Neural Computing and Applications
https://doi.org/10.1007/s00521-020-05400-w

ORIGINAL ARTICLE

Learning region-guided scale-aware feature selection for object detection

Liu Liu[1,2] · Rujing Wang[2] · Chengjun Xie[2] · Rui Li[2] · Fangyuan Wang[1,2] · Man Zhou[1,2] · Yue Teng[1,2]

Received: 8 May 2020 / Accepted: 24 September 2020

Abstract
Scale variation is one of the major challenges in object detection task. Modern region-based object detection architectures often adopt Feature Pyramid Network (FPN) as feature extraction neck to achieve multi-scale feature representation in solving scale variation problem. However, due to the rough feature selection strategy in Region of Interest (RoI) feature extraction step, these methods might not perform well on object detection under strong scale variation. In this work, we are motivated by the limitations of current FPN-based two-stage object detectors and then present a novel module, namely scale-aware feature selective (SAFS) module, that flexibly and adaptively selects feature levels in two-stage object detectors. Specifically, we firstly build the RoI Pyramid in standard FPN structure to extract RoI features from various scale levels. Next, in order to achieve scale-aware mechanism for solving scale variation issue, we develop a novel weighting gate function containing one set of trainable parameters to automatically learn the fusion weight for each RoI feature level, which relieves the limitation of hard feature selection strategy guided by online instance size. Outputs from the RoI features with the learned weights are fused for classification and bounding box regression. Furthermore, we design a multi-level SAFS architecture to obtain different types of RoI feature combinations that ensures our method is more robust to various instance scales. Experimental results show that our SAFS module is very compatible with most of two-stage object detectors and could achieve state-of-the-art results with Average Precision of 48.3 on COCO *test-dev* and other popular object detection benchmarks. Our code will be made publicly available.

Keywords Scale variation · Object detection · RoI Pyramid · Scale-aware feature selective

1 Introduction

In modern computer vision field, Convolutional Neural Network (CNN) has shown the high efficiency on automatically extracting powerful features on various visual tasks guided by supervised learning [1–4]. The past decade has witnessed the superior performance when CNN is employed in object detection architectures [5–7]. Among these methods, two-stage approaches are fast becoming a key instrument in generic object detection task due to its high-quality candidate boxes outputted from Region Proposal Network (RPN). Furthermore, Feature Pyramid Network (FPN) [8], introduced as model neck component, can play a vital role in addressing the issue of difficulty on detecting small-scale objects. This could be explained by the multi-scale feature representation where all the levels contain strong semantic information through top-down pathway and lateral connection mechanism. Intuitively, most of the region-based architectures adopt FPN to detect objects across a large range of scales.

However, a central issue in these two-stage object detectors lies in handling scale variation problem [9], which is manifested as large number of instances with various scales (widths and heights) in one single image simultaneously. To remedy this issue, an intuitive way is to leverage FPN into two-stage object detectors, in which strong evidence shows that current methods could benefit

✉ Rujing Wang
rjwang@iim.ac.cn

[1] University of Science and Technology of China, Hefei 230026, China

[2] Institute of Intelligent Machines, Chinese Academy of Sciences, Hefei 230031, China

Published online: 10 October 2020

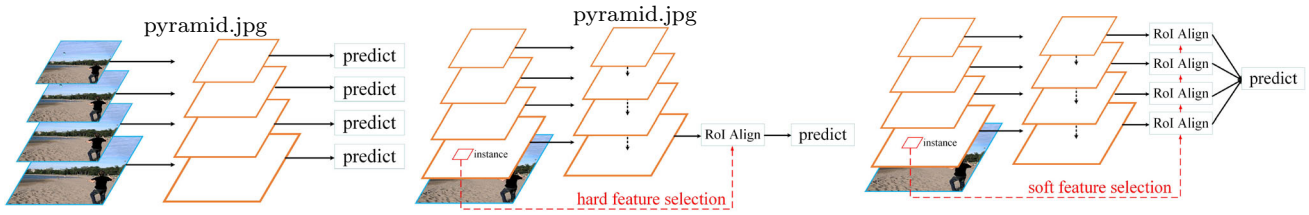
from FPN component in dealing with multi-scale object detection [10]. Generally, there is an expert consensus that objects with specific scale are supposed to be detected on single feature level. Specifically, instances with small sizes are expected to be featurized in low-level feature maps as small objects' information might vanish in high-level semantic features with large sampling stride. Following this consensus, standard FPN is associated with the increasing risk of the hard feature level selection strategy, which leads to some potential limitations. On the contrary, lots of valuable information from the other feature levels might be neglected when corresponding level is chosen for RoI Align to extract RoI feature [11]. Moreover, from our observation, R-CNN head network employed in two-stage object detectors might not achieve satisfied region classification accuracy, while Region Proposal Network (RPN) provides high-quality candidate boxes with high recall. In this case, richer contextual and semantic features are essential to obtain a better detection performance. Thus, to solve scale variation issue, RoI feature fusion might be a potential way to leverage the multi-scale feature pyramid structure.

Apart from feature pyramid architecture, current researches attempt to achieve multi-scale feature fusion from the various aspects. Featurized image pyramid is a common and simple strategy to address multi-scale issue, which utilizes brute-force data augmentation during training, including scale jittering, resizing and cropping [12]. In inference phase, Test Time Augmentation (TTA) is used to test the input image with various sizes and obtain the final result by combining them together [13]. Besides, it is also feasible to design a single model with various filters for different scales to generate the scale-aware feature maps, which are outputted from several parallel branches [14, 15]. To ensure the feature maps with different scales could be extracted, each branch owns its specific receptive field for filter. Nevertheless, due to the intra-class variance of large-scale and small-scale instances in one image, handling the different feature responses in a single feature level might be unreasonable. Therefore, it is necessary to develop a dramatic and soft feature selection and weighting mechanism to remedy the scale-variance issue.

Motivated by the idea of feature fusion, we investigate the feasibility of softly mapping and weighting feature levels to each region proposal by developing a novel architecture unit, termed scale-aware feature selective (SAFS) module guided by input instances' sizes. Our goal is to design a universal flexible and adaptive feature level selection module deployed in most of region-based object detection methods with feature pyramid structure. In order to implement this function, our SAFS module could learn to use sizes of region proposals to adaptively emphasize more informative features and suppress those containing noisy information for the specific instance. The structure comparison of our proposed SAFS with image pyramid and feature pyramid is shown in Fig. 1, which exploit features from all the levels rather than the multi-scale images or single feature level. In our method, the input image is firstly processed by a CNN backbone and FPN neck for extracting multi-scale features in various levels. Similar to standard two-stage object detection approaches, region proposals are provided by standard RPN in each level. Secondly, we build a RoI Pyramid structure that employs RoI Align operation into all the feature levels rather than harshly selecting the specific feature level. In this way, we could obtain the multi-scale RoI features. Next, in order to achieve scale-aware mechanism for solving scale variation issue, we develop a novel weighting gate function containing one trainable parameter to learn the weights for each RoI feature level, which address the limitation of hard feature selection strategy by online instance size guidance. Outputs from the RoI features with the learned weights are fused for classification and bounding box regression. Such the scale-aware weighting mechanism could be regarded as a soft activation for RoI features from different levels. Furthermore, we design the multi-level feature selective architecture to obtain the different RoI feature combinations that ensures our method is more robust to various instance scales. Therefore, in this way, SAFS module could achieve an adaptive nonlinear mapping between input region proposal and RoI feature levels, in which this relationship could be automatically learned through model training.

The major contributions of this paper are as follows:

- We propose a novel feature selective module termed as scale-aware feature selective (SAFS) module to achieve soft RoI feature fusion and weighting guided by instance sizes. This module is very compatible to be deployed into most of state-of-the-art two-stage object detectors for softly and flexibly selecting RoI features.
- Our SAFS module exploits scales of region proposals to learn the optimal weights for different feature levels. Besides, a novel weight gate function is designed to improve the robustness to scale variance, which is learned by one trainable parameter.
- The comparative and depth experiments show that our SAFS module could help to improve the performance of state-of-the-art methods in generic object detection tasks, which achieves Average Precision (AP) of 48.3 on COCO *test-dev* and other popular object detection benchmarks. Our code will be made publicly available.



