnaeve, Nicolas Usunier, Alexander Kirillov, and Sergey Zagoruyko. End-to-end object detection with transformers. In *ECCV*, 2020. 1

[4] Kai Chen, Jiangmiao Pang, Jiaqi Wang, Yu Xiong, Xiaoxiao Li, Shuyang Sun, Wansen Feng, Ziwei Liu, Jianping Shi, Wanli Ouyang, Chen Change Loy, and Dahua Lin. Hybrid task cascade for instance segmentation. In *CVPR*, 2019. 1

[5] Kai Chen, Jiaqi Wang, Jiangmiao Pang, Yuhang Cao, Yu Xiong, Xiaoxiao Li, Shuyang Sun, Wansen Feng, Ziwei Liu, Jiarui Xu, Zheng Zhang, Dazhi Cheng, Chenchen Zhu, Tianheng Cheng, Qijie Zhao, Buyu Li, Xin Lu, Rui Zhu, Yue Wu, Jifeng Dai, Jingdong Wang, Jianping Shi, Wanli Ouyang, Chen Change Loy, and Dahua Lin. Mmdetection: Open mmlab detection toolbox and benchmark. In *arXiv*, 2019. 5

[6] Bowen Cheng, Yunchao Wei, Honghui Shi, Rogério Schmidt Feris, Jinjun Xiong, and Thomas S. Huang. Revisiting RCNN: on awakening the classification power of faster RCNN. In *ECCV*, 2018. 2

[7] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Fei-Fei Li. Imagenet: A large-scale hierarchical image database. In *CVPR*, 2009. 6

[8] Ross B. Girshick. Fast R-CNN. In *ICCV*, 2015. 1, 2

[9] Ross B. Girshick, Jeff Donahue, Trevor Darrell, and Jitendra Malik. Region-based convolutional networks for accurate object detection and segmentation. *IEEE Trans. Pattern Anal. Mach. Intell.*, 38(1):142–158, 2016. 3

[10] Kaiming He, Georgia Gkioxari, Piotr Dollár, and Ross B. Girshick. Mask R-CNN. In *ICCV*, 2017. 1

[11] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *CVPR*, 2016. 6

[12] Borui Jiang, Ruixuan Luo, Jiayuan Mao, Tete Xiao, and Yuning Jiang. Acquisition of localization confidence for accurate object detection. In *ECCV*, 2018. 2, 3

[13] Wei Ke, Tianliang Zhang, Zeyi Huang, Qixiang Ye, Jianzhuang Liu, and Dong Huang. Multiple anchor learning for visual object detection. In *CVPR*, 2020. 3, 5, 7, 8

[14] Kang Kim and Hee Seok Lee. Probabilistic anchor assignment with iou prediction for object detection. *ECCV*, 2020. 3, 5, 7, 8

[15] Hei Law and Jia Deng. Cornernet: Detecting objects as paired keypoints. In *ECCV*, 2018. 6

[16] Hengduo Li, Zuxuan Wu, Chen Zhu, Caiming Xiong, Richard Socher, and Larry S. Davis. Learning from noisy anchors for one-stage object detection. In *CVPR*, 2020. 3

[17] Xiang Li, Wenhai Wang, Lijun Wu, Shuo Chen, Xiaolin Hu, Jun Li, Jinhui Tang, and Jian Yang. Generalized focal loss: Learning qualified and distributed bounding boxes for dense object detection. *NeurIPS*, 2020. 3, 8

[18] Yanghao Li, Yuntao Chen, Naiyan Wang, and Zhaoxiang Zhang. Scale-aware trident networks for object detection. In *ICCV*, 2019. 1

[19] Tsung-Yi Lin, Piotr Dollár, Ross B. Girshick, Kaiming He, Bharath Hariharan, and Serge J. Belongie. Feature pyramid networks for object detection. In *CVPR*, 2017. 5, 6

[20] Tsung-Yi Lin, Priya Goyal, Ross B. Girshick, Kaiming He, and Piotr Dollár. Focal loss for dense object detection. In *ICCV*, 2017. 1, 2, 3, 5, 6, 8

[21] Tsung-Yi Lin, Michael Maire, Serge J. Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C. Lawrence Zitnick. Microsoft COCO: common objects in context. In *ECCV*, 2014. 2, 5

