HCE: Hierarchical Context Embedding for Region-Based Object Detection

Zhao-Min Chen, Xin Jin, Bo-Rui Zhao, Xiaoqin Zhang, and Yanwen Guo, *Member, IEEE*

***Abstract*— State-of-the-art two-stage object detectors apply a classifier to a sparse set of object proposals, relying on region-wise features extracted by RoIPool or RoIAlign as inputs. The region-wise features, in spite of aligning well with the proposal locations, may still lack the crucial context information which is necessary for filtering out noisy background detections, as well as recognizing objects possessing no distinctive appearances. To address this issue, we present a simple but effective Hierarchical Context Embedding (HCE) framework, which can be applied as a plug-and-play component, to facilitate the classification ability of a series of region-based detectors by mining contextual cues. Specifically, to advance the recognition of context-dependent object categories, we propose an image-level categorical embedding module which leverages the holistic image-level context to learn object-level concepts. Then, novel RoI features are generated by exploiting hierarchically embedded context information beneath both whole images and interested regions, which are also complementary to conventional RoI features. Moreover, to make full use of our hierarchical contextual RoI features, we propose the early-and-late fusion strategies (*i.e.*, feature fusion and confidence fusion), which can be combined to boost the classification accuracy of region-based detectors. Comprehensive experiments demonstrate that our HCE framework is flexible and generalizable, leading to significant and consistent improvements upon various region-based detectors, including FPN, Cascade R-CNN, Mask R-CNN and PA-FPN. With simple modification, our HCE framework can be conveniently adapted to fit the structure of one-stage detectors, and achieve improved performance for SSD, RetinaNet and EfficientDet.**

***Index Terms*— Object detection, context embedding, region-based CNNs.**

I. Introduction

THE region-based object detectors [1]–[6] popularized by R-CNN framework [1] are conceptually intuitive and flexible, and have achieved top accuracies on challenging benchmarks like MS-COCO [7]. Region-based detectors first generate a sparse set of object proposals, and then refine the proposal locations and classify them as one of the foreground classes or as background using a detection head. One crucial module in such a proposal-driven pipeline is the RoIPool [2] or RoIAlign [8] operator, which is responsible for extracting RoI (Region of Interests) features aligned with the proposal locations for the detection head.

In this paper, we revisit the RoI features in region-based detectors from the perspective of context information embedding. Our key motivation relies on the fact that while each RoI in very deep CNNs may have a very large theoretical receptive field that often spans the entire input image [2]. However, the effective receptive field [9] may only occupy a fraction of the full theoretical receptive field, making the RoI features insufficient for characterizing objects that are highly dependent on context information, such as "bowl", "skateboard" etc. Here, the contextual information means any auxiliary information that can assist in suppressing the false positive detections in noisy backgrounds, or recognizing objects that have no distinctive appearances themselves. For example, as shown in Fig. 1 (a), the semantic features of "traffic light" are strong evidences for filtering out the activations of irrelevant object categories like "potted plant". On the other hand, as shown by Fig. 1 (b), the cloth and even the human pose are useful clues for correctly classifying a proposal as "tie".

Recently, several works exploited the region-level context information to improve the localization ability of two-stage detectors. Chen *et al.* [10] demonstrated that rich contextual information from neighboring regions can better refine the proposal locations for two-stage detectors. Kantorov *et al.* [11] leveraged the surrounding context regions to improve weakly supervised object localization. However, to the best of our knowledge, currently there is no enabling framework which is systematically designed for embedding context information to improve the *classification ability* of region-based detectors.

In this paper, we present a novel Hierarchical Context Embedding (HCE) framework for region-based object detectors. Our framework consists of three modules. Firstly, we consider that the simplest way to break the contextual limit in object detection, is to partially cast the object-level feature learning as an image-level multi-label classification task. Building upon this realization, we design an image-level categorical embedding module, which in essence is a multi-label classifier upon the detection backbone, in parallel with the existing region-based detection head. It enables the backbone

Manuscript received December 15, 2020; revised May 15, 2021 and July 7, 2021; accepted July 21, 2021. Date of publication August 2, 2021; date of current version August 6, 2021. This work was supported in part by the National Natural Science Foundation of China under Grant 61922064 and Grant U2033210, in part by the Zhejiang Provincial Natural Science Foundation under Grant LR17F030001, and in part by the Project of Science and Technology Plans of Wenzhou City under Grant C20170008. The associate editor coordinating the review of this manuscript and approving it for publication was Dr. Xiaolin Hu. *(Corresponding author: Xiaoqin Zhang.)*