(a) Image Pyramid **(b)** Feature Pyramid **(c)** Our SAFS

Fig. 1 **a** Multi-scale images are used as input, and the image pyramid methods perform feature extraction as well as object prediction at each scale. **b** Feature pyramid methods extract multi-scale feature maps and predict object using hard feature selection at single level. **c** Our proposed SAFS structure takes multi-scale RoI features into consideration by building RoI Pyramid and predicts by the result of feature fusion

2 Related work

DeepConv object detectors Convolutional Neural Network (CNN) has proved to be a predominant solution applied in generic image identification as well as object detection due to its effective high-quality feature extraction in 2D static images. As one type of these approaches, one-stage detectors target at finding the objects from the whole feature maps directly assisted through anchors with various pre-defined sizes [6, 16, 17]. Meanwhile, these anchors are refined by a parallel regression layer. Considering objects with various sizes, SSD [7] and DSSD [18] perform to find instances at multilayers. In order to improve the detection accuracy, RetinaNet [19] introduces a novel loss function named focal loss to alleviate the imbalance risk of background–foreground samples. Besides, feature pyramid structure is also employed in RetinaNet to consider both large-scale and small-scale objects. In recent years, RefineDet [20] proposes to sample and adjust the pre-defined anchors, followed by further box refinement.

Apart from one-stage detectors, two-stage architectures, also known as region-based methods, aim to follow an idea of coarse-to-fine strategy. Fast R-CNN [3] introduces Region of Interest (RoI) as region proposals for fine-tuning classification and bounding box regression, which are generated from feature maps by RoI pooling operation. However, Fast R-CNN utilizes selective search [21] to pre-compute region proposals as extra input. To solve this problem, Faster R-CNN [22] develops Region Proposal Network (RPN) to replace the stand-alone compute-intensive method and achieves the end-to-end training as well as inference. Based on Faster R-CNN, a large number of region-based approaches emerge to continually improve the object detection performance, such as R-FCN [10], Cascade R-CNN [23] and Libra R-CNN [24].

Scale-variance solving. Scale-variance problem has always been a serious challenge in modern two-stage object detection methods. Image pyramid with multi-scales is a common choice to achieve scale aware to improve detection accuracy, especially for small-scale objects. Among these methods, SNIP [13] proposes to only train the specific scale instances corresponding to each image scale but remain the time-consuming issue. Another direction is multi-scale feature fusion, in which HyperNet [25] brutally concentrates the features maps from low level and high level. Nevertheless, direct feature fusion by normalization might cause information loss. In this case, FPN [8] builds top-down pathway and lateral connections to help information interaction in different levels that obtains a large improvement in small object detection. Based on feature pyramid structure, PANet [26] and Libra R-CNN [24] attempt to achieve further feature refinement by introducing additional paths. Moreover, with the help of dilated convolution [27], TridentNet [15] proposes to generate scale-specific feature maps rather than fusing them, which achieves state-of-the-art performance.

Methods on feature selective. While most of object detection methods try to solve scale-variance problem through multi-level feature fusion, a few researches focus on feature selection. FSAF [28] presents a feature selective module deployed in RetinaNet [19] but holds an obvious potential risk: hard feature selection strategy, which is similar to FPN. Differently, [14] develops a scale weighting approach used in pedestrian detection based on Faster R-CNN [22]. However, this method trains two parallel subnetworks for detecting large and small instances, respectively, which results in longer inference time. In this context, we design a region-guided scale-aware feature selective module deployed in most of two-stage FPN-style object detection methods to achieve flexible feature selection.

3 Motivation

3.1 Scale variation problem

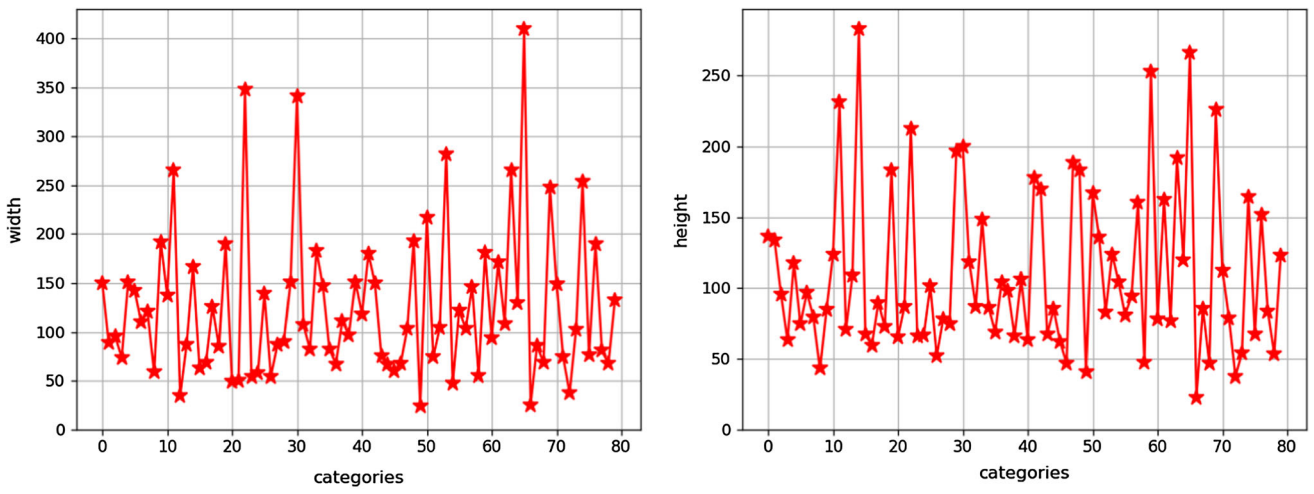
In this paper, we aim to solve the scale variation issue, which is one of the major challenges in object detection task. Specifically, scale variation appears to be an image containing numerous instances with various sizes including widths and heights. This size difference derives from a large number of labelled categories in the dataset. Figure 2 illustrates the size difference phenomenon in one of the popular object detection dataset COCO [29]. As it can be seen, due to the specific attribute of each object class, the widths and heights of labelled instances are unevenly distributed among these categories. This problem inevitably results in network training being more biased toward specific object sizes but ignoring the actual scale distribution of the whole dataset. Currently, due to the effectiveness of feature pyramid, two-stage detectors show the state-of-the-art performance in object detection task. Therefore, in order to discuss the motivation of the idea of this paper, we attempt to analyze the advantages and limitations of current two-stage object detectors.

3.2 Rethinking Region Proposal Network

Firstly, we investigate the performance of the Region Proposal Network in the standard Feature Pyramid Network algorithm. As it is well known, the major goal of RPN is to provide lots of high-quality region proposals that could be adopted as potential object locations. So, here we ignore the categories of these region proposals and treat the region generation as a binary classification task (objectness or non-objectness). Instead, we only focus on the recall of their localization that indicates that how well the region proposals cover the ground truths. Table 1 shows the localization results of region proposals on COCO *minival* subset generated by RPN using Faster R-CNN with ResNet50 backbone. As it can be seen, there is only 35.3% recall obtained under intersection over union (IoU) of 0.5 when maximum detection quantity is set to 10 because the total number of objects per image is high in COCO dataset. However, when more region proposals are selected, RPN could provide much more correct regions that could achieve more than 90% recall. Even with the IoU of 0.7, still 77.7% ground truths could be well found and localized. Therefore, the RPN could achieve satisfied performance on object localization task. The provided region proposals show a high recall rate that could cover almost all the ground truths. Furthermore, excellent recall could be also obtained under higher IoU threshold which indicates that these regions are with high quality.

3.3 Rethinking R-CNN head

Apart from RPN, the second step is designing a R-CNN head to classify and fine-tune the region proposals in two-stage object detectors. Here, we select the top 1000 region proposals used for classification. Note that we only focus on the positive region samples' classification performance rather than all the regions because these region proposals might contain a large number of negative samples that are meaningless for object detection evaluation. The classification accuracy results are illustrated in Fig. 3. As it can be observed, the R-CNN could only correctly recognize 60.1% positive regions. Furthermore, under the increasing classification score threshold, the accuracy dramatically drops to less than 40%. This indicates that the high-quality



(a) Width distribution **(b)** Height distribution

Fig. 2 Sample size distribution of instances on COCO dataset

Table 1 Recall performance of region proposals on COCO *minival* subset

IoU	0.5	0.55	0.6	0.65	0.7	0.75	0.8	0.85	0.9	0.95
Recall_{10}	0.353	0.338	0.319	0.295	0.261	0.215	0.157	0.094	0.035	0.005
Recall_{100}	0.726	0.706	0.682	0.647	0.590	0.499	0.361	0.198	0.069	0.008
Recall_{300}	0.838	0.819	0.796	0.759	0.699	0.594	0.425	0.225	0.074	0.008
Recall_{1000}	0.908	0.892	0.872	0.839	0.777	0.662	0.468	0.241	0.078	0.009

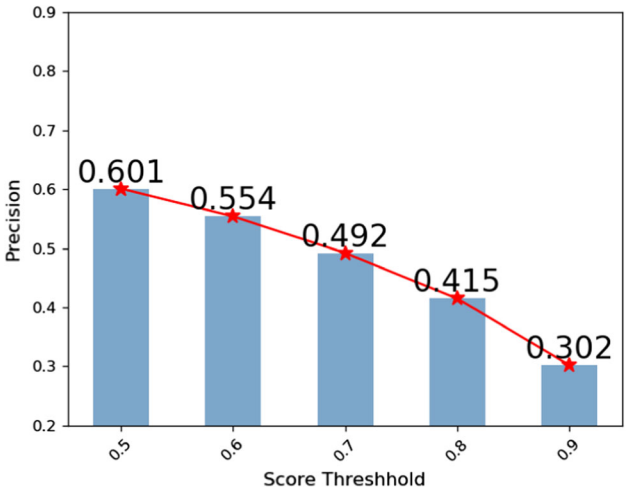


Fig. 3 COCO *minival* set classification accuracy

region proposals might not be well classified by R-CNN head. Therefore, we can conclude that the major limitation of current two-stage object detectors lies on the powerless region classification performance. So, we design our SAFS module deployed in R-CNN head of standard FPN to relieve the scale variation issue.

