



# Small Object Detection *via* Pixel Level Balancing With Applications to Blood Cell Detection

Bin Hu<sup>1</sup>, Yang Liu<sup>2,3\*</sup>, Pengzhi Chu<sup>1</sup>, Minglei Tong<sup>4\*</sup> and Qingjie Kong<sup>5</sup>

<sup>1</sup>Department of Compute Science and Engineering, School of Electronic Information and Electrical Engineering, Shanghai Jiao Tong University, Shanghai, China, <sup>2</sup>Department of Dermatology, Shanghai Ninth People's Hospital Affiliated to Shanghai Jiao Tong University School of Medicine, Shanghai, China, <sup>3</sup>Department of Laser and Aesthetic Medicine, Shanghai Ninth People's Hospital Affiliated to Shanghai Jiao Tong University School of Medicine, Shanghai, China, <sup>4</sup>College of Electronics and Information Engineering, Shanghai University of Electric Power, Shanghai, China, <sup>5</sup>Riseye Research, Riseye Intelligent Technology (Shanghai) Co., Ltd., Shanghai, China

## OPEN ACCESS

### Edited by:

Zhibin Niu,  
Tianjin University, China

### Reviewed by:

Bo Jin,  
East China Normal University, China  
Xiaobin Zhu,  
University of Science and Technology  
Beijing, China  
Min Cao,  
Soochow University, China  
Changsheng Li,  
Beijing Institute of Technology, China

### \*Correspondence:

Yang Liu  
liuyang870422@126.com  
Minglei Tong  
tongminglei@shiep.edu.cn

### Specialty section:

This article was submitted to  
Medical Physics and Imaging,  
a section of the journal  
Frontiers in Physiology

**Received:** 02 April 2022

**Accepted:** 24 May 2022

**Published:** 17 June 2022

### Citation:

Hu B, Liu Y, Chu P, Tong M and  
Kong Q (2022) Small Object Detection  
*via* Pixel Level Balancing With  
Applications to Blood Cell Detection.  
*Front. Physiol.* 13:911297.  
doi: 10.3389/fphys.2022.911297

Object detection technology has been widely used in medical field, such as detecting the images of blood cell to count the changes and distribution for assisting the diagnosis of diseases. However, detecting small objects is one of the most challenging and important problems especially in medical scenarios. Most of the objects in medical images are very small but influential. Improving the detection performance of small objects is a very meaningful topic for medical detection. Current researches mainly focus on the extraction of small object features and data augmentation for small object samples, all of these researches focus on extracting the feature space of small objects better. However, in the training process of a detection model, objects of different sizes are mixed together, which may interfere with each other and affect the performance of small object detection. In this paper, we propose a method called pixel level balancing (PLB), which takes into account the number of pixels contained in the detection box as an impact factor to characterize the size of the inspected objects, and uses this as an impact factor. The training loss of each object of different size is adjusted by a weight dynamically, so as to improve the accuracy of small object detection. Finally, through experiments, we demonstrate that the size of objects in object detection interfere with each other. So that we can improve the accuracy of small object detection through PLB operation. This method can perform well with blood cell detection in our experiments.

**Keywords:** medical image detection, object detection, small object, pixel level balance, blood cell detection

## 1 INTRODUCTION

With the development of artificial intelligence technology, deep learning based on CNN (Convolutional Neural Network) has been widely used in medical image processing field. Using computer aided technology to analyze and process medical images can assist doctors doing qualitative and quantitative analysis of diseases, thereby improving the accuracy and reliability of medical diagnosis greatly. Medical image detection is one of the main tasks in the field of medical image processing. Many medical institutions in the world have rapidly entered this field. Medical image detection has been combined with artificial intelligence technology for a long time. As early as 1993, CNN has been used for lung nodule detection. In 1995, the technology was also applied to detect micro-calcification in mammography.

Medical detection technology has been continuously developed by applying CNN and other deep learning methods (McInerney and Terzopoulos, 1996; Handels et al., 2013; Litjens et al., 2017) to various medical images with different imaging mechanisms. For example, Setio et al. detected lung nodules in 3D chest CT scans and extracted 2D patches in nine different orientations centered on these candidates (Golan et al., 2016). Ross et al. utilized CNN to improve three existing CAD systems for the detection of colonic polyps, sclerosing spinal deformity, and lymphadenopathy in CT imaging (Roth et al., 2016). In recent years, object detection technology has been widely used in pathology (Janowczyk and Madabhushi, 2016), especially in blood cells detection (Yang et al., 2017; Pan et al., 2018; Fujita and Han, 2020). Detecting blood cells can assist diagnosing many kinds of diseases, such as diagnosing breast cancer by detecting mitosis or lymphocytes (Cireşan et al., 2013; Zhang et al., 2022). Object detection technology is constantly applied into application scenarios of medical image processing, and thus bringing more commercial value.

Object detection is mainly aimed at locating and identifying objects in different positions in the image. In medical image detection, the detection of small objects gets more attentions than the detection of large objects. For example, lesions are identified by detecting whether there are tiny abnormalities in the images, determining whether a patient is likely to develop the disease. In particular, in the detection of blood cells, some types of cells are rarer and smaller than others, but they play an indispensable and key role in the diagnosis of diseases. In such tasks, the large adjacent objects in the image can be ignored, but the detection of small objects is very important. However, object detection for small objects is precisely a more difficult task. Objectively speaking, small objects contain low pheromones and human beings have the disadvantage in recognizing small objects. These reasons bring a higher challenge to the detection task.

Previous research work mainly focused on how to enhance the detection model's extraction of the feature of small objects, and through some methods to solve the unbalance of the samples. In addition, there is also the use of rotation detection with angle factors (Yang et al., 2019a; Yang et al., 2019b; Qian et al., 2019; Yang and Yan, 2020) to better approximate the true position of small objects. However, it is rarely mentioned that under the multi-object detection task, there is a possibility of interference among the multiple objects within one image. In the objective logic of human observation, people can see large objects most intuitively among objects of various sizes, but it is easy to ignore the existence of small objects, let alone the recognition of the small objects. In the detection model, theoretically, it is also necessary to consider the mutual interference between objects of different sizes during the training process, and the contribution of objects with various sizes to the training loss are different. Therefore, it is meaningful to study the interference of large and small objects, and use some means to alleviate such interference and guide the model to the optimization direction of small object detection.

We studied the interference between objects of different sizes in the training process, and designed a weight coefficient called PLB weight to adjust the training effect of the model. The size of

the detected object is characterized by the size of the detection rectangle. We use the number of pixels contained in it as the calculation input of the weight coefficient. In the process of model training, this coefficient can be used to dynamically adjust the training loss of each object with various sizes, so that the model can be training to the direction of improving the accuracy of small object detection.

Our innovations and contributions to this work are summarized as follows:

- 1) Instead of setting fixed empirical values before training, we dynamically set the loss weights for objects of different sizes during the training process of the detection model.
- 2) With our proposed method PLB, the training trend of two-stage detector can be adjusted and the detection accuracy of small objects can be further improved.
- 3) Our method can be combined with other methods to improve the detection effect, bringing more potential capacity for some medical applications which need higher detection effect of small objects than bigger ones, such as blood cell detection.

## 2 RELATED WORK

Current detection models are divided into two categories: two-stage detectors and one-stage detectors. Two-stage detection network is represented by the RCNN series (Girshick et al., 2014; Girshick, 2015; He et al., 2015; Ren et al., 2015; Dai et al., 2016), the second category is represented by the YOLO series (Redmon et al., 2016; Redmon and Farhadi, 2017; Redmon and Farhadi, 2018; Bochkovskiy et al., 2020) and SSD series (Liu et al., 2016; Shrivastava et al., 2016; Fu et al., 2017; Jeong et al., 2017; Li and Zhou, 2017; Shen et al., 2017). Among them, the former adopts the RPN network (Girshick, 2015). When performing localization and recognition tasks, candidate rectangular boxes are proposed in the RPN phase. In the second stage, the candidate proposals are adjusted and the objects in the boxes are identified. One-stage detectors use an end-to-end deep neural network, and the model structure is simpler than two-stage detectors, bring a faster computing speed, so that it is more suitable for some time-sensitive application scenarios. But for the improvement of detection accuracy, one-stage detector usually weaker than two-stage detector.