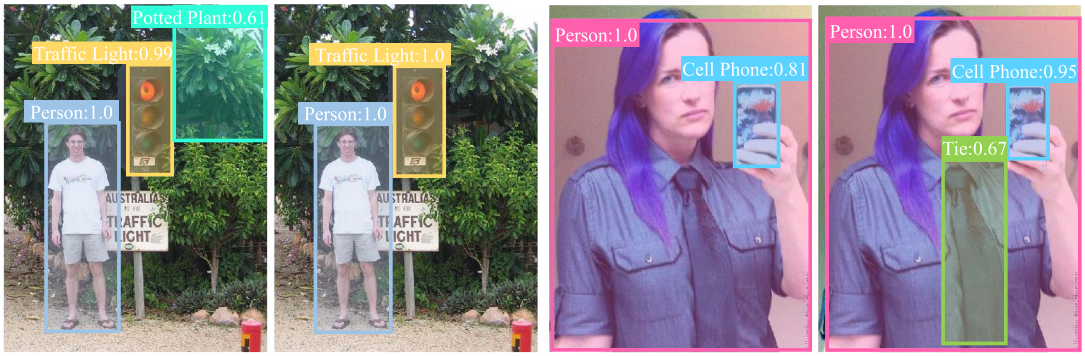
Zhao-Min Chen and Xiaoqin Zhang are with the College of Computer Science and Artificial Intelligence, Wenzhou University, Wenzhou 325035, China (e-mail: chenzhaomin123@gmail.com; zhangxiaoqinnan@gmail.com).

Xin Jin is with Samsung Research Nanjing, Samsung Research, Nanjing 210012, China (e-mail: jinxin.ashin@outlook.com).

Bo-Rui Zhao is with Megvii Research Nanjing, Megvii Technology, Nanjing 210046, China (e-mail: zhaoborui@megvii.com).

Yanwen Guo is with the National Key Laboratory for Novel Software Technology, Nanjing University, Nanjing 210023, China (e-mail: ywguo@nju.edu.cn).

Digital Object Identifier 10.1109/TIP.2021.3099733



(a) Filtering out noisy detections.

(b) Recognizing indistinctive objects.

Fig. 1. Motivation and example results of our Hierarchical Context Embedding (HCE) framework. By incorporating discriminative context information, our framework can effectively filter out the noisy false positive background detections, and correctly classify objects which possess no distinctive appearances.

to exploit the whole image context to learn discriminative features for context-dependent object categories. Even as a standalone enhancement, our image-level categorical embedding module can lead to improvements over existing region-based detectors.

Upon the image-level categorical embedding module, at the instance-level, we design a simple but effective process to generate hierarchical contextual RoI features which can be directly utilized by the region-wise detection head. Because our contextual RoI features are enhanced by image-level categorical supervisions and exploit larger contexts, they are by nature complementary to conventional RoI features, which is trained by region-based detectors and only exploits limited context. Later, the early-and-late modules, *i.e.*, feature fusion and confidence fusion, are designed to make full use of our contextual RoI features. By quantitative experiments, we demonstrate that they can be combined to further boost the classification accuracy of the detection head.

In general, our proposed HCE framework is easy to implement and is end-to-end trainable. We conduct extensive experiments on MS-COCO 2017 [7] to validate the effectiveness of our HCE framework. Without bells and whistles, our HCE framework delivers consistent accuracy improvements for almost all existing mainstream region-based detectors and one-stage detectors, including FPN [4], Mask R-CNN [8], Cascade R-CNN [5], PA-FPN [12], NAS-FPN [13], SSD [14] and EfficientDet [15]. We also conduct ablation studies to verify the effectiveness of each module involved in our HCE framework. Fig. 1 gives the example images of the baseline method and our method, which demonstrates that our framework can effectively filter out the noisy background detections and correctly classify indistinctive objects by leveraging the context information it exploited.