4 Scale-aware feature selective module

In this section, we are going to introduce the pipeline of proposed SAFS module in our method shown in Fig. 4, including the RoI Pyramid, weighting gate function as well as its optimization. Furthermore, we also introduce the multi-level SAFS structure.

4.1 RoI Pyramid

As discussed before, one reason causing unsatisfied classification results under scale variation is the hard feature selection strategy used in standard feature pyramid architectures. It extracts feature map by RoI Align for each region proposal with size of $w \times h$ from the feature level corresponding to its scale:

$$l(w,h) = \left\lfloor \log_2 \sqrt{wh}/\theta \right\rfloor \tag{1}$$

where θ is the hyper-parameter that is usually set to 32. $\lfloor \cdot \rfloor$ is the floor operation. This strategy follows a well-known expert consensus. Specifically, for the instances with small sizes (smaller than $\theta \times \theta$), it is more inclined to select low-level feature maps as their RoI feature descriptors due to the sparse information of the small instances in high-level semantic feature maps with down-sampling. Obviously, there is a limitation that the cross-level information might be ignored, which is also of great value for object detection. In order to find a way to simultaneously take the features from all the levels into consideration, we design a RoI Pyramid structure, in which the input image is firstly learned by standard FPN and the output is multi-scale feature maps $F = (F_1, F_2, \ldots, F_p)$. Secondly, we adopt RPN module into each feature level, respectively, and collect region proposals from all the scales. Then, several parallel RoI Align operations are applied to extract multi-scale RoI features $X = (X_1, X_2, \ldots, X_p)$ for every region proposal by:

$$X_p(i,j,c) = \frac{k^2}{w_p h_p} \sum_{m=x_1}^{w_p k/i} \sum_{n=y_1}^{h_p k/i} F_p(m,n,c) \tag{2}$$

where k is the output size of RoI Align. w_p and h_p are the sampling sizes of region proposal in p_{th} feature level, which are defined as:

$$\begin{aligned} w_p &= w \times s_p \\ h_p &= h \times s_p \end{aligned} \tag{3}$$

s_p is the down-sampling spatial stride that is represented as (1/4, 1/8, 1/16, 1/32) in RoI Pyramid. In this way, we could extract RoI features for region proposals in different feature levels that contain multi-scale object information. So, this structure is named as RoI Pyramid.

4.2 Weighting gate function

Given the multi-scale RoI features $X = (X_1, X_2, \ldots, X_p)$, we aim to fuse these into a more powerful RoI feature. Different from harshly summing or concatenating, a specific weighting gate function is proposed to achieve

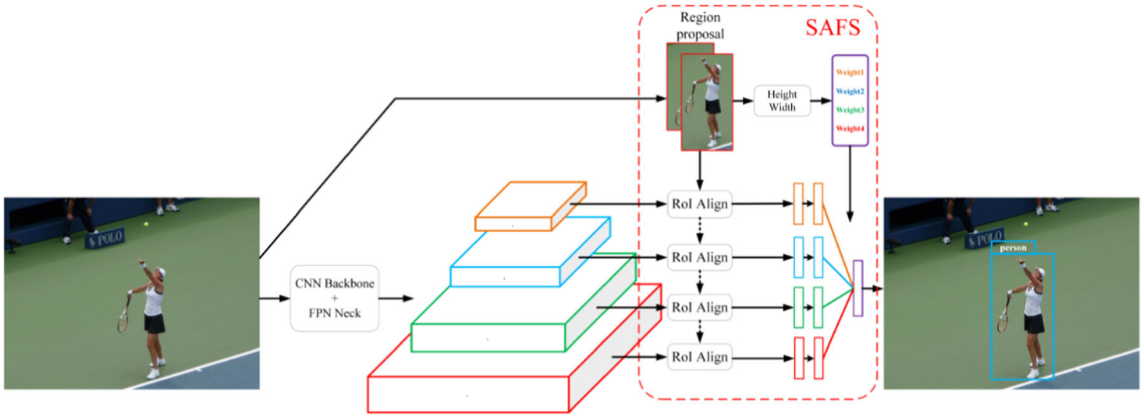


Fig. 4 Illustration of our proposed SAFS architecture. For each instance (region proposal), we build RoI Pyramid to extract multi-scale RoI features operating by RoI Align recurrently. The weights for different feature levels are computed with the input of instance size (height and width). Final feature used for classification is the result of weighted feature fusion

scale-aware feature fusion so as to make it adaptive to the region proposal's size. This function could automatically learn the weights of RoI features from different scales rather than introducing more complex neural networks. In our design, the weighting gate function is expected to follow three constraints: (1) Given a instance with small size, the weight should be more biased toward the low-level RoI feature scales and vice versa. (2) Weighting gate function is optimized by one trainable parameter to ensure the flexible nonlinear mapping between input instance size and output weights. (3) It is better to introduce fewer parameters to avoid much computational cost.

In this context, we define our weighting gate function to calculate the weight of p_{th} RoI feature λ_p by:

$$\lambda_p(s_p, h, w) = \frac{e^{-z_p(s_p,h,w)}}{\sum_i e^{-z_p(s_p,h,w)}} \tag{4}$$

in which z_p is a temporary variable that is computed by:

$$z_p(s_p, h, w) = \frac{\text{sign}^*(h, w)\sqrt{|h - h_0||w - w_0|}}{s_p \beta_p} \tag{5}$$

where $\beta = (\beta_1, \beta_2, \ldots, \beta_p)$ is a trainable parameter. w and h are the width and height of the instance. In addition, $sign^*(w, h)$ is a novel designed sign function that indicates whether the input instance is small object or not, which is defined as:

$$sign^*(h, w) = \begin{cases} 1 & w - w_0 < 0 \text{ and } h - h_0 < 0 \\ -1 & \text{elsewise} \end{cases} \tag{6}$$

where w_0 and h_0 are two pre-defined hyper-parameters. As it could be found, the instance is processed as small object only when $w - w_0 < 0$ and $h - h_0 < 0$. Therefore, when the network is fed a small instance, the $z_p > 0$ and z_p shows a growth trend as the level of RoI Pyramid keeps increasing, i.e., s_p reduces. In this case, the e^{-z_p} in Eq. (4) is a decreasing function, so the weight λ_p for low-level RoI feature is larger than others. On the contrary, the weight for high level is becoming larger when the input instance is determined to be a large object by $\text{sign}^*(w, h)$. Furthermore, due to the fact that the absolute value of z_p is computed by $\sqrt{|h - h_0||w - w_0|}$ and β_p, the weight distribution for each RoI feature scale shows a nonlinear mapping, in which the weights are automatically learned during training to achieve scale-aware soft feature selection mechanism.

Finally, we could fuse the features from RoI Pyramid and obtain the powerful feature representation $\tilde{X}$ of region proposals:

$$\tilde{X} = \sum_i X_i \cdot e^{\lambda_i} \tag{7}$$

Here, we employ the exponential weighting operation to maintain the original RoI features as well as expand those soft-selected levels. By combining features extracted from RoI Pyramid with learned weights, our SAFS module could be more robust to scale variation, which could be considered as a soft selection of different feature levels. Besides, the fused feature map could effectively select information that is the most suitable for R-CNN, which alleviates the problem of weak classification performance in two-stage object detectors.