No matter which detection model is used, the CNN model is used as the feature extractor to obtain the feature space of the train set. With the continuous development of CNN models in recognition tasks, especially the ResNet model (He et al., 2016) and the DenseNet model (Huang et al., 2017), it has been confirmed that the CNN model has a high accuracy and universality for feature extraction in classification tasks. In terms of improving the overall accuracy of the detection model, a deeper CNN can be used as the backbone network to extract the image features (Zhang et al., 2022). Attention mechanism such as SENet (Hu et al., 2018) is used to improve the sensitivity of the model to channel features. The model can be adjusted through EfficientNet (Tan and Le, 2019) by adjusting the

depth, width and pixel accuracy in the model to optimize the overall performance, such as EfficientDet (Tan et al., 2020). In addition, other methods such as NMS (Neubeck and Van Gool, 2006) and BN (Li et al., 2019) can be used to optimize the detection model comprehensively.

The cost of manual labeling of medical datasets is more expensive, and the acquisition and labeling of datasets is more difficult than other scenarios. Therefore, for some incompletely labeled datasets, some methods are also needed to improve the accuracy of object detection. Unsupervised active learning methods can be applied into this task to improve detection performance (Changsheng et al., 2019). Such as Active Learning Matrix Sketching (ALMS) (Li et al., 2021) which is used to do simultaneous sample and feature selection in an unsupervised setting. These methods aim to improve the effectiveness of the latent feature space (Li et al., 2022), so that the detection model can achieve more stable and good performance.

Improving the detection accuracy of small objects is a more difficult challenge. For the detection improvement of small target objects, rotation detection is also an effective method. Traditional detection models generally use horizontal rectangular boxes as labels for localization tasks. However, for small objects, the rotating detection boxes with an angle can more closely approximate the real position (Yang et al., 2021a; Yang et al., 2021b; Yang and Yan, 2022). Small objects have higher sensitivity requirements to position, and rotation detection can bring better training effects for the detection of small objects (Yang et al., 2020; Yang et al., 2021c; Yang et al., 2022).

In the detection model based on deep neural network, with the deepening of the network, the image features can be extracted better to fit our detection task. But in the feature space at the end of the model, the represented receptive field is getting bigger and bigger, while the features corresponding to the small objects may disappear. FPN network (Lin et al., 2017a; Pan et al., 2018) is a good solution to this problem. In this network structure, middle layers in the feature extraction process are reserved and combined with the upper and lower layers, so that the feature of small objects will not disappear with the deepening of the network. Finally multi-layer feature vectors are obtained by FPN. Among them, the low-dimensional feature has a smaller receptive field for small objects which is biased towards the shape features of the object, while the high-dimensional feature has a larger receptive field which is biased towards the semantic features. The FPN network improves the feature extraction effect especially for small objects as it can retain more features. As a result, the FPN network can effectively improve the detection accuracy of small objects.

Different kinds of imbalances within the training samples are also the reasons for the difficulty of detecting small objects. These imbalances mainly include the imbalance of the object categories, and the imbalance of the proportion of small objects and large objects in the samples. In addition, the imbalance between the foreground and background is also an important factor that disturbs the training effect. In the detection model, the corresponding weights can be set for each category in the data set through a prior knowledge, and the loss in the training process is weighted to adjust for the category imbalance of samples. For the

spatial imbalance of the detection task, some data augmentation techniques (Pan et al., 2018; Kisantal et al., 2019) can be used to deal with this problem. For example, copy the small objects and paste them at different positions in the image multiple times to increase the proportion of small objects. Besides, data augmentation of training samples can also be performed through image fusion (Li et al., 2013; Xu et al., 2013) and image adversarial generation (Fang et al., 2020) techniques. This expansion method can alleviate the imbalance of samples in space. For the imbalance between foreground and background, the weight of difficult and easy samples can be adjusted reference to the theoretical method Focal Loss (Lin et al., 2017b). The main idea of this method is to use an appropriate function to measure the contribution of hard-to-classify and easy-to-classify samples to the total loss for a better training effect.

### 3 PROPOSED METHOD

#### 3.1 Overall Structure of Our Model

Figure 1 shows the structure of our detection model. In our new model, the network structure of the model is mostly like a general faster-rcnn structure. Our design is that the “RPN Header” module and the “ROI Header” module in the detection will output the coordinate of the detect box which will be put in the calculation process of the loss function. We modified the calculation of the loss function in a general two-stage detection model, using the size of the “Detect box” as a weight factor for the training loss contributed by each detect box. Through the computational design of the weight factor, we can appropriately adjust the loss contribution of each object with different sizes, and then improve the detection effect of small objects during the training process.

In the structure of our detection model, there are four loss functions that can be adjusted in this way, which correspond to the training effect of all classification task and localization task in the two-stage detection model. As the four new loss function shown in the right of Figure 1, these components use the weight factors to adjust the contribution of each detect box to the corresponding original loss.

The total loss of the detection is calculated by weighting the above four components and setting a certain weight coefficient for the original Smooth L1 Loss (Ren et al., 2015) and Cross Entropy Loss (Ren et al., 2015). For different application scenarios, the PLB (Pixel Level Balancing) operation can selectively adopt a combination strategy of these four new loss functions.

In our detection model, the training loss for each image is defined as:

$$L(p_i, t_i) = \frac{1}{N_{cls\_rpn}} \sum_i^n Plb_i L_{cls\_rpn}(p_i, p_i^*) + \lambda \frac{1}{N_{reg\_rpn}} \sum_i^n p_i^* Plb_i L_{reg\_rpn}(t_i, t_i^*) + \frac{1}{N_{cls\_roi}} \sum_i^n Plb_i L_{cls\_roi}(p_i, p_i^*) + \lambda \frac{1}{N_{reg\_roi}} \sum_i^n Plb_i L_{reg\_roi}(t_i, t_i^*) \quad (1)$$

where  $p_i$  and  $t_i$  are the predicted category and position results,  $\lambda$  is a parameter to weight the classification and the localization task.