This paper is an extension version of previous conference paper [16]. Specifically, we make three major extensions. Firstly, we embrace the flexibility of our HCE framework by adapting it for one-stage detectors, and achieve improved performance for SSD [14], RetinaNet [17] and EfficientDet [15]. Second, we conduct more comparisons and ablation studies about our HCE framework, *i.e*, comparing HCE with the non-local module [18] and confirming that HCE is complementary to the non-local module for context information embedding, and the effects of the accuracy of the multi-label classifier on the detection performance, and setting different weights to our losses. Thirdly, we provide more visualization of our methods, demonstrating that the activation maps produced by our image-level categorical embedding module have more proper responses on objects of interests, and thus can provide useful contextual clues for the detection head. We also show some failure cases of our HCE framework, and give analysis about the root cause. These extensions further demonstrate the generality and flexibility on our HCE framework.

II. Related Work

A. CNN-Based Object Detection

Convolutional neural networks have lead to a paradigm shift of object detection in the past decades [19]. Among a large number of approaches, the two-stage R-CNN series [1]–[5] have become the leading detection framework. The pioneer work, *i.e.*, R-CNN [1], extracts region proposals from image with selective search [20], and applies a convolutional network to classify each region of interests independently. Fast R-CNN [2] improves R-CNN by sharing convolutional features among RoIs, which enables fast training and inference. Then, Faster R-CNN [3] advances the region proposal generation with a Region Proposal Network (RPN). RPN shares the feature extraction backbone with the detection head, which in essence is a Fast R-CNN [3]. Faster R-CNN is a famous two-stage detection framework, and is the foundation for many follow-up works [4], [21].

Over very recent years, several algorithms have been proposed to further improve the two-stage Faster R-CNN framework. For example, Feature Pyramid Networks (FPN) [4] constructed inherent CNN feature pyramids, which can largely improve the detection performance of small objects. Mask R-CNN [8] extended Faster RCNN by constructing the mask branch, and boosted the performance of both object detection and instance segmentation. Cascade R-CNN [5] utilized multi-stage training strategy to progressively improve the quality of region proposals, and demonstrated significant gains for high quality (measured by higher IoUs) object detection. Complementary to these works, in this paper, we focus on developing a Hierarchical Context Embedding (HCE) framework to improve the *classification ability* of all region-based detectors. Thanks to the simplicity and generalization ability of our HCE framework, it brings consistent and significant improvements over aforementioned leading region-based detectors, *e.g.*, FPN, Mask R-CNN and Cascade R-CNN.

Besides the R-CNN series, the one-stage dense object detectors originated by YOLO [22] and SSD [14] are growing more and ore popular. One-stage detectors perform regular pixel-by-pixel dense prediction on feature maps, eliminating RoI-based feature aggregation and classification operations of R-CNN series. In recent years, one-stage detectors have achieved competitive results with two-stage detectors, by introducing the feature pyramid networks (FPN) to reduce the ambiguity in label assignment for objects for different sizes, and focal loss to make the training focus on a sparse set of hard examples and prevent the vast number of easy negatives from overwhelming the training process. In this paper, we demonstrate that our HCE framework, with minor modification, can also benefit one-stage detectors like SSD [14] and RetinaNet [17].

B. Context Information for Object Detection

In object detection, both global context [23] and local context [24] are widely exploited for improving performance, especially when object appearances are insufficient due to small object size, occlusion, or poor image quality. Our work is inspired by some of previous works, but the key motivation or implementation significantly differ with these works. Next, we review several topics in object detection, which are closely related to our work.

1) Combined Localization and Classification: Before the era of deep learning, Harzallah *et al.* [25] proposed to combine two closely related tasks, *i.e.*, object localization and image classification. They demonstrated that classification can improve detection by a contextual combination and vice versa. Similar in spirit, we utilize the fully image-level context to learn object-level concepts. But differently, we utilize global context to learn CNN features rather than hand-crafted features adopted in [25]. Furthermore, we integrate hierarchical contextual clues beneath both whole images and interested regions to modern region-based CNN detectors, rather than the traditional sliding window detector used by [25].

2) Region Proposal Refinement: Recently, Chen *et al.* [10] explored the rich contextual information to refine the region proposals for object detection. The neighboring regions with useful contexts can benefit the localization quality of region proposals, which further lead to better detection performance. Instead of refining proposals, we focus on improving the *classification ability* of region-based detectors by embedding hierarchical contextual clues.

3) Weakly-Supervised Object Detection: In weakly supervised object detection, the bounding box annotations are not provided, and only image-level categorical labels are available. The common practice [11], [26]–[28] in this area is to first generate a set of noisy object proposals, and then learn from these noisy proposals with specially designed robust algorithms. Among them, Kantorov *et al.* [11] proposed a context-aware deep network which leverages the surrounding context regions to improve localization. Unlike the usage of region-level context information [11] for weakly supervised detection, we focus on the task of fully-supervised object detection, and particularly exploit global image-level context to advance the recognition of context-dependent object categories.