4.3 Multi-level feature selective structure

In our weighting gate function, we design a sign function $\text{sign}^*(w, h)$ as to determine the input instance is small or large object, in which the criterion relies on two hyper-parameters w_0 and h_0 that are set manually. However, it is difficult to generalize the single w_0 and h_0 into various datasets. So, we design the multi-level feature selective architecture apart from the proposed single level structure,

as shown in Fig. 5. In the single level method, we only adopt one set of hyper-parameters as the criterion for determining small instances, where the w_0 and h_0 are chosen as the average width and height among all the instances in training set:

$$w_0 = \bar{w} = \frac{1}{N}\sum_{b=1}^{N} w_b$$
$$h_0 = \bar{h} = \frac{1}{N}\sum_{b=1}^{N} h_b \tag{8}$$

Differently, we aim to find various feature fusion strategies in our multi-level feature selective architecture. Here, we select three different hyper-parameter combinations for w_0 and h_0. Specifically, we pre-analyze the scale variations in the whole dataset and apply an unsupervised clustering learning [30] to find three kernels from all the labelled instances, as shown in Fig. 6. It is obvious that most of labelled instances are with small sizes, but the aspect ratios of bounding boxes are widely distributed. In this way, we could select three different types of kernels as the hyper-parameter combinations, which also ensure the model could take various shapes of objects into account.

4.4 Optimization

In the weighting gate function, we introduce a trainable parameter β. Here, we will discuss their optimization during model training. Firstly, we apply neural network chain derivation strategy [32] to compute the gradient for β. Because that β_p participates the weight computation of all the feature levels, the gradient of β_p is also from two situations. When $p = i$, the gradient of λ_i to β_p is:

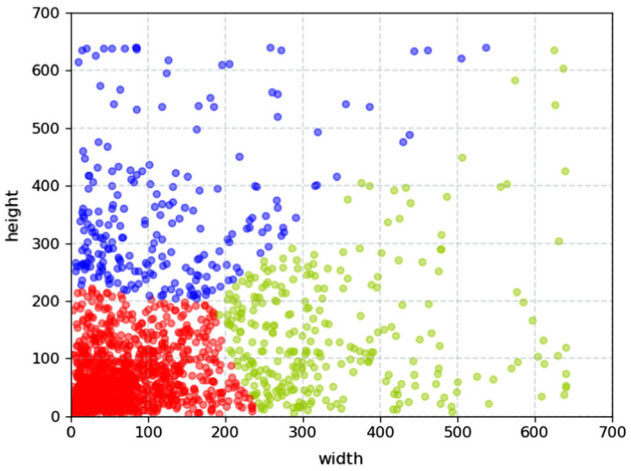


Fig. 6 Height and width cluster result of instances on COCO dataset. Note that part of labelled instances' sizes is shown here for the best view

$$\frac{\partial \lambda_i}{\partial \beta_p} = \frac{z_p e^{-z_p}\sum_i e^{-z_i} - z_p e^{-z_i} e^{-z_p}}{\beta_p (\sum_i e^{-z_i})^2} = \frac{z_p e^{-z_p}}{\beta_p \sum_i e^{-z_i}}\left(1 - \frac{e^{-z_i}}{\sum_i e^{-z_i}}\right) \tag{9}$$

The other situation is when $p \neq i$, the gradient is computed by:

$$\frac{\partial \lambda_i}{\partial \beta_p} = -\frac{z_p e^{-z_i} e^{-z_p}}{\beta_p (\sum_i e^{-z_i})^2} \tag{10}$$

Therefore, we could obtain the final gradient of training loss to β_p by summing them up:

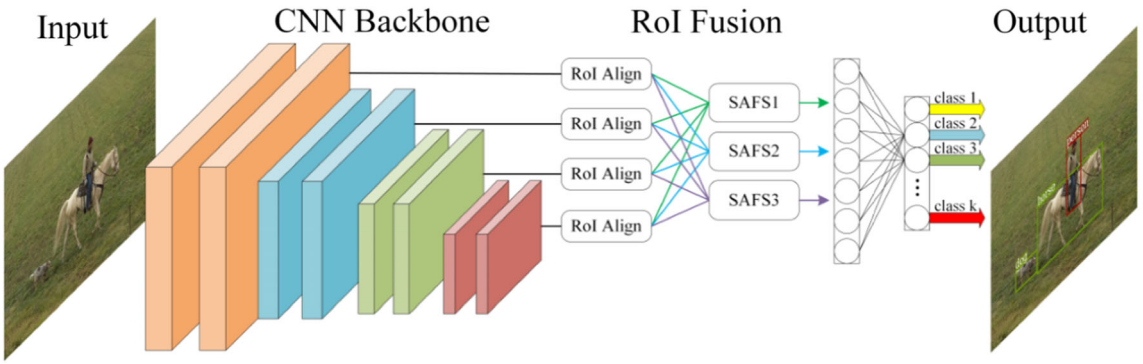


Fig. 5 Illustration of multi-level scale-aware feature selective module. We design three parallel SAFS modules with different hyper-parameters to fuse RoI features from various scales in RoI Pyramid. Final detections from multiple SAFS branches will be combined by non-maximum suppression (NMS) [31]. The RPN and R-CNN heads are shared with other two-stage object detectors and ignored for simplicity in this figure

Table 2 Ablative experiments for the SAFS on the COCO *minival* set

Method	RoI feature	Single level	Multi-level	AP	AP_{50}	AP_{75}	AP_s	AP_m	AP_l
Baseline	Hard selection			36.4	58.4	38.9	21.0	38.9	45.3
SAFS	Sum	✓		36.8	58.7	39.1	21.1	39.2	45.4
	Concat	✓		36.7	58.4	38.7	20.8	39.8	45.0
	Soft selection	✓		36.9	58.5	39.6	21.6	40.3	47.6
	Soft selection		✓	**37.2**	**59.1**	**39.7**	**22.2**	**40.4**	**47.9**

Best results of our own methods comparing to others are highlighted in bold

The baseline is Faster R-CNN with FPN method, and SAFS is also deployed in this approach. ResNet-50 [35] is the backbone network for all experiments in this table

Table 3 State-of-the-art comparison on MS COCO *test-dev* set [29]

Method	Backbone	AP	AP_{50}	AP_{75}	AP_s	AP_m	AP_l
YOLOv2 [17]	DarkNet-19	21.6	44.0	19.2	5.0	22.4	35.5
YOLOv3 [6]	DarkNet-53	33.0	57.9	34.4	18.3	35.4	41.9
SSD512 [7]	ResNet-101	31.2	50.4	33.3	10.2	34.5	49.8
DSSD512 [18]	ResNet-101	33.2	53.3	35.2	13.0	35.4	51.1
RetinaNet [19]	ResNet-101-FPN	39.1	59.1	42.3	21.8	42.7	50.2
FSAF [28]	ResNet-101-FPN	40.9	61.5	44.0	24.0	44.2	51.3
TridentNet [15]	ResNet-101-FPN	42.7	63.6	46.5	23.9	46.6	56.6
Faster R-CNN [22]	ResNet-101	34.9	55.7	37.4	15.6	38.7	50.9
Faster R-CNN [22]	ResNet-50-FPN	36.7	59.0	39.3	21.8	39.8	45.2
Faster R-CNN [22]	ResNet-50-PAFPN [26]	36.9	58.6	39.4	21.6	40.2	47.0
Faster R-CNN [22]	ResNet-101-FPN	38.8	60.9	42.1	22.6	42.4	48.5
Faster R-CNN+SAFS (Ours)	ResNet-50-FPN	37.3	59.4	40.1	21.8	40.4	46.1
Faster R-CNN+SAFS (Ours)	ResNet-50-PAFPN [26]	38.1	60.2	40.6	22.4	42.2	49.0
Faster R-CNN+SAFS (Ours)	ResNet-101-FPN	39.3	61.4	42.6	23.0	42.8	49.3
Libra R-CNN [24]	ResNet-50-FPN	38.7	59.9	42.0	22.7	41.2	47.6
Libra R-CNN [24]	ResNet-101-FPN	40.3	61.3	43.9	22.9	43.1	51.0
Libra R-CNN+SAFS (Ours)	ResNet-50-FPN	39.4	60.6	42.9	23.4	41.9	48.4
Libra R-CNN+SAFS (Ours)	ResNet-101-FPN	41.2	62.2	45.1	23.7	44.2	51.5
Cascade R-CNN [23]	ResNet-50-FPN	40.7	59.3	44.1	23.1	43.6	51.4
Cascade R-CNN [23]	ResNet-101-FPN	42.4	61.1	46.1	23.6	45.4	54.1
Cascade R-CNN [23]	ResNeXt-101-64x4d-FPN	44.8	63.8	48.7	26.2	48.0	56.8
Cascade R-CNN [23]	ResNeXt-101-64x4d-DCN-FPN	47.7	66.9	51.5	27.8	50.9	61.2
Cascade R-CNN+SAFS (Ours)	ResNet-50-FPN	41.3	59.8	44.9	23.4	43.7	52.7
Cascade R-CNN+SAFS (Ours)	ResNet-101-FPN	43.0	61.6	46.7	24.1	45.9	55.0
Cascade R-CNN+SAFS (Ours)	ResNeXt-101-64x4d-FPN	45.3	64.2	49.2	26.5	48.3	57.8
Cascade R-CNN+SAFS (Ours)	ResNeXt-101-64x4d-DCN-FPN	**48.3**	**67.5**	**52.4**	**28.6**	**51.5**	**62.3**