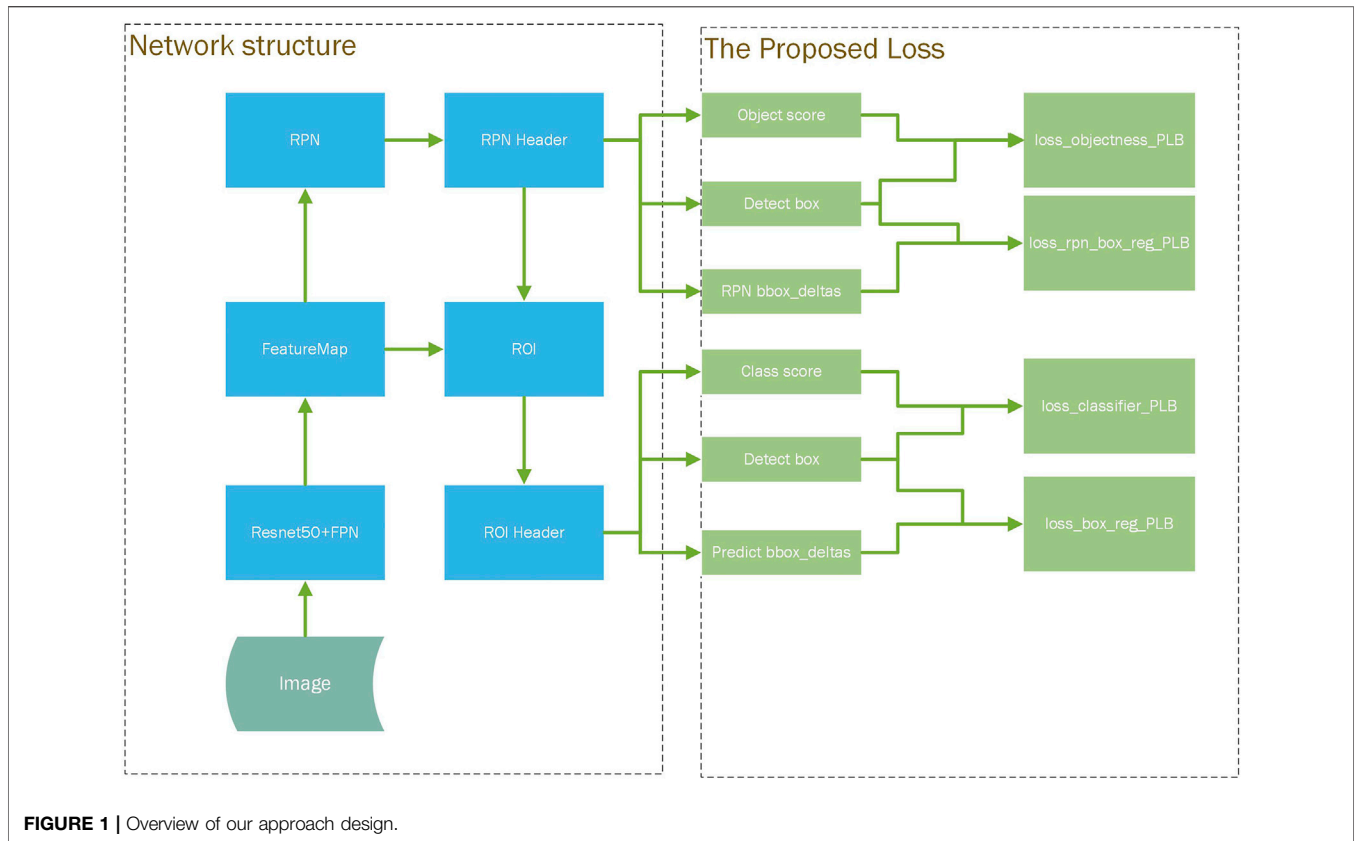


FIGURE 1 | Overview of our approach design.

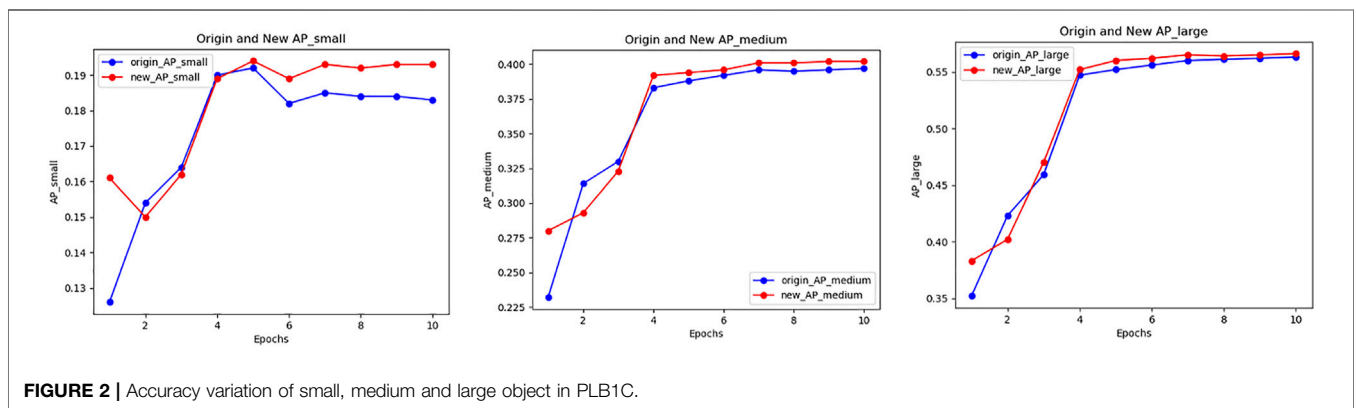


FIGURE 2 | Accuracy variation of small, medium and large object in PLB1C.

In the RPN stage, only the loss of the foreground object is calculated. The total loss function can optionally use the new loss function weighted by PLB weight factors to replace the original loss function components. If using the original loss function, just set the PLB weights to 1.

### 3.2 Design of Pixel Level Balance Factor

In the training process of the detection model, due to the different sizes of the objects, the sensitivity of the training to the size of the objects is different, and there exists a potential mutual interference. Pixel level balance refers to adjust the weight

coefficients for the training loss caused by each object under inspection when multiple objects appear in the same image, and considering their different sizes as a factor to change their mutual interference. In particular, it can be assumed that large objects will adversely affect the detection of small objects, so that in the model, the detection accuracy of small objects is further reduced. On the contrary, we can actively guide the model to change towards the optimization of small object detection by adjusting the weight coefficient of each inspected object.

The number of pixels of the inspected object can be used to measure the sensitivity of its size to detection. During the training

process, the size of each object is measured by the specific rectangular box area. For the selection of the rectangular box, the predicted box in the model training process can be used, or the ground-truth labeled box that best matches the candidate box can be used, they can be used as the representation of the object size.

The pixel level balance factor is defined as follows:

$$\text{area\_mean} = \frac{\sum_i^n \text{box\_area}}{n} \quad (2)$$

$$\text{PLB\_weight} = \frac{\text{area\_mean} * 2}{\text{area\_predict} + \text{area\_mean}} \quad (3)$$

where  $n$  is the number of detection boxes after filtered for the loss in this training, “area\_predict”: the area of the predicted box or labeled box.

If the contribution of the loss is determined according to the number of pixels, it can be considered that an object of average size has a balance factor of 1. Taking the detection accuracy of small objects as the goal, in the above formula, when “area\_predict” approaches to 0, PLB value is equal to 2, which increases the weight coefficient of small objects; when “area\_predict” approaches to the largest object among the  $n$  inspected objects, assumed that sizes of the rest objects is close to 0, then

$$\text{PLB\_weight} = \frac{2 * \frac{\text{area\_max}}{n}}{\frac{\text{area\_max}}{n} + \text{area\_max}} = \frac{2}{n + 1} \quad (4)$$

Obviously, when there is only one object under detection, that means  $n$  is 1, then PLB weight is 1. Larger the object is, the PLB weight of the object is getting smaller.

The value scope of the PLB factor is in  $(2/(n+1), 2)$ , and the object with the average size has a corresponding weight of 1. It can be considered that after adding the pixel balance factor as a weight, each inspected object, regardless of its size, will contribute equally to the loss function. Using such a design, the training of the new model has a better effect compared to the original method on optimizing the detection accuracy of small objects.

### 3.3 Loss Function Combined With Pixel Level Balance

Taking “faster\_resnet50\_fpn” as a basic model, for its classification loss and border regression loss, PLB operations can be integrated in the four loss functions.

For the loss of classification, we still use the Cross Entropy Loss function as the loss standard of the model. But we need to calculate the pixel level balancing factor according to the size of the object corresponding to each detected box. Then we use it as the weight for multiple classification loss in a batch of images. The pixel level balancing factor can be calculated by the predicted box or its corresponding labeled box. The implementation logic of the function is shown in Algorithm 1.