C. Context Information for Other Vision Tasks

Beyond object detection, context information has also been utilized to improve other vision tasks [29]–[34]. For example, Wang *et al.* [29] leveraged attention mechanisms and LSTMs to discover semantic-aware regions and capture the long-range contextual dependencies for multi-label image recognition. He *et al.* [30] proposed an adaptive context module to generate multi-scale context representations for semantic segmentation. Qu *et al.* [35] embedded multi-context information (the appearance of the input image and semantic understanding) to obtain the shadow matte. Byeon *et al.* [31] leveraged the LSTM units to capture the entire available past context on video prediction. Li *et al.* [32] adopted the dilated convolution to acquire more contextual information for single image deraining.

III. Approach

A. Framework Overview

We begin by briefly describing our Hierarchical Context Embedding (HCE) framework (see Fig. 2) for region-based object detection. Firstly, an image-level categorical embedding module is employed to advance the feature learning of the objects that are highly dependent on larger context clues. Then, hierarchical contextual RoI features are generated by fusing both instance-level and global-level information derived from the image-level categorical embedding module. Finally, early-and-late fusion modules are designed to make full use of the contextual RoI features to improve the classification performance. Our HCE framework is flexible and generalizable, as it can be applied as a plug-and-play component for almost all mainstream region-based object detectors.

B. Image-Level Categorical Embedding

As aforementioned, conventional RoI-based training for region-based detectors may lack the context information, which is crucial for learning discriminative filters for context-dependent objects. To break this limitation, in parallel with the RoI-based branch, we exploit image-level categorical embedding upon the detection backbone, enabling the backbone to learn object-level concepts adaptively from *global-level* context. Our image-level categorical embedding module does not require additional annotations, as the image-level labels can be conveniently obtained by collecting all instance-level categories in an image.

Essentially, our image-level categorical embedding module is based on a multi-label classifier. As shown in Fig. 2 and Fig. 3 (a), we first apply a 3×3 convolution layer on the output of ResNet conv_5 to obtain the input feature map, and then employ both global max-pooling (GMP) and global average-pooling (GAP) for feature aggregation (as in [36]). Here,

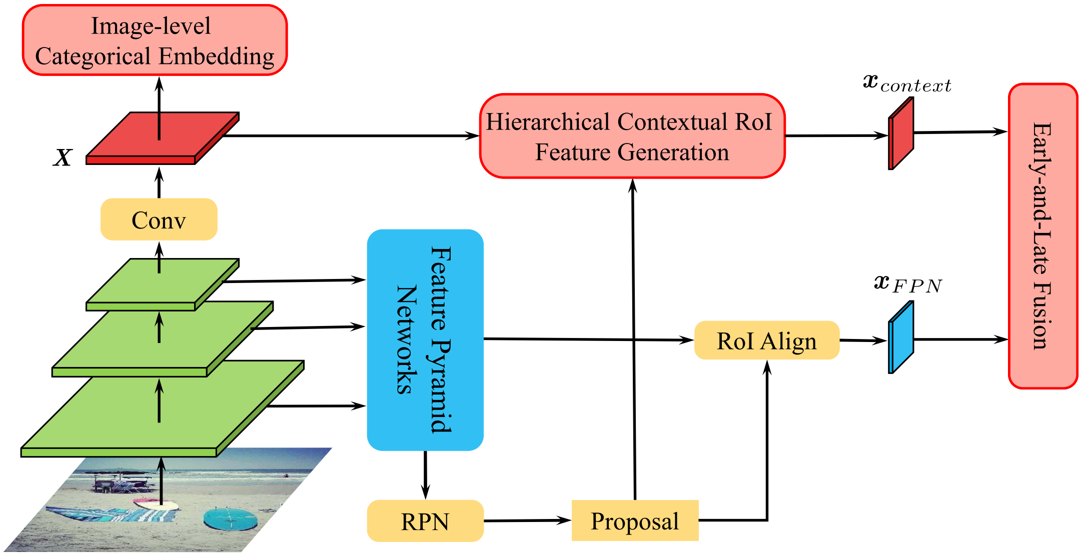
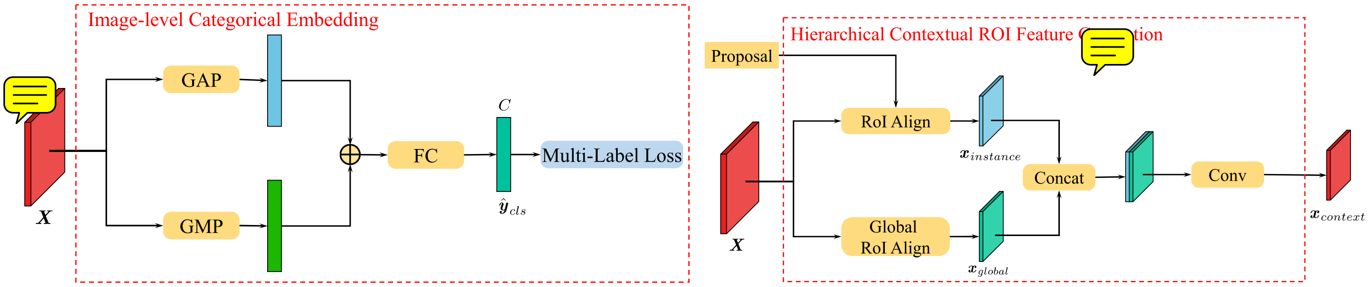