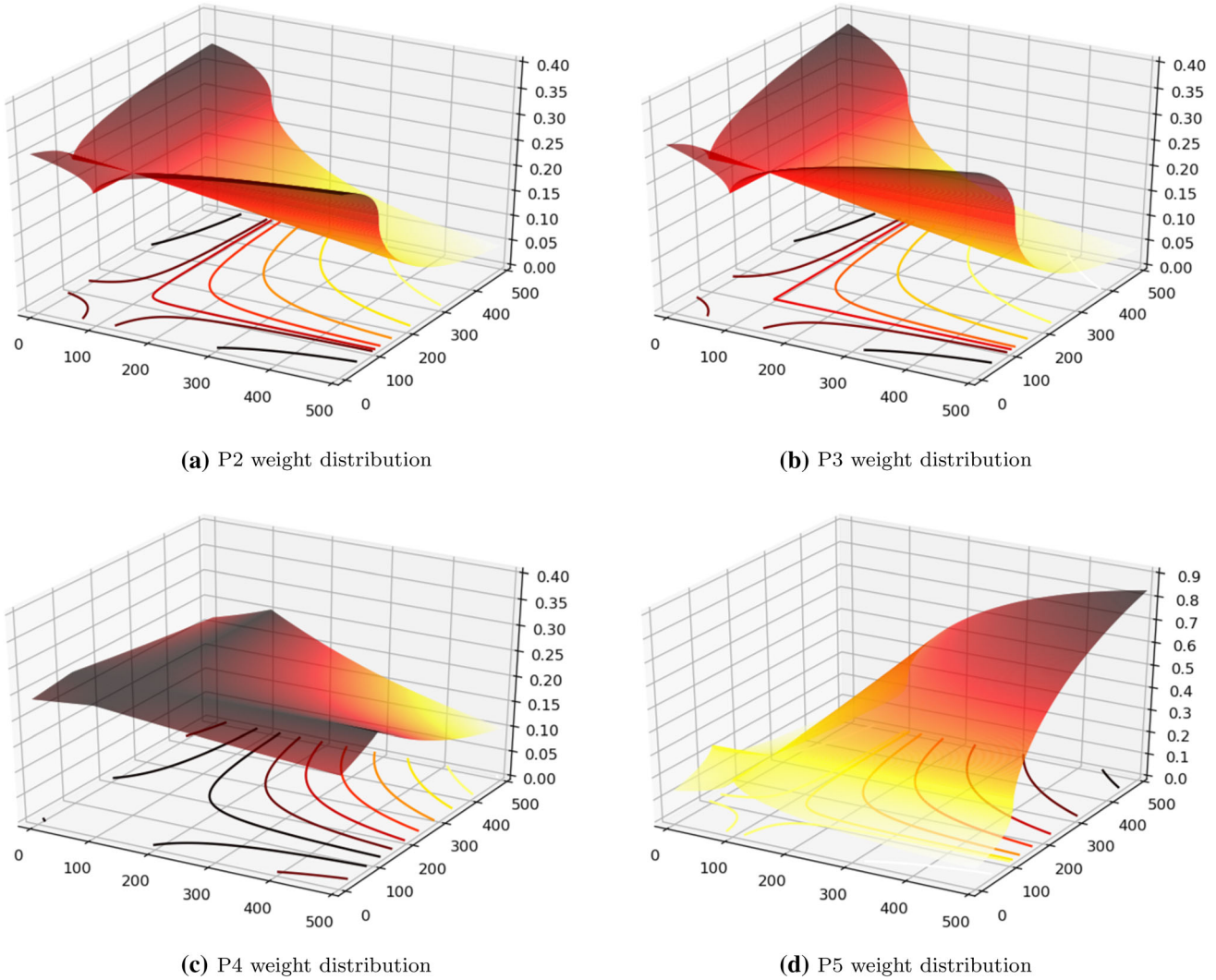
Best results of our own methods comparing to others are highlighted in bold

ResNet [35] and ResNeXt [36] are two popular CNN backbones in current object detection methods. DCN represents the deformable convolutional network [37], and PAFPN represents the path aggregation feature pyramid network [26]

$$\frac{\partial L}{\partial \beta_p} = \sum_i \left(\frac{\partial L}{\partial \lambda_i} \frac{\partial \lambda_i}{\partial \beta_p} \right) = \sum_{p=i} \left(\frac{\partial L}{\partial \lambda_i} \frac{\partial \lambda_i}{\partial \beta_p} \right) + \sum_{p \neq i} \left(\frac{\partial L}{\partial \lambda_i} \frac{\partial \lambda_i}{\partial \beta_p} \right) \tag{11}$$

Finally, given the gradient of β_p, the automatic training of the parameter could be achieved by back-propagation with learning rate η by:

$$\beta_p = \beta_p - \eta \frac{\partial L}{\partial \beta_p} \tag{12}$$



(a) P2 weight distribution

(b) P3 weight distribution

(c) P4 weight distribution

(d) P5 weight distribution

Fig. 7 Weight distribution in P2 to P5 RoI feature levels in our method

5 Experiments

In this section, we conduct experiments on the COCO dataset [29] as well as PASCAL VOC [38] and UVM [39] dataset. Following the general training strategy for COCO, we train our models with the union of 80k training images and 35k subset of validation images (*trainval*). In this section, we describe the implementation details and evaluation metrics. Then, we conduct ablation experiments on COCO *minival* set. Finally, we illustrate the comparison of our proposed method with state-of-the-art object detectors on a set of 20k *test-dev* set.

5.1 Implementation details

Due to the hardware differences, we reimplement our method as well as state-of-the-art methods in our experimental environment. Specifically, both baselines and SAFS methods are trained in a batch size of 8 on 4 NVIDIA 1080Ti GPUs with 12 GB memory. Following standard object detectors' training strategy, the network backbones are pre-trained on the ImageNet dataset [33] and then fine-tuned on the detection dataset. For fair comparison, the input images are resized to have a short side of 800 pixels and long side is no more than 1333. All the models are trained in an end-to-end manner. Following the learning schedule in detectron [34], we train 12 epochs ($\sim$ 90k iterations) in total with learning rate starting from 0.01 and decreased by a factor of 0.1 at the eighth and tenth training epoch. As for the trainable parameter β in SAFS module, we use constant initialization 0.1.

For the evaluation, we experiment our method on MS COCO benchmark as well as PASCAL VOC [38] and UVM [39] datasets. We report the standard COCO evaluation metrics of Average Precision (AP) as well as AP_{50} and AP_{75}. For validating the scale variation performance,

we report COCO-style AP_s, AP_m and AP_l on objects of small (less than 32×32), medium (from 32×32 to 96×96) and large (greater than 96×96) sizes.

5.2 Ablation studies

The key idea of SAFS module is to develop a soft feature fusion strategy to weight the RoI features from different levels. Apart from soft selection mechanism, there are several RoI feature fusion methods such as summing and concatenation. In order to discuss which fusing strategy is better in current two-stage object detection approaches, we compare the effectiveness and difference of them, in which these different strategies are performed in the same baseline. Table 2 reports the AP of various IoU threshold as well as AP for small, medium and large objects separately. As it could be observed, the results for medium and large objects present a large margin (2–4%) between our SAFS and baseline method. In addition, even with RoI Pyramid to extract multi-scale features, concatenation strategy to fuse RoI features seems not to be a satisfied way. This might be explained that concatenation is the operation with more tricks leading to large difficulty in optimization during training. Comparing with other strategies, our SAFS using soft selection in multi-level feature selective structure could obtain the best results in *minival* subset. Therefore, we perform the rest experiments with this strategy.