#### Algorithm 1. Cross Entropy Loss With Pixel Level Balancing

Algorithm 1 Cross Entropy Loss with pixel level balancing

**Input:**

$P$ : The predict score of proposals  
 $T$ : The proposal class label of matched target  
 $B$ : The detect box of proposals, predict\_box or target\_box

**Output:**

$l_{plb}$ : new loss value after PLB operation  
 1. initialize  $l_{total} = 0$ ,  $n_{boxes}$  = number of detect boxes,  $S$  = areas of  $B$  ;  
 2.  $s_u = \frac{1}{n_{boxes}} \sum_i s_i$  ; //compute the average area of detect boxes  
 3. **for each**  $i \in [1, n_{boxes}]$  **do**  
 4.  $l_i = CE(p_i, t_i)$ ; // compute Cross Entropy Loss for  $i$ -th detect\_boxes,  $p_i$  is the  $i$ -th predict score and  $t_i$  is the  $i$ -th target label  
 5.  $w_i = \frac{2s_u}{s_u + s_i}$ ; // compute the PLB weight for the  $i$ -th detect\_boxes  
 6.  $l_i' = w_i l_i$ ; //compute new loss value after PLB operation for the  $i$ -th detect\_boxes  
 7.  $l_{total} = l_{total} + l_i'$  ;  
 8. **end for**  
 9.  $l_{plb} = l_{total} / n_{boxes}$  ;  
 10. **return**  $l_{plb}$  ;

For the border regression loss function, Smooth L1 Loss is also used as the loss standard of the model, but it is necessary to calculate the pixel level balance factor according to the size of objects corresponding to each predicted box, and then calculate the pixel level balance factor of multiple objects of different sizes within a batch of images. Then we use it as the weight for localization loss for each object in our new function. The specific implementation process is shown in Algorithm 2.

#### Algorithm 2. Smooth L1 Loss With Pixel Level Balancing

Algorithm 2 Smooth L1 Loss with pixel level balancing

**Input:**

$P$ : The predict position of proposals  
 $T$ : The position label of matched target  
 $B$ : The detect box of proposals, predict\_box or target\_box  
 $\sigma$ : A parameter for Smooth L1 Loss

**Output:**

$l_{plb}$ : new loss value after PLB operation  
 1. initialize  $l_{total} = 0$ ,  $n_{boxes}$  = number of detect boxes,  $S$  = areas of  $B$  ;  
 2.  $s_u = \frac{1}{n_{boxes}} \sum_i s_i$  ; //compute the average area of detect boxes  
 3. **for each**  $i \in [1, n_{boxes}]$  **do**  
 4.  $l_i = L1_{smooth}(p_i, t_i, \sigma)$ ; // compute Smooth L1 Loss for  $i$ -th detect\_boxes,  $p_i$  is the  $i$ -th predict position and  $t_i$  is the  $i$ -th target position label  
 5.  $w_i = \frac{2s_u}{s_u + s_i}$ ; // compute the PLB weight for the  $i$ -th detect\_boxes  
 6.  $l_i' = w_i l_i$ ; //compute new loss value after PLB operation for the  $i$ -th detect\_boxes  
 7.  $l_{total} = l_{total} + l_i'$  ;  
 8. **end for**  
 9.  $l_{plb} = l_{total} / n_{boxes}$  ;  
 10. **return**  $l_{plb}$  ;

It can be seen that in the new loss function, the prediction information of the box is also added to the calculation of the classification loss, so that the classification loss and the width and height of the detected box have a certain correlation. The border regression loss will focus on the coordinate position of the border and its width and height at the same time.

### 3.4 Proposed Framework MindSpore

We implement our PLB method in PyTorch for research and exploration. At the same time, we recommend using an implementation version under the MindSpore framework as the final application. MindSpore is an enterprise-level application framework based on Huawei’s AI ecosystem. It has been used by Huawei in the medical field, and has open sourced the code of many detection models. This framework is an open-sourced product in the AI field that Huawei has been promoting in the past two years. Based on the hardware environment of Huawei’s Ascend series chips, it optimizes a large number of



**TABLE 1** | Training accuracy effect of PLB2C (IOU = 0.50:0.95) on PyTorch.

	MAP	AP_small	AP_Medium	AP_large
Original faster_rcnn	48.9	18.3	39.7	56.3
PLB2C (default predict box)	46.6 (-4.70%)	19.2 (+4.92%)	38.8 (-2.27%)	53.9 (-4.26%)
PLB2C (matched labeled box)	47.7 (-2.45%)	16.8 (-8.20%)	38.3 (-3.53%)	55.1 (-2.13%)

calculations in the model, speeds up the training and inference of the model. Due to the framework is easy to develop, efficient for execution and its full scene coverage, it can bring higher application value to our PLB method.

The implementation of our method in the MindSpore framework is basically the same as that under PyTorch, but we need to do some extra processing on the PLB weight computation. In order to prevent that all predict box areas may be zero during the training process, the area value of the predict box need to add by 1 to avoid division by zero exceptions.

## 4 EXPERIMENT RESULTS ANALYSIS

We use the dataset Pascal VOC2007 to explore the right way to apply our PLB methods, and use the BCCD blood cell detection dataset (Banik et al., 2020) to verify the effectiveness of PLB for medical image detection. Through exploration the effectiveness of our methods on Pascal VOC 2007, we verify it on BCCD datasets with our implementations both on PyTorch and MindSpore (<https://www.MindSpore.cn/en>). In our comparison experiment group, we use faster-rcnn model as the reference, chose resnet50 and FPN as the backbone network, and basically set the default values in the PyTorch library for its hyper-parameters, the SGD optimizer with momentum = 0.9, and the initial learning rate lr = 0.005, the adjustment step size of the learning rate step\_size = 3.

The source code was released at: <https://gitee.com/hubindijun/faster-rcnn-plb-MindSpore.git> (MindSpore version); <https://github.com/hubindijun/faster-rcnn-plb-PyTorch.git> (PyTorch version).

### 4.1 Exploration of PLB in Natural Image Detection

The experiment uses the Pascal VOC2007 dataset (5011 images for training and 4,952 images for validation, 20 different categories). Then we evaluate our method with the coco evaluation standard. Finally, we mainly focus on the MAP and the detection accuracy of objects with different sizes to analyze the effect of PLB. The area range of small objects is  $[0, 32 \times 32]$ , the area range of medium-sized objects is  $[32 \times 32, 96 \times 96]$  and the area range of large objects is greater than  $96 \times 96$ , using pixel point number as the unit of object size. In the dataset, the ratio of small, medium and large objects is 845:2,698:4,301 in the training dataset, while the ratio in the testing dataset is 909:2,706:4,203.

We selectively perform PLB operations on different parts of the loss function. When the training epoch is 10, both the original model and new one reach a status of convergence.

PLB method in the four different loss components are named as follows, all of the four PLB operations use predict box as the default standard for size representation.

**PLB1C:** PLB in the first RPN stage of the detection model within coarse-grained classification loss;

**PLB1B:** PLB in the first RPN stage of the detection model within bounding box regression loss;

**PLB2C:** PLB in the second stage of the detection model within fine-grained classification loss;

**PLB2B:** PLB in the second stage of the detection model within further bounding box regression loss.

#### 4.1.1 The Selection of Predict Box or Matched Labeled Box for Size Representation

Firstly, we conduct the two experiments about PLB2C with default predict box and matched labeled box as the standard for size representation. The training accuracy effect are showed in **Table 1**. The results of the PLB2C shows that only use PLB operation in the fine-grained classification loss can significantly improve the detection accuracy of small objects, but the overall accuracy of the model is reduced. PLB2C means higher requirements for small objects and reduces the expectation of the detection effect of medium and large objects. Although the detection accuracy of small objects gets improved, the detection accuracy of medium and large ones will decrease. Finally, due to the proportion of small objects is relatively small in the dataset, the overall detection accuracy will also decrease in the training.

However, after replacing predict box with matched labeled box as the representation of the object size, the detection effect is reduced, even the detection effect for small objects is reduced by 8.2% as shows in **Table 1**. We can draw a conclusion that compare to the predict box, the matched labeled box is not suitable for the representation of the object size in the model training process.