Fig. 2. Overview of our Hierarchical Context Embedding (HCE) framework. At the image-level, we design an *image-level categorical embedding* module upon the detection backbone, which enables the network to learn object-level concepts from global-level context. At the instance-level, we generate *hierarchical contextual RoI features* that are complementary to conventional RoI features, and design the early-and-late fusion modules (*i.e.*, *feature fusion* and *confidence fusion*) to make full use of the contextual RoI features for improving the classification accuracy of the detection head.



(a) Image-level categorical embedding.

(b) Hierarchical contextual RoI feature.

Fig. 3. The design of our image-level categorical embedding module and hierarchical contextual RoI feature generation module.

the additional 3 × 3 convolution layer aims to alleviate the possible slide effects over the original detection backbone.

We refer to the input feature map to our image-level embedding module as *context-embedded image feature*. This is because the input feature map conveys whole image context for learning all object categories that appear in the image, and in turn, each location of the feature map is supervised by all object categories. By contrast, conventional RoI-based trained by region-based detectors only exploits limited context for learning each object category.

Formally, let $\boldsymbol{X} \in \mathbb{R}^{d\times h\times w}$ denote the input feature map, where d is the channel dimensionality, h and w are the height and width, respectively. Then, the multi-label classifier is constructed by C binary classifiers for all categories:

$$\hat{\boldsymbol{y}}_{cls} = f_{cls}((f_{gmp}(\boldsymbol{X}) + f_{gap}(\boldsymbol{X}))) \in \mathbb{R}^{C}, \quad (1)$$

where C denotes the number of categories, each element of $\hat{\boldsymbol{y}}_{cls}$ is a confidence score (logits), and f_{cls} is binary classifier modeled as one fully-connected layer. We assume that the ground truth label of an image is $\boldsymbol{y} \in \mathbb{R}^{C}$, where $y^{i} = \{0, 1\}$ denotes whether object of category i appears in the image or not. The multi-label loss can be formulated as follows

$$\mathcal{L}_{mll} = -\sum_{c=1}^{C} y^{c} \log(\sigma(\hat{y}^{c}_{cls})) + (1 - y^{c}) \log(1 - \sigma(\hat{y}^{c}_{cls})), \quad (2)$$

where $\sigma(\cdot)$ is the sigmoid function.

Because the global feature learning strategy is complementary to RoI-based training, our image-level categorical embedding module standalone can boost the performance of existing region-based detectors (demonstrated later by experiments, cf. Table II). However, one limitation of image-level categorical embedding might be that the derived context-embedded image feature cannot be directly used by the detection head.

C. Hierarchical Contextual RoI Feature Generation

To further benefit region-wise classification, we generate hierarchical contextual RoI features by combining the instance-level and global-level information from the context-embedded image features. The hierarchical contextual RoI feature generation process is shown in Fig. 3 (b).