5.3 Comparison with state of the arts

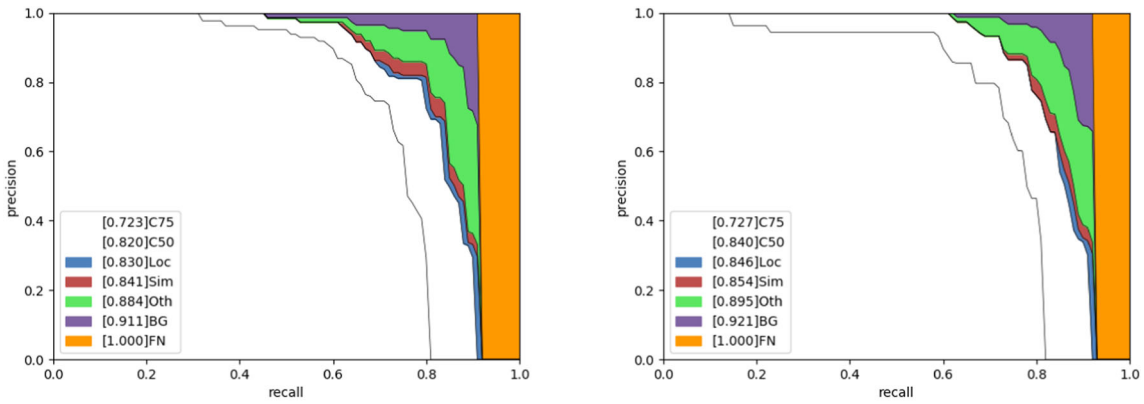
We evaluate our complete SAFS method on the COCO *test-dev* set to compare with current state-of-the-art object detection approaches. For a fair comparison, we report the results of single model with single-scale testing for all methods. The detection results are shown in Table 3. As it can be seen, our proposed SAFS pushes the envelope of accuracy boundary to a new level. Under the cooperation of multi-scale RoI Pyramid and scale-aware feature, two-stage object detectors could obtain approximately 0.6% to 0.7% AP improvement. Comparing with AP under various IoU threshold, our SAFS could achieve 0.8% AP_{75} higher than corresponding approaches that are not deployed with SAFS, which indicates that our method plays a significant role in more precise object localization. In addition, SAFS module could also alleviate scale variation problem in object detection task, which brings different amplitudes of AP improvement. In particular, there is a huge margin in AP_l with 2% on the comparison. This phenomenon explains that multi-scale RoI features with region-guided weighting and fusion are much more important than single feature consideration. Finally, our method performs 48.3% AP on COCO *test-dev* set, which achieves the state-of-the-art object detection performance.

5.4 Analysis and discussion

In order to further study the effect of our SAFS module, we visualize the weight distribution on a different RoI Pyramid with various sizes of input instances, which is shown in Fig. 7. Note that P2 to P5 represent the different down-sampling spatial stride in RoI Pyramid. Obviously, under the guidance of our learned SAFS module, the RoI feature at each level in RoI Pyramid could be effectively learned and recalibrated according to the size of input instance. In detail, the learned weight distribution follows the popular feature selection strategy of most researchers, in which the network tends to capture more information from low-level RoI features (P2 and P3 levels) when the input instance is regarded as small object ($w < w_0$ and $h < h_0$). On the contrary, high-level RoI features (P5 level) might be endowed with a large weight for classifying the instances with large sizes. However, different from current expert consensus of hard feature selection strategy, our SAFS module proves that only 40% low-level information is enough for small instance classification task in current two-stage object detectors. Besides, high-level RoI features are also significant for these small objects in improving detection performance with 30% feature selected from P4 level. And for large objects, the network not only samples RoI features on P5 level, but also automatically captures around 10% from P2 and P3 feature levels. This indicates that a small amount of low-level information could dramatically help the classification accuracy of large-size objects.

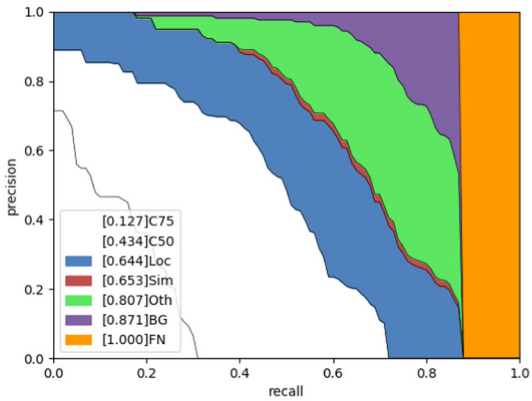
For detailed quantitative evaluation of our proposed SAFS, we conduct an error analysis on COCO *minival* set. Figure 8 shows the comparison of baseline method (the left column) and our SAFS approach (the right column) in three categories frisbee, skis and sheep that indicate small, medium and large objects, respectively. The overall AP performance of our method could achieve 72.7% and 49.9% at IoU = 0.75 for frisbee and sheep categories. Specifically, comparing with baseline, our SAFS module boosts the AP to 84.6%, 65.8% and 80.2% under perfect localization. For small and medium sizes of objects (frisbee and skis), the AP boundary could be pushed more than 1 point higher than baseline, which shows that when eliminating background false positives, the BG for these two categories increases from 91.1 to 92.1% and 87.1 to 88.1%. This is in line with the goal of SAFS module to improve the instance classification accuracy. Therefore, our approach shows a superior performance over baseline.

In terms of qualitative evaluation, we visualize some of detection results in Fig. 9. It turns out that SAFS module

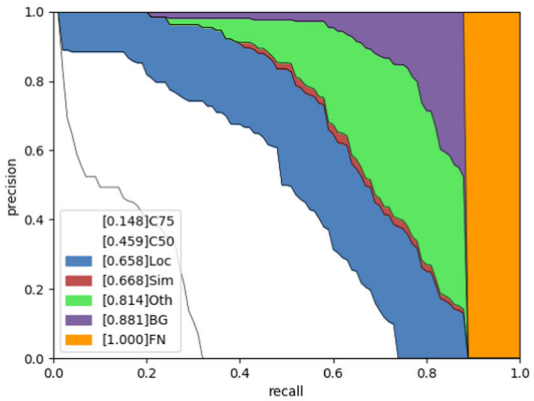


(a) Baseline (label=frisbee)

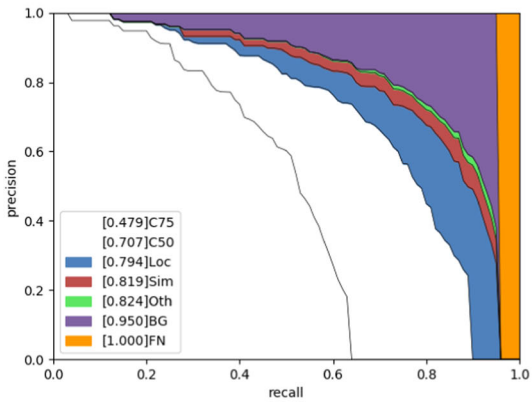
(b) SAFS (label=frisbee)



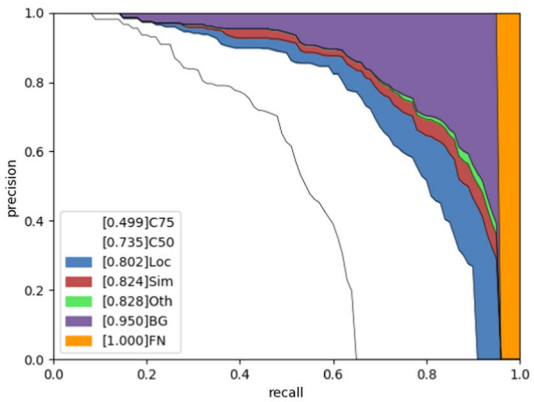
(c) Baseline (label=skis)



(d) SAFS (label=skis)