#### 4.1.2 Ablation Experiments Analysis

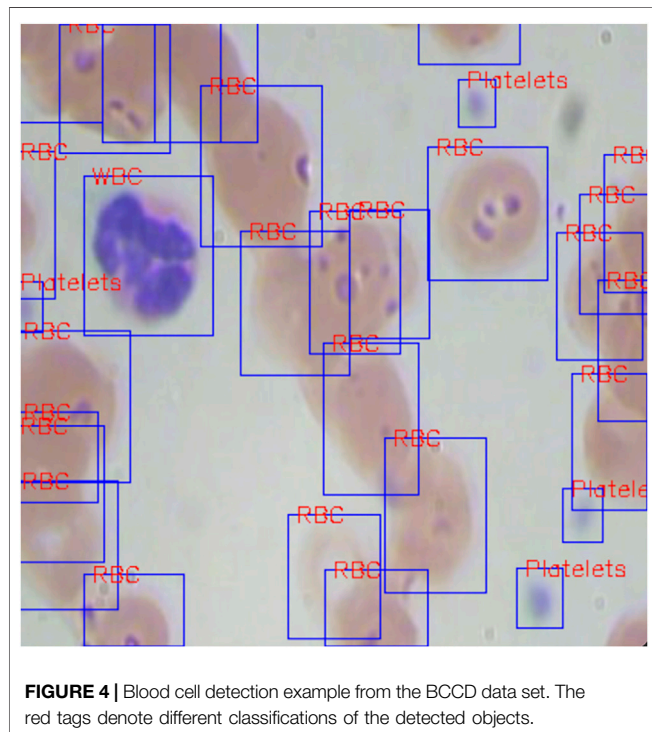
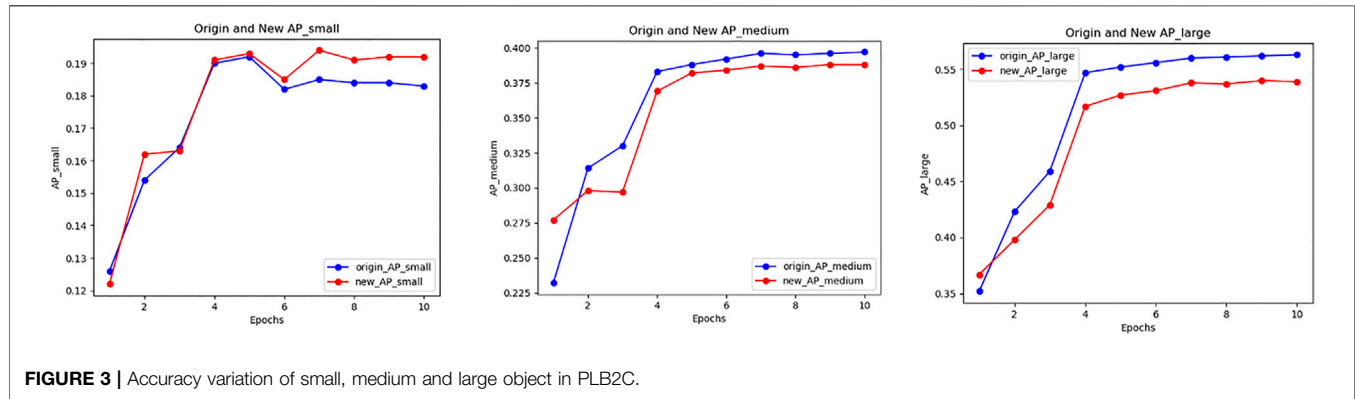
Through our design and experiments, we summarize the detection effects of each scheme on the accuracy of small objects, as shown in **Table 2**. Comparing the results of each scheme, all of the PLB methods can obviously improve the detection accuracy of small objects. However, the detect effect of the PLB methods is different for medium and large objects.

With method PLB1C or PLB1B, the overall effect of the model keeps well, especially the detection accuracy of small objects has a significant improvement. Meanwhile, the methods have little impact on medium and large objects. The training accuracy tendency of PLB1C is shown in **Figure 2**.

PLB operations in the second stage also improve the detection accuracy of small objects, as the results of PLB2C and PLB2B.

**TABLE 2** | Comparison of the PLB methods (IOU = 0.50:0.95) on PyTorch.

PLB1C	PLB1B	PLB2C	PLB2B	MAP	AP_small	AP_medium	AP_large
Not used	-	-	-	48.9	18.3	39.7	56.3
✓	-	-	-	49.3	19.3 (+5.46%)	40.2	56.5
-	✓	-	-	49.2	19.1 (+4.37%)	39.9	56.8
-	-	✓	-	46.6	19.2 (+4.92%)	38.8	53.9
-	-	-	✓	48.7	19.0 (+3.83%)	39.2	56.2
✓	-	✓	-	49.3	18.6 (+1.64%)	39.9	56.7
✓	✓	-	-	49.1	19.3 (+5.46%)	39.9	56.5



However, that methods have negative impact on detection of medium and large objects. Due to there are more samples of medium and large objects than small ones, the overall detection accuracy is not well. As the accuracy tendency of PLB2C shows us in **Figure 3**.

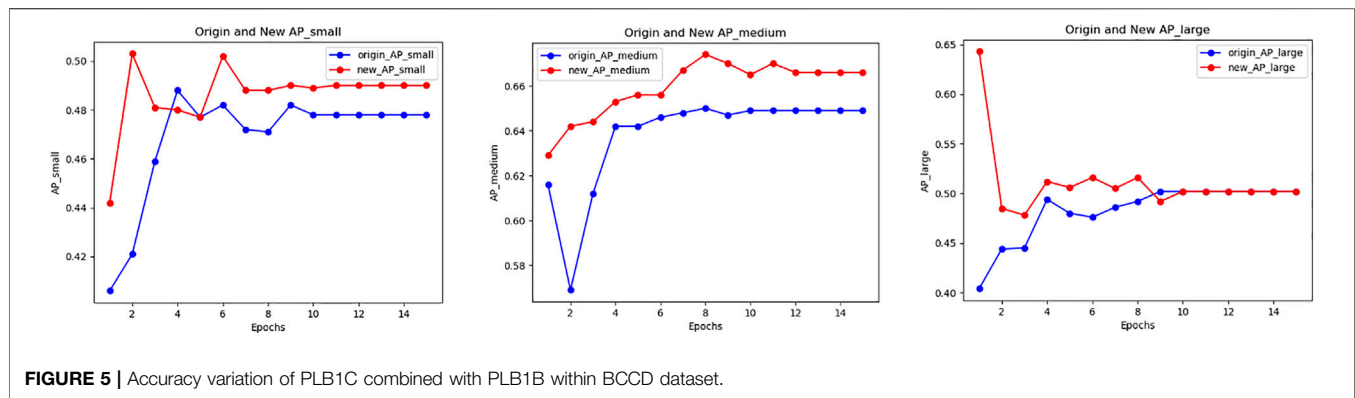
**TABLE 3** | Accuracy effect with PLB in BCCD Dataset (IOU = 0.50:0.95) on PyTorch.