1) Context-Embedded Instance-Level Feature: We apply RoIAlign [8] with proposals generated by RPN on the context-embedded feature map $\boldsymbol{X}$ to obtain RoI features $\boldsymbol{x}_{instance}$:

$$\boldsymbol{x}_{instance} = f_{RoIAlign}(\boldsymbol{X}; h', w') \in \mathbb{R}^{d\times 7\times 7}, \qquad (3)$$

where $f_{RoIAlign}(\cdot)$ is the RoIAlign operation and h' and w' are the height and width of the RoI, respectively. As $\boldsymbol{x}_{instance}$ is extracted from the context-embedded image feature $\boldsymbol{X}$, we term it as *context-embedded instance-level feature*.

2) Context-Aggregated Global-Level Feature: To leverage larger context, we exploit RoIAlign on the context-embedded image feature $\boldsymbol{X}$ to aggregate the global-level context. We refer to the derived RoI feature as *context-aggregated global-level feature* $\boldsymbol{x}_{global}$:

$$\boldsymbol{x}_{global} = f_{RoIAlign}(\boldsymbol{X}; H, W) \in \mathbb{R}^{d\times 7\times 7}, \qquad (4)$$

where H and W are the height and width of the input image, respectively.

Once context-embedded instance-level feature $\boldsymbol{x}_{instance}$ and context-aggregated global-level feature $\boldsymbol{x}_{global}$ obtained, we concatenate these two RoI features and apply a 1×1 convolution layer to obtain our hierarchical contextual RoI feature $\boldsymbol{x}_{context}$:

$$\boldsymbol{x}_{context} = f_{conv}([\boldsymbol{x}_{instance} : \boldsymbol{x}_{global}]) \in \mathbb{R}^{d\times 7\times 7}, \qquad (5)$$

where $f_{conv}(\cdot)$ denotes the 1×1 convolution operation, [:] refers to concatenation and the ReLU nonlinearity operations are performed following the convolution layer. As the resulting hierarchical contextual RoI feature $\boldsymbol{x}_{context}$ absorbs rich context information from the context-embedded image feature $\boldsymbol{X}$, it is by nature complementary to the conventional RoI feature extracted from the feature pyramid network (FPN) [4].

D. Early-and-Late Fusion and Inference

To make full use of our contextual RoI feature $\boldsymbol{x}_{context}$, we design the early-and-late fusion modules, *i.e.*, feature fusion and confidence fusion, which has been proven effective in many applications [37], [38]. We show that early-and-late fusion is also well suited to improve region-wise detectors, as it can fully absorb hierarchically embedded information from different levels.

1) Feature Fusion: To incorporate our contextual RoI features $\boldsymbol{x}_{context}$ into region-based detection pipeline, the simplest way is fusing them with the original RoI features extracted from the feature pyramid network (FPN) [4] with element addition. Formally, let $\boldsymbol{x}_{FPN}$ denote the original RoI feature extracted from FPN, and $\boldsymbol{x}_{fusion}$ denote the fused RoI feature, then we have:

$$\boldsymbol{x}_{fusion} = \boldsymbol{x}_{context} + \boldsymbol{x}_{FPN} \in \mathbb{R}^{d\times 7\times 7}. \qquad (6)$$

As shown in Fig. 4, the fused feature map $\boldsymbol{x}_{fusion}$ is then fed into the $2fc$ detection head to produce refined bounding boxes and classification scores.

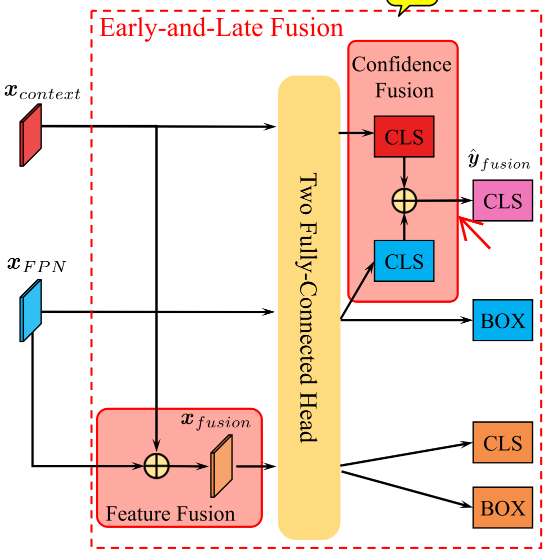


Fig. 4. The design of our early-and-late fusion modules.