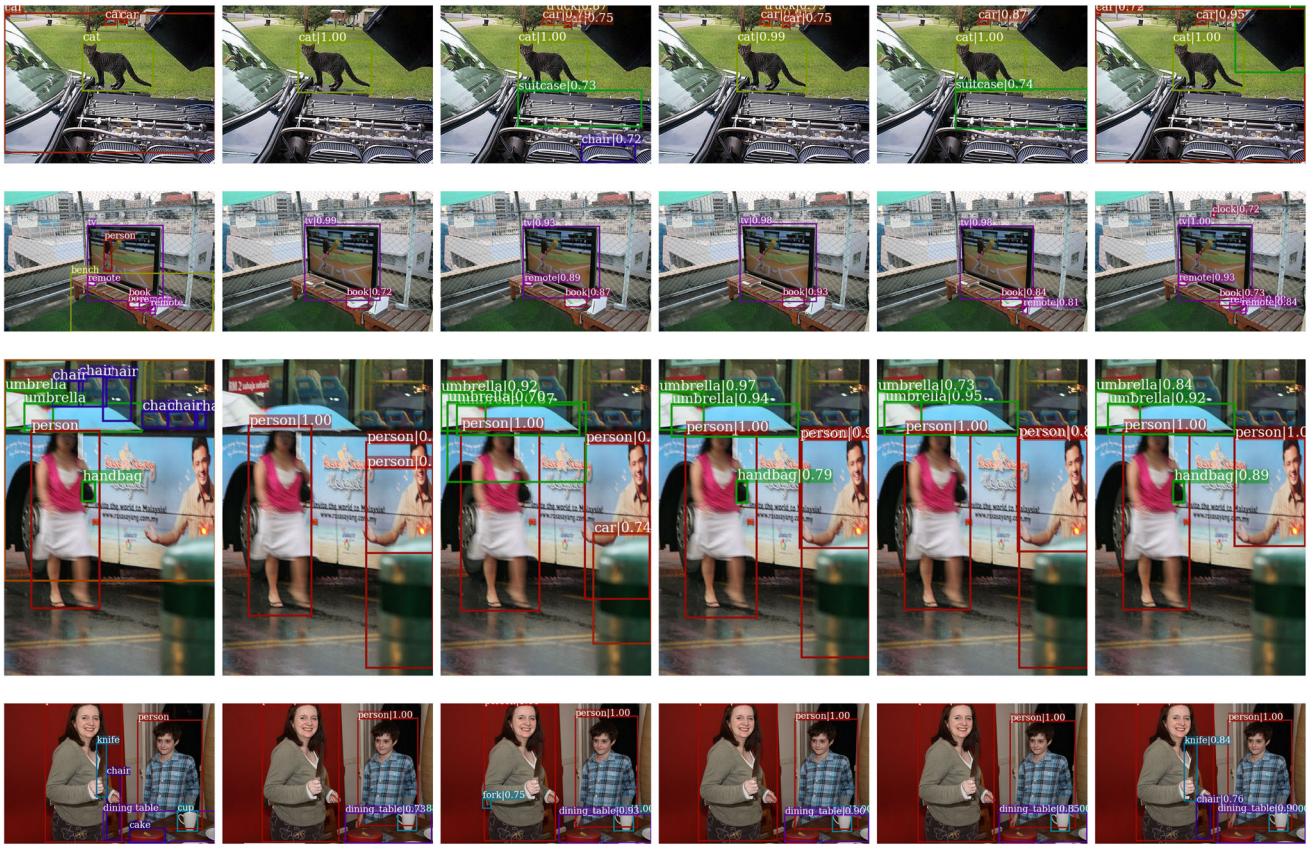
(e) Baseline (label=sheep)



(f) SAFS (label=sheep)

Fig. 8 Quantitative evaluation of detection performance on COCO *minival* set. From top to bottom row: performance comparison on frisbee, skis and sheep categories. The baseline is performed by Faster R-CNN with ResNet-50, and our SAFS is also deployed in this approach

could help current two-stage detectors in improving performance on detecting objects with various scales. In our method, under the RoI feature fusion, the misdetected or misclassified instances could be found and corrected precisely. In addition, the confidence scores of instances are also largely improved, which indicates that SAFS module is more robust to object detection. Therefore, it could be concluded to prove the effectiveness of our proposed SAFS module.



(a) Ground Truth (b) SSD512 (c) Faster R-CNN (d) Libra R-CNN (e) Cascade R-CNN (f) Ours

Fig. 9 Qualitative detection performance on COCO *minival* set. From left to right column: performance comparison on ground truth, Faster R-CNN [22], Libra R-CNN [24], Cascade R-CNN [23] and our SAFS method. Note that SAFS is deployed on Cascade R-CNN detector and the CNN backbone is VGG16 [2] for SSD512 [7] and ResNet-50 [35] for others. Each color belongs to an object category

5.5 Generalization capacity

Detection performance on other benchmark: We also validate the detection performance of our SAFS on the other popular benchmarks, e.g., PASCAL VOC and Retail UVM. The results are shown in Table 4. As it could be seen, on PASCAL VOC benchmark, the baseline detector helps improve a slight point when employed with our SAFS module. In addition, on another object detection dataset UVM [39], we could also find that SAFS module contributes to an obvious improvement on total AP metric, especially AP under IoU 0.75. This could conclude that our method shows to be a good way to detect objects in a finer level.

Inference timing: The real-time performance of our method SAFS is demonstrated in Fig. 10. The baseline is Faster R-CNN [22], Libra R-CNN [24] and Cascade R-CNN [23] with backbone ResNet-50 to ResNet-101 [35],

Table 4 Detection performance comparison on Retail UVM datasets [39]

Benchmark	Method	Backbone	SAFS	AP	AP_{50}	AP_{75}	AP_s	AP_m	AP_l
PASCAL VOC [38]	Faster R-CNN* [22]	ResNet-50-FPN		–	78.9	–	–	–	–
	Faster R-CNN [22]	ResNet-50-FPN	✓	–	**79.6**	–	–	–	–
UVM [39]	Faster R-CNN* [22]	ResNet-50-FPN		74.4	96.8	86.4	–	31.8	74.8
	Faster R-CNN [22]	ResNet-50-FPN	✓	**79.9**	**99.0**	**95.4**	–	**34.0**	**80.0**

* is reimplemented in our experiment environment for fair comparison. This might be some slight differences (around 0.1 to 0.3)

Here, we adopt Faster R-CNN [22] with ResNet-50 [35] as baseline. Our method SAFS is also employed in this approach. *Indicates our reimplementation

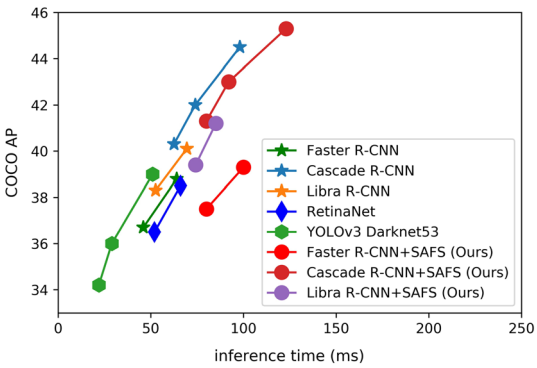
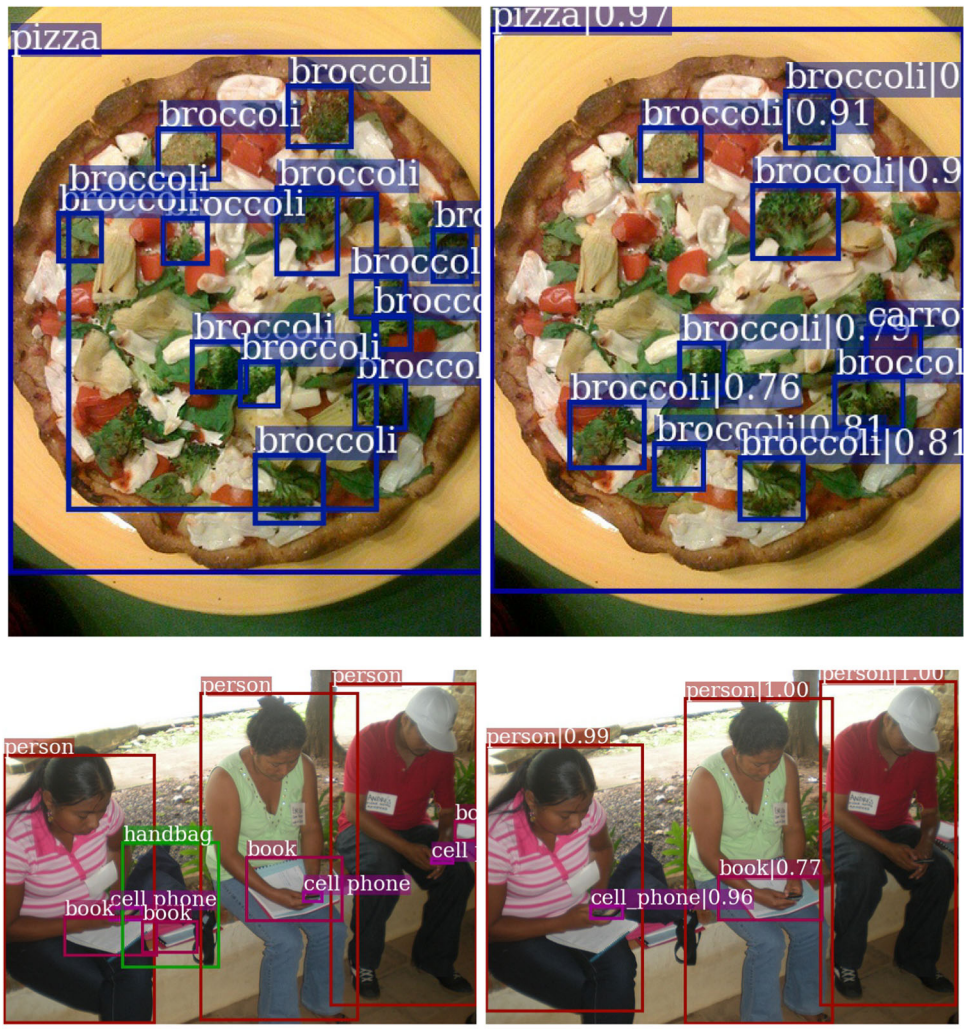


Fig. 10 Inference time comparison on our method (SAFS) with some other popular object detectors. Expect for Yolo v3, each node in this line chart indicates the different CNN backbone adopted, from shallow to deep

and our SAFS is also deployed in this setting. It could be concluded in Fig. 10 that our method could achieves a good trade-off on detection performance and inference time. This is because the computational overhead of SAFS module is usually small with a few training parameters compared to various popular object detectors, at testing phase.