[22] Wei Liu, Dragomir Anguelov, Dumitru Erhan, Christian Szegedy, Scott E. Reed, Cheng-Yang Fu, and Alexander C. Berg. SSD: single shot multibox detector. In *ECCV*, 2016. 1

[23] Joseph Redmon, Santosh Kumar Divvala, Ross B. Girshick, and Ali Farhadi. You only look once: Unified, real-time object detection. In *CVPR*, 2016. 1

[24] Shaoqing Ren, Kaiming He, Ross B. Girshick, and Jian Sun. Faster R-CNN: towards real-time object detection with region proposal networks. In *NIPS*, 2015. 1, 2, 3, 8

[25] Hamid Rezatofighi, Nathan Tsoi, JunYoung Gwak, Amir Sadeghian, Ian D. Reid, and Silvio Savarese. Generalized intersection over union: A metric and a loss for bounding box regression. In *CVPR*, 2019. 5, 6

[26] Guanglu Song, Yu Liu, and Xiaogang Wang. Revisiting the sibling head in object detector. In *CVPR*, 2020. 2

[27] Peize Sun, Yi Jiang, Enze Xie, Wenqi Shao, Zehuan Yuan, Changhu Wang, and Ping Luo. What makes for end-to-end object detection? In *ICML*, 2021. 3

[28] Zhiyu Tan, Xuecheng Nie, Qi Qian, Nan Li, and Hao Li. Learning to rank proposals for object detection. In *ICCV*, 2019. 2

[29] Zhi Tian, Chunhua Shen, Hao Chen, and Tong He. FCOS: fully convolutional one-stage object detection. In *ICCV*. IEEE, 2019. 1, 2, 3, 5, 6, 8

[30] Yue Wu, Yinpeng Chen, Lu Yuan, Zicheng Liu, Lijuan Wang, Hongzhi Li, and Yun Fu. Rethinking classification and localization for object detection. In *CVPR*, 2020. 2

[31] Yuxin Wu and Kaiming He. Group normalization. *Int. J. Comput. Vis.*, 128(3):742–755, 2020. 5

[32] Ze Yang, Shaohui Liu, Han Hu, Liwei Wang, and Stephen Lin. Reppoints: Point set representation for object detection. In *ICCV*, pages 9656–9665. IEEE, 2019. 1

[33] Heng Zhang, Élisa Fromont, Sébastien Lefèvre, and Bruno Avignon. Localize to classify and classify to localize: Mutual guidance in object detection. In *ACCV*, 2020. 2

[34] Haoyang Zhang, Ying Wang, Feras Dayoub, and Niko Sünderhauf. Varifocalnet: An iou-aware dense object detector. In *CVPR*, 2021. 2, 8

[35] Shifeng Zhang, Cheng Chi, Yongqiang Yao, Zhen Lei, and Stan Z. Li. Bridging the gap between anchor-based and anchor-free detection via adaptive training sample selection. *CVPR*, 2019. 2, 3, 8

[36] Xiaosong Zhang, Fang Wan, Chang Liu, Rongrong Ji, and Qixiang Ye. Freeanchor: Learning to match anchors for visual object detection. In *NeurIPS*, 2019. 3, 8

[37] Xingyi Zhou, Dequan Wang, and Philipp Krähenbühl. Objects as points. *ArXiv*, 2019. 1, 6

[38] Benjin Zhu, Jianfeng Wang, Zhengkai Jiang, Fuhang Zong, Songtao Liu, Zeming Li, and Jian Sun. Autoassign: Differentiable label assignment for dense object detection. *ArXiv*, 2020. 3, 5, 7, 8

[39] Chenchen Zhu, Fangyi Chen, Zhiqiang Shen, and Marios Savvides. Soft anchor-point object detection. In *ECCV*, 2020. 8

[40] Xizhou Zhu, Han Hu, Stephen Lin, and Jifeng Dai. Deformable convnets v2: More deformable, better results. In *CVPR*, 2019. 8

[41] Xizhou Zhu, Weijie Su, Lewei Lu, Bin Li, Xiaogang Wang, and Jifeng Dai. Deformable DETR: deformable transformers for end-to-end object detection. In *ICLR*, 2021. 1