	MAP	AP_small	AP_Medium	AP_large
Origin model	63.1	47.8	64.9	50.2
PLB1C	63.6	49.7 (+3.97%)	65.2 (+0.46%)	49.6 (-1.20%)
PLB1B	63.1	48.6 (+1.67%)	64.0 (-1.40%)	50.6 (+0.80%)
PLB1C + PLB1B	63.7	49.0 (+2.51%)	66.6 (+2.62%)	50.2 (+0%)

**TABLE 4** | Accuracy effect with PLB in BCCD Dataset (IOU = 0.50:0.95) on MindSpore.

	MAP	AP_small	AP_Medium	AP_large
Origin model	62.9	48.7	65.4	48.0
PLB1C	63.3	51.4 (+5.54%)	65.6 (+0.31%)	48.3 (+0.63%)
PLB1B	63.1	49.6 (+1.85%)	64.2 (-1.83%)	50.3 (+4.79%)
PLB1C + PLB1B	63.6	50.4 (+3.49%)	66.6 (+1.83%)	49.2 (+2.50%)

Through the different results of the PLB methods, we can draw a conclusion that in the training process of the two-stage detector, the PLB operation utilized in the RPN stage can improve the detection effect of small objects and maintain the accuracy of medium and large objects. The purpose of PLB operation is to balance the contribution to the model loss of objects which have different sizes during the training process. In particular, we can adjust the design of the PLB factor so that the training of the model is transformed towards the detection accuracy improvement of small objects.



So how does PLB methods be more effective in the first stage? We guess that in the first stage of the training process, the coarse-grained classification task is mainly to classify the inspected objects as foreground or background, among them the background will not be included in the subsequent loss calculation. The smaller the object is, the easier it is to be misclassified as a background. So that in the second stage of detection, it is no longer involved in training process. Therefore, PLB method has a relatively obvious effect in the RPN stage of the two-stage detection model. Moreover, compared with the transformation of border regression loss with PLB, the loss transformation effect for the classification with PLB is more effective.

## 4.2 Practice of PLB in Medical Image Detection

We have verified that PLB has a certain adjustment effect for object detection, and using PLB in the RPN stage is more effective. Our design can also be used in specific medical application scenarios, such as routine blood testing and breast cancer diagnosis through lymphocyte detection. All of these application scenarios are based on detecting and measuring various types of blood cells to assist disease diagnosis. We use the BCCD data set (765 pictures for training, and 73 pictures for evaluating) to check the effect of PLB in cell detection. **Figure 4** shows the blood cell detection in general, where there are three different cell types, in which the platelet size is relatively small and hard to detect.

We carried out three sets of experiments with this dataset. Using PLB with the model only in the coarse-grained classification loss, or only in the RPN border regression loss, or both of them at the same time, to demonstrate the effectiveness of it in medical application scenarios. When the training epochs is 15, both the old and new models reach to convergence. We conduct the experiments with our implementation both on PyTorch version and MindSpore version. **Tables 3** and **Table 4** shows the final results of pixel level balance respectively to these three experiments.

Experimental results show that the PLB methods can effectively improve the detection effect of small objects in the process of medical image detection tasks. When using PLB in the two loss functions in the RPN stage at the same time, the overall

detection effect is improved, especially for the detection accuracy of small objects. **Figure 5** shows the accuracy variation of small, medium and large object when using PLB in the two loss components in the RPN stage.

## 5 CONCLUSIONS AND OUTLOOK

In this paper, we have proposed pixel level balance different from previous research, which focuses in the correlation of large and small objects in the training process. This method can be combined with other effective methods to improve small object detection, such as FPN network to improve the feature extraction, or data augmentation on the input dataset samples, etc. In some specific application scenarios, pixel level balance can provide more special effects. Obviously, in a train model, we can improve the detection accuracy of small objects while ignoring the large one by modifying the design of pixel level balance factors.

Pixel level balance can perform well in the problem of higher requirements for small objects in medical image detection. In our experiments, the effectiveness of this method for blood cell detection tasks has been demonstrated. It can be used in more other medical detection tasks in the future and achieve more development space or commercial value to medical image detection technology.

For future work, to alleviate the strong label requirement for deep learning-based detection, we would like to explore the possible way of applying visual matching-based approaches (Jiang et al., 2021a) for object detection and recognition. One promising technique is adopting graph matching with (higher-order) structure information (Yan et al., 2018) which can be more generalizable to new objects, and the detection may be performed in a joint matching fashion with multiple candidate objects with different techniques from heuristic optimization (Yan et al., 2015; Yan et al., 2016a) to dynamic programming based one (Jiang et al., 2021b). Moreover, the recently developed deep learning-based graph matching models (Wang et al., 2020; Wang et al., 2021) can also be explored which can better model the visual features for matching and object recognition. Readers are referred to the survey papers for more comprehensive study of these areas, in terms of both traditional learning-free methods (Yan et al., 2016b) as well as deep learning models (Yan et al., 2020). The hope is that a more



structure information can be effectively used for object detection, against outliers, deformation, occlusion and other noise.

## DATA AVAILABILITY STATEMENT

Publicly available datasets were analyzed in this study. This data can be found here: Pascal VOC2007 [http://host.robots.ox.ac.uk/pascal/VOC/voc2007/VOCtrainval\\_06-Nov-2007.tar](http://host.robots.ox.ac.uk/pascal/VOC/voc2007/VOCtrainval_06-Nov-2007.tar) BCCD <https://public.roboflow.com/object-detection/bccd/4/download>.

## AUTHOR CONTRIBUTIONS

BH and YL contributed to the conception and design of the study. BH wrote the first draft of the manuscript. PC, MT and QK

## REFERENCES

- Banik, P. P., Saha, R., and Kim, K.-D. (2020). An Automatic Nucleus Segmentation and CNN Model Based Classification Method of White Blood Cell. *Expert Syst. Appl.* 149, 113211. doi:10.1016/j.eswa.2020.113211
- Bochkovskiy, A., Wang, C. Y., and Liao, H. Y. M. (2020). *Yolov4: Optimal Speed and Accuracy of Object Detection*, Ithaca: Cornell University, <http://arXiv.org/abs/2004.10934>.
- Changsheng, C., Xiangfeng Wang, X., Weishan Dong, W., Junchi Yan, J., Qingshan Liu, Q., and Hongyuan Zha, H. (2019). Joint Active Learning with Feature Selection via Cur Matrix Decomposition. *IEEE Trans. Pattern Anal. Mach. Intell.* 41 (6), 1382–1396. doi:10.1109/TPAMI.2018.2840980
- Ciresan, D. C., Giusti, A., Gambardella, L. M., and Schmidhuber, J. (2013). Mitosis Detection in Breast Cancer Histology Images with Deep Neural Networks." in International Conference on Medical Image Computing and Computer-Assisted Intervention, Springer, Berlin, Heidelberg, September, 2013 Editors H. Ishikawa, C. Lin Liu, T. Pajdla, and S. Jianbo 411 Berlin Heidelberg: Springer.
- Dai, J., Li, Y., He, K., and Sun, J. (2016). "R-fcn: Object Detection via Region-Based Fully Convolutional Networks," in Advances in Neural Information Processing Systems, Barcelona, Spain, December, 2021 29, 379–387.
- Fang, Y., Deng, W., Du, J., and Hu, J. (2020). Identity-aware CycleGAN for Face Photo-Sketch Synthesis and Recognition. *Pattern Recognit.* 102, 107249. doi:10.1016/j.patcog.2020.107249
- Fu, C. Y., Liu, W., Ranga, A., Tyagi, A., and Berg, A. C. (2017). *Dssd: Decorvolutional Single Shot Detector*. Ithaca: Cornell University. <http://arXiv.org/abs/arXiv:1701.06659>.
- Fujita, S., and Han, X. H. (2020). "Cell Detection and Segmentation in Microscopy Images with Improved Mask R-CNN," in Proceedings of the Asian Conference on Computer Vision, December, 2020. Online Meeting.
- Girshick, R., Donahue, J., Darrell, T., and Malik, J. (2014). "Rich Feature Hierarchies for Accurate Object Detection and Semantic Segmentation," in Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Columbus, OH, June, 2014 (New Jersey: IEEE), 580–587. doi:10.1109/cvpr.2014.81
- Girshick, R. (2015). "Fast R-Cnn," in Proceedings of the IEEE International Conference on Computer Vision, Santiago, December, 2015 (New Jersey: IEEE), 1440–1448. doi:10.1109/iccv.2015.169
- Golan, R., Jacob, C., and Denzinger, J. (2016). "Lung Nodule Detection in CT Images Using Deep Convolutional Neural Networks," in 2016 International Joint Conference on Neural Networks (IJCNN), Vancouver, BC, July, 2016 (New Jersey: IEEE), 243–250. doi:10.1109/ijcnn.2016.7727205
- Handels, H., Mersmann, S., Palm, C., Tolxdorff, T., Wagenknecht, G., and Wittenberg, T. (2013). Viewpoints on Medical Image Processing: from Science to Application. *Curr. Med. Imaging* 9 (2), 79. doi:10.2174/1573405611309020002
- He, K., Zhang, X., Ren, S., and Sun, J. (2016). "Deep Residual Learning for Image Recognition," in Proceedings of the IEEE Conference on Computer

reviewed and revised the manuscript. All authors contributed to manuscript revision and approved the submitted version.