Limitation and future work: Even though our SAFS has shown an acceptable generalization capacity on generic object detection task, there exist several limitations when adopting it into more practical and general circumstance. We visualize some detection failures in Fig. 11. It could be summarized that our method might not achieve a satisfied detection performance on the objects that are densely distributed with small sizes. This could be explained by the limited contribution from coarse feature levels for tiny objects' semantics. In addition, our method attempts to improve the detection performance in local level, while the qualities of region proposals for dense tiny objects in



(a) Ground Truth **(b)** Detection Results by our method

Fig. 11 Failure cases on COCO *minival* set. From left to right: ground truth and detection results performed by Faster R-CNN with ResNet-50 involving our SAFS

global level might be decreased. Therefore, in the future work, we aim to design a specific detector to deal with this situation.

6 Conclusion

In this paper, we present a simple but effective module for improving the performance and alleviating scale variation problem of current two-stage object detectors named scale-aware feature selective (SAFS) module. In our method, we propose a RoI Pyramid structure to extract multi-scale RoI features. Instead of simply harshly summing or concatenating them, we develop a novel weighting gate function to automatically learn the weights under the input instance size, which could achieve scale-aware training and inference scheme. Experimental results show that our SAFS module could be very compatible with most of two-stage object detection architectures and bring a significant improvement comparing with state of the arts. We believe that SAFS could benefit current computer vision community and beyond.

Acknowledgements This work was supported in part by the National Natural Science Foundation of China (NSFC) under Grant Nos. 61773360 and 31671586 and in part by the Major Special Science and Technology Project of Anhui Province under Grant No. 201903a06020006.

Compliance with ethical standards

Conflict of interest We declare that we do not have any commercial or associative interest that represents a conflict of interest in connection with the work submitted.

References

1. Krizhevsky A, Sutskever I, Hinton GE (2012) Imagenet classification with deep convolutional neural networks. In: Advances in neural information processing systems, pp 1097–1105
2. Simonyan K, Zisserman A (2014) Very deep convolutional networks for large-scale image recognition. arXiv preprint arXiv:14091556
3. Girshick R (2015) Fast R-CNN. In: Proceedings of the IEEE international conference on computer vision, pp 1440–1448
4. Long J, Shelhamer E, Darrell T (2015) Fully convolutional networks for semantic segmentation. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp 3431–3440
5. Girshick R, Donahue J, Darrell T, 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, pp 580–587
6. Redmon J, Farhadi A (2018) YOLOv3: an incremental improvement. arXiv preprint arXiv:180402767
7. Liu W, Anguelov D, Erhan D, Szegedy C, Reed S, Fu CY, Berg AC (2016) SSD: single shot multibox detector. In: European conference on computer vision. Springer, pp 21–37
8. Lin TY, Dollár P, Girshick R, He K, Hariharan B, Belongie S (2017) Feature pyramid networks for object detection. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp 2117–2125
9. Yang F, Choi W, Lin Y (2016) Exploit all the layers: fast and accurate CNN object detector with scale dependent pooling and cascaded rejection classifiers. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp 2129–2137
10. Dai J, Li Y, He K, Sun J (2016) R-FCN: object detection via region-based fully convolutional networks. In: Advances in neural information processing systems, pp 379–387
11. He K, Gkioxari G, Dollár P, Girshick R (2017) Mask R-CNN. In: Proceedings of the IEEE international conference on computer vision, pp 2961–2969
12. Hu P, Ramanan D (2017) Finding tiny faces. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp 951–959
13. Singh B, Davis LS (2018) An analysis of scale invariance in object detection snip. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp 3578–3587
14. Li J, Liang X, Shen S, Xu T, Feng J, Yan S (2017) Scale-aware fast R-CNN for pedestrian detection. IEEE Trans Multimedia 20(4):985–996
15. Li Y, Chen Y, Wang N, Zhang Z (2019) Scale-aware trident networks for object detection. In: Proceedings of the IEEE international conference on computer vision, pp 6054–6063
16. Redmon J, Divvala S, Girshick R, Farhadi A (2016) You only look once: unified, real-time object detection. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp 779–788
17. Redmon J, Farhadi A (2017) YOLO9000: better, faster, stronger. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp 7263–7271
18. Fu CY, Liu W, Ranga A, Tyagi A, Berg AC (2017) DSSD: deconvolutional single shot detector. arXiv preprint arXiv:170106659
19. Lin TY, Goyal P, Girshick R, He K, Dollár P (2017) Focal loss for dense object detection. In: Proceedings of the IEEE international conference on computer vision, pp 2980–2988
20. Zhang S, Wen L, Bian X, Lei Z, Li SZ (2018) Single-shot refinement neural network for object detection. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp 4203–4212
21. Uijlings JR, Van De Sande KE, Gevers T, Smeulders AW (2013) Selective search for object recognition. Int J Comput Vis 104(2):154–171
22. Ren S, He K, Girshick R, Sun J (2015) Faster R-CNN: towards real-time object detection with region proposal networks. In: Advances in neural information processing systems, pp 91–99
23. Cai Z, Vasconcelos N (2018) Cascade R-CNN: delving into high quality object detection. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp 6154–6162
24. Pang J, Chen K, Shi J, Feng H, Ouyang W, Lin D (2019) Libra R-CNN: towards balanced learning for object detection. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp 821–830
25. Kong T, Yao A, Chen Y, Sun F (2016) Hypernet: towards accurate region proposal generation and joint object detection. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp 845–853
26. Liu S, Qi L, Qin H, Shi J, Jia J (2018) Path aggregation network for instance segmentation. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp 8759–8768
27. Yu F, Koltun V (2015) Multi-scale context aggregation by dilated convolutions. arXiv preprint arXiv:151107122

28. Zhu C, He Y, Savvides M (2019) Feature selective anchor-free module for single-shot object detection. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp 840–849
29. Lin TY, Maire M, Belongie S, Hays J, Perona P, Ramanan D, Dollár P, Zitnick CL (2014) Microsoft coco: common objects in context. In: European conference on computer vision. Springer, pp 740–755
30. Krishna K, Murty MN (1999) Genetic k-means algorithm. IEEE Trans Syst Man Cybern Part B (Cybern) 29(3):433–439
31. Neubeck A, Van Gool L (2006) Efficient non-maximum suppression. In: 18th international conference on pattern recognition (ICPR'06), vol 3. IEEE, pp 850–855
32. Hecht-Nielsen R (1992) Theory of the backpropagation neural network. In: Neural networks for perception. Elsevier, pp 65–93
33. Deng J, Dong W, Socher R, Li LJ, Li K, Fei-Fei L (2009) Imagenet: a large-scale hierarchical image database. In: 2009 IEEE conference on computer vision and pattern recognition. IEEE, pp 248–255
34. Girshick R, Radosavovic I, Gkioxari G, Dollár P, He K (2018) Detectron. https://github.com/facebookresearch/detectron
35. He K, Zhang X, Ren S, Sun J (2016) Deep residual learning for image recognition. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp 770–778
36. Xie S, Girshick R, Dollár P, Tu Z, He K (2017) Aggregated residual transformations for deep neural networks. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp 1492–1500
37. Zhu X, Hu H, Lin S, Dai J (2019) Deformable convnets v2: more deformable, better results. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp 9308–9316
38. Everingham M, Van Gool L, Williams CK, Winn J, Zisserman A (2010) The pascal visual object classes (VOC) challenge. Int J Comput Vis 88(2):303–338
39. Zhang H, Li D, Ji Y, Zhou H, Wu W, Liu K (2019) Towards new retail: a benchmark dataset for smart unmanned vending machines. IEEE Trans Ind Inform 16:7722–7731