2) Confidence Fusion: We also consider a simple confidence fusion strategy which is complementary to feature fusion. We apply the $2fc$ head on our hierarchical contextual RoI feature $\boldsymbol{x}_{context}$ to produce a classification confidence (logits), and then fuse it with that from the corresponding FPN RoI feature $\boldsymbol{x}_{FPN}$ by addition. Formally, let $\hat{\boldsymbol{y}}_{fusion}$ denote the fused the confidence:

$$\hat{\boldsymbol{y}}_{fusion} = f_{2fc}(\boldsymbol{x}_{context}) + f_{2fc}(\boldsymbol{x}_{FPN}) \in \mathbb{R}^{C}. \qquad (7)$$

The fused confidence is transformed by a soft-max layer to produce a novel classification score.

For each proposal, the classification score $\hat{\boldsymbol{y}}_{fusion}$, paired with the refined bounding box predicted the FPN RoI feature, forms another prediction in parallel with the prediction from the feature fusion branch. It is worth mentioning that the weights of the $2fc$ head applied on different RoI features are shared.

3) Inferences: Our early-and-late fusion modules produce two different predictions for a single object proposal. To obtain the final result, as shown by the pipeline in Fig. 4, we firstly collect all the boxes and confidences from two prediction branches (*i.e.*, feature fusion and confidence fusion), and then perform NMS over all these boxes. Furthermore, as demonstrated later in experiments, while our two fusion strategies are complementary during training, using only one prediction branch during inference will not cause obvious performance drop but reduce computational cost. However, the performance by only using one fusion strategy for training is inferior to that by using two fusion strategies together.

4) Loss Function: The whole network is trained end-to-end, and the overall loss is computed as follows:

$$\mathcal{L} = \alpha \times \mathcal{L}_{feat} + \beta \times \mathcal{L}_{conf} + \lambda \times \mathcal{L}_{mll} + \mathcal{L}_{rpn}, \qquad (8)$$

where $\mathcal{L}_{feat}$ and $\mathcal{L}_{conf}$ are the losses for the feature fusion and confidence fusion branches, respectively. α, β and λ are the hyper-parameters that are all set to 1.0. All loss terms are considered equally important, which reveals that HCE is generalized and not tricky.

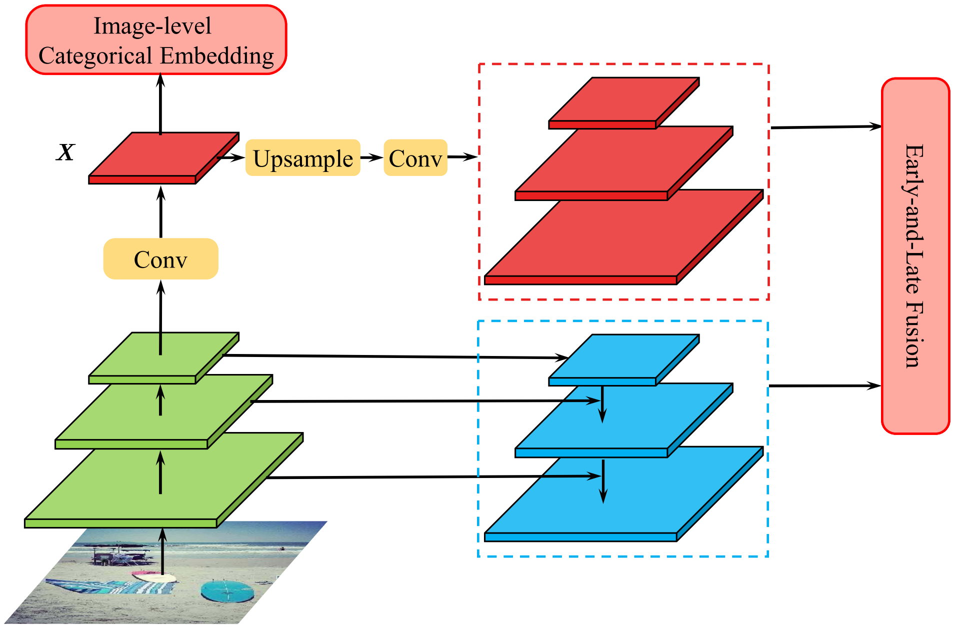


Fig. 5. Overview of our HCE framework modified for one-stage detectors. We keep our image-level categorical embedding module unchanged and up-sample (append with 1 × 1 convolutional layer) the context-embedded image feature $\boldsymbol{X}$ to generate a novel feature pyramid, where the feature map has the shape (spatial size and channel number) with the original FPN at each level. We also adopt the early-and-late fusion modules to obtain the final detection results.