## FUNDING

This research was partly funded by the Interdisciplinary Program of Shanghai Jiao Tong University (YG2021QN67), CAAI-Huawei MindSpore Open Fund (CAIXSJLJJ-2020-022A).

## ACKNOWLEDGMENTS

The authors would like to appreciate the Student Innovation Center of SJTU for providing GPUs.

- Vision and Pattern Recognition (USA: IEEE), 770–778. doi:10.1109/cvpr.2016.90
- He, K., Zhang, X., Ren, S., and Sun, J. (2015). Spatial Pyramid Pooling in Deep Convolutional Networks for Visual Recognition. *IEEE Trans. Pattern Anal. Mach. Intell.* 37 (9), 1904–1916. doi:10.1109/tpami.2015.2389824
- Hu, J., Shen, L., and Sun, G. (2018). "Squeeze-and-excitation Networks," in Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Salt Lake, June, 2018 (USA: IEEE), 7132–7141. doi:10.1109/cvpr.2018.00745
- Huang, G., Liu, Z., Van Der Maaten, L., and Weinberger, K. Q. (2017). "Densely Connected Convolutional Networks," in Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, HI, July, 2017 (USA: IEEE), 4700–4708. doi:10.1109/cvpr.2017.243
- Janowczyk, A., and Madabhushi, A. (2016). Deep Learning for Digital Pathology Image Analysis: A Comprehensive Tutorial with Selected Use Cases. *J. Pathol. Inf.* 7, 29. doi:10.4103/2153-3539.186902
- Jeong, J., Park, H., and Kwak, N. (2017). *Enhancement of SSD by Concatenating Feature Maps for Object Detection*. Ithaca: Cornell University. <http://arXiv.org/abs/arXiv:1705.09587>.
- Jiang, X., Fan, A., Jiang, J., and Yan, J. (2021). Image Matching from Handcrafted to Deep Features: A Survey. *Int. J. Comput. Vis.* 129, 23–79. doi:10.1007/s11263-020-01359-2
- Jiang, Z., Wang, T., and Yan, J. (2021). Unifying Offline and Online Multi-Graph Matching via Finding Shortest Paths on Supergraph. *IEEE Trans. Pattern Anal. Mach. Intell.* 43 (10), 3648–3663. doi:10.1109/tpami.2020.2989928
- Kisantal, M., Wojna, Z., Murawski, J., Naruniec, J., and Cho, K. (2019). *Augmentation for Small Object Detection*, Ithaca: Cornell University, <http://arXiv.org/abs/arXiv:1902.07296>.
- Li, C., Li, R., Yuan, Y., Wang, G., and Xu, D. (2021). Deep Unsupervised Active Learning via Matrix Sketching. *IEEE Trans. Image Process.* 30, 9280–9293. doi:10.1109/tip.2021.3124317
- Li, C., Ma, H., Yuan, Y., Wang, G., and Xu, D. (2022). Structure Guided Deep Neural Network for Unsupervised Active Learning. *IEEE Trans. Image Process.*, 2767–2781. doi:10.1109/tip.2022.3161076
- Li, Q., Du, J., Song, F., Wang, C., Liu, H., and Lu, C. (2013). "Region-Based Multi-Focus Image Fusion Using the Local Spatial Frequency," in 2013 25th Chinese Control and Decision Conference (CCDC), Guiyang, GuiZhou, May, 2013 (USA: IEEE), 3792.
- Li, X., Chen, S., Hu, X., and Yang, J. (2019). "Understanding the Disharmony between Dropout and Batch Normalization by Variance Shift," in Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (USA: IEEE), 2682–2690. doi:10.1109/cvpr.2019.00279
- Li, Z., and Zhou, F. (2017). "FSSD: Feature Fusion Single Shot Multibox Detector," in (Ithaca: Cornell University). <http://arXiv.org/abs/arXiv:1712.00960>.
- Lin, T. Y., Dollár, P., Girshick, R., He, K., Hariharan, B., and Belongie, S. (2017a). "Feature Pyramid Networks for Object Detection," in Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, HI, July, 2017 (USA: IEEE), 2117–2125. doi:10.1109/cvpr.2017.106