E. Extension to One-Stage Detectors

Although our HCE framework is originally designed for two-stage (multi-stage) region-based detectors, it can be conveniently adapted to fit the structure of one-stage detectors. Since one-stage detectors perform regular pixel-by-pixel dense prediction on feature maps, the RoI-based feature aggregation operations in our HCE framework are no longer needed, allowing us to largely simplify the hierarchical contextual RoI feature generation module.

Fig. 5 demonstrates our design choices for extending HCE to one-stage detectors. Taking the RetinaNet [17] detector as example, we keep the image-level categorical embedding module unchanged, which produces context-embedded image feature $\boldsymbol{X}$. Then, we up-sample and append a series of 1 × 1 convolutional layers on $\boldsymbol{X}$ to generate a novel feature pyramid, where the feature maps have the same shape (spatial resolution and channel number) with the original FPN at each level. By simple element-wise addition, we obtain feature maps and classification confidences that absorb useful contextual cues from the context-embedded image features $\boldsymbol{X}$ for dense prediction.

We note that this modification process can also apply to classic one-stage detectors (*e.g.*, SSD [14]) without constructing FPN, by generating novel feature maps from $\boldsymbol{X}$ that have the same shape with the original feature maps used for prediction in similar manner.

IV. Experiments

A. Implementation Details

We implement our method and re-implement all baseline methods based on MMDetection codebase [39]. The re-implementations of the baselines strictly follow the default settings of MMDetection. Images are resized such that the short edge has 800 pixels while the long edge has less than 1333 pixels. We use no data augmentation except horizontal flipping for training. The ResNet is exploited as backbone, which is pre-trained on ImageNet [40]. Models are trained in a batch size of 16 on 8 GPUs. We train all models with SGD optimizer for 12 epochs in the total, with the initial learning rate as 0.02 and decreased by a factor of 0.1 at 8th epoch and 11th epoch. Weight decay and momentum are set as 0.0001 and 0.9, respectively. We also adopt the linear warming up strategy to begin the training of our model.

B. Comparisons With Baselines

To demonstrate the generality of our HCE framework, we consider three well-known region-based object detectors as our baseline systems, including Feature Pyramid Network (FPN) [4], Mask R-CNN [8] and Cascade R-CNN [5]. All detectors are instantiated with two different backbones, *i.e.*, ResNet-50 and ResNet-101 with FPN. Integrating our framework with Mask R-CNN and Cascade R-CNN is as straightforward as with FPN. For example, we apply our framework within each training stage of Cascade R-CNN.

Table I shows the comparison results on MS-COCO 2017 `val`. Our HCE framework achieves consistent accuracy gains overall all baseline detectors on all evaluation metrics. Specifically, without the bells and whistles, our method improves 2.1% and 1.7% AP for FPN with ResNet-50 and ResNet-101 backbones, respectively. While for more advanced Mask R-CNN and Cascade R-CNN, our method also brings more than 1% AP improvement on both ResNet-50 and ResNet-101 backbones, *e.g.*, improving the AP for Mask R-CNN with ResNet-50-FPN from 37.3% to 38.8%.

We note that our improvements for Mask R-CNN and Cascade R-CNN baselines are not as significant as FPN.

TABLE I

Compared With Baselines (FPN [4], Mask R-CNN [8], Cascade R-CNN [5]), SSD [14], RetinaNet [17], PA-FPN [12], NAS-FPN [13], and EfficientDet [15] on MS-COCO 2017 Val. "HCE" Denotes That the Models Are Trained and Inferred on Both Feature Fusion and Confidence Fusion. Clearly, Our HCE Framework Achieves Consistent Accuracy Gains Overall All Baseline Detectors on All Evaluation Metrics

Backbone	Method	HCE	AP	AP^{50}	AP^{75}	AP^{S}	AP^{M}	AP^{L}
ResNet-50-FPN	FPN		36.3	58.3	39.1	21.6	40.2	46.9
		✓	**38.4**	**61.0**	**41.8**	**22.9**	**42.5**	**49.1**
	Mask R-CNN		37.3	59.1	40.3	22.2	41.1	48.3
		✓	**38.8**	**61.3**	**42.1**	**23.2**	**42.8**	**49.7**
	Cascade R-CNN		40.5	58.7	44.1	22.3	43.6	53.8
		✓	**41.7**	**60.5**	**45.0**	**23.4**	**44.9**	**55.2**
ResNet-101