- Lin, T. Y., Goyal, P., Girshick, R., He, K., and Dollár, P. (2017b). "Focal Loss for Dense Object Detection," in Proceedings of the IEEE International Conference on Computer Vision, Venice, Italy, October, 2017 (USA: IEEE), 2980–2988. doi:10.1109/iccv.2017.324
- Litjens, G., Kooi, T., Bejnordi, B. E., Setio, A. A. A., Ciompi, F., and Ghafoorian, M. (2017). A Survey on Deep Learning in Medical Image Analysis. *Med. Image Anal.* 42, 60–88. doi:10.1016/j.media.2017.07.005
- Liu, W., Angelov, D., Erhan, D., Szegedy, C., Reed, S., Fu, C.-Y., et al. (2016). "SSD: Single Shot MultiBox Detector," in European Conference on Computer Vision, Springer, Cham, October, 2016 (Cham: Springer), 21–37. doi:10.1007/978-3-319-46448-0\_2
- McInerney, T., and Terzopoulos, D. (1996). Deformable Models in Medical Image Analysis: a Survey. *Med. Image Anal.* 1 (2), 91–108. doi:10.1016/s1361-8415(96)80007-7
- Neubeck, A., and Van Gool, L. (2006). Efficient Non-maximum Suppression." in 18th 457 International Conference on Pattern Recognition (ICPR'06), Hong Kong, China, August, 2006. *Sarcoidosis* 3, 101.
- Pan, X., Yang, D., Li, L., Liu, Z., Yang, H., and Cao, Z. (2018). Cell Detection in Pathology and Microscopy Images with Multi-Scale Fully Convolutional Neural Networks. *World Wide Web* 21 (6), 1721–1743. doi:10.1007/s11280-017-0520-7
- Qian, W., Yang, X., Peng, S., Guo, Y., and Yan, J. (2019). *Learning Modulated Loss for Rotated Object Detection*, <http://arXiv.org/abs/1911.08299>. United States: arxiv.
- Redmon, J., Divvala, S., Girshick, R., and Farhadi, A. (2016). "You Only Look once: Unified, Real-Time Object Detection," in Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, NV, June, 2016 (USA: IEEE), 779–788. doi:10.1109/cvpr.2016.91
- Redmon, J., and Farhadi, A. (2017). "YOLO9000: Better, Faster, Stronger," in Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, HI, July, 2017 (USA: IEEE), 7263–7271. doi:10.1109/cvpr.2017.690
- Redmon, J., and Farhadi, A. (2018) *Yolov3: An Incremental Improvement*. Ithaca: Cornell University, <http://arXiv.org/abs/1804.02767>.
- Ren, S., He, K., Girshick, R., and Sun, J. (2015). Faster R-Cnn: Towards Real-Time Object Detection with Region Proposal Networks. *Adv. neural Inf. Process. Syst.* 28, 91–99. doi:10.5555/2969239.2969250
- Roth, H. R., Lu, L., Liu, J., Yao, J., Seff, A., Cherry, K., et al. (2016). Improving Computer-Aided Detection Using Convolutional Neural Networks and Random View Aggregation. *IEEE Trans. Med. Imaging* 35 (5), 1170–1181. doi:10.1109/tmi.2015.2482920
- Shen, Z., Liu, Z., Li, J., Jiang, Y. G., Chen, Y., and Xue, X. (2017). "Dsod: Learning Deeply Supervised Object Detectors from Scratch," in Proceedings of the IEEE International Conference on Computer Vision, Venice, Italy, October, 2017 (USA: IEEE), 1919–1927. doi:10.1109/iccv.2017.212
- Shrivastava, A., Sukthankar, R., Malik, J., and Gupta, A. (2016). *Beyond Skip Connections: Top-Down Modulation for Object Detection*. Ithaca: Cornell University. <http://arXiv.org/abs/1612.06851>
- Tan, M., and Le, Q. (2019). "Efficientnet: Rethinking Model Scaling for Convolutional Neural Networks," in International Conference on Machine Learning, California, United States, June, 2019 (USA: IEEE), 6105.
- Tan, M., Pang, R., and Le, Q. V. (2020). "Efficientdet: Scalable and Efficient Object Detection," in Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Seattle, WA, June, 2020 (USA: IEEE), 10781–10790. doi:10.1109/cvpr42600.2020.01079
- Wang, R., Yan, J., and Yang, X. (2020). Combinatorial Learning of Robust Deep Graph Matching: an Embedding Based Approach. *IEEE Trans. Pattern Anal. Mach. Intell.* 42 (6), 1–16. doi:10.1109/TPAMI.2020.3005590
- Wang, R., Yan, J., and Yang, X. (2021). Neural Graph Matching Network: Learning Lawler's Quadratic Assignment Problem with Extension to Hypergraph and Multiple-Graph Matching." in *IEEE Transactions on Pattern Analysis and Machine Intelligence*. USA: IEEE.
- Xu, L., Du, J., and Li, Q. (2013). Image Fusion Based on Nonsubsampled Contourlet Transform and Saliency-Motivated Pulse Coupled Neural Networks. *Math. Problems Eng.* 2013 (3), 1–10. doi:10.1155/2013/135182
- Yan, J., Cho, M., Zha, H., Yang, X., and Chu, S. M. (2016a). Multi-Graph Matching via Affinity Optimization with Graduated Consistency Regularization. *IEEE Trans. Pattern Anal. Mach. Intell.* 38 (6), 1228–1242. doi:10.1109/tpami.2015.2477832
- Yan, J., Li, C., Li, Y., and Cao, G. (2018). Adaptive Discrete Hypergraph Matching. *IEEE Trans. Cybern.* 48 (2), 765–779. doi:10.1109/tcyb.2017.2655538
- Yan, J., Wang, J., Zha, H., Yang, X., and Chu, S. (2015). Consistency-Driven Alternating Optimization for Multigraph Matching: A Unified Approach. *IEEE Trans. Image Process.* 24 (3), 994–1009. doi:10.1109/tip.2014.2387386
- Yan, J., Yang, S., and Hancock, E. (2020). *Learning Graph Matching and Related Combinatorial Optimization Problems, International Joint Conferences on Artificial Intelligence*. USA: ACM, 4988
- Yan, J., Yin, X., Lin, W., Deng, C., Zha, H., and Yang, X. (2016b). A Short Survey of Recent Advances in Graph Matching. ACM International Conference on Multimedia Retrieval, New York, NY, June, 2016, 167–174. doi:10.1145/2911996.2912035
- Yang, S., Fang, B., Tang, W., Wu, X., Qian, J., and Yang, W. (2017). "December) Faster R-CNN Based Microscopic Cell Detection," in 2017 International Conference on Security, Pattern Analysis, and Cybernetics (SPAC) (New Jersey: IEEE), 345.
- Yang, X., Hou, L., Zhou, Y., Wang, W., and Yan, J. (2021c). "Dense Label Encoding for Boundary Discontinuity Free Rotation Detection," in Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, June, 2021 (USA: IEEE), 15819–15829. Online Meeting. doi:10.1109/cvpr46437.2021.01556
- Yang, X., Liu, Q., Yan, J., Li, A., Zhang, Z., and Yu, G. (2019b). "R3det: Refined Single-Stage Detector with Feature Refinement for Rotating Object," in Advances in Neural Information Processing Systems, December, 2021. Online Meeting 2 (4), 18381–18394. doi:10.48550/ARXIV.1911.08299
- Yang, X., and Yan, J. (2020). "Arbitrary-oriented Object Detection with Circular Smooth Label," in European Conference on Computer Vision (Cham: Springer), 677–694. doi:10.1007/978-3-030-58598-3\_40
- Yang, X., Yan, J., Ming, Q., Wang, W., Zhang, X., and Tian, Q. (2021a). "Rethinking Rotated Object Detection with Gaussian Wasserstein Distance Loss," in International Conference on Machine Learning, July, 2021 (New York, NY: PMLR), 11830. Online Meeting.
- Yang, X., and Yan, J. (2022). On the Arbitrary-Oriented Object Detection: Classification Based Approaches Revisited. *Int. J. Comput. Vis.* 130 (5), 1340–1365. doi:10.1007/s11263-022-01593-w
- Yang, X., Yan, J., Yang, X., Tang, J., Liao, W., and He, T. (2020). *Scrdet++: Detecting Small, Cluttered and Rotated Objects via Instance-Level Feature Denoising and Rotation Loss Smoothing*. Ithaca: Cornell University. <http://arXiv.org/abs/2004.13316>.
- Yang, X., Yang, J., Yan, J., Zhang, Y., Zhang, T., Guo, Z., et al. (2019a). "Scrdet: Towards More Robust Detection for Small, Cluttered and Rotated Objects," in Proceedings of the IEEE/CVF International Conference on Computer Vision, Seoul, South Korea, October, 2019 (New Jersey: IEEE), 8232–8241. doi:10.1109/iccv.2019.00832
- Yang, X., Yang, X., Yang, J., Ming, Q., Wang, W., Tian, Q., et al. (2021b). Learning High-Precision Bounding Box for Rotated Object Detection via Kullback-Leibler Divergence. *Adv. Neural Inf. Process. Syst.* 34. doi:10.48550/ARXIV.2106.01883
- Yang, X., Zhou, Y., Zhang, G., Yang, J., Wang, W., Yan, J., et al. (2022). *The KFIOU Loss for Rotated Object Detection*. Ithaca: Cornell University, <http://arXiv.org/abs/2201.12558>.
- Zhang, X., Zhu, X., Tang, K., Zhao, Y., Lu, Z., and Feng, Q. (2022). DDTNet: A Dense Dual-Task Network for Tumor-Infiltrating Lymphocyte Detection and Segmentation in Histopathological Images of Breast Cancer. *Med. Image Anal.* 78, 102415. doi:10.1016/j.media.2022.102415

**Conflict of Interest:** Author QK was employed by Riseey Intelligent Technology (Shanghai) Co., Ltd.

The authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

**Publisher's Note:** All claims expressed in this article are solely those of the authors and do not necessarily represent those of their affiliated organizations, or those of the publisher, the editors and the reviewers. Any product that may be evaluated in this article, or claim that may be made by its manufacturer, is not guaranteed or endorsed by the publisher.

Copyright © 2022 Hu, Liu, Chu, Tong and Kong. This is an open-access article distributed under the terms of the Creative Commons Attribution License (CC BY). The use, distribution or reproduction in other forums is permitted, provided the original author(s) and the copyright owner(s) are credited and that the original publication in this journal is cited, in accordance with accepted academic practice. No use, distribution or reproduction is permitted which does not comply with these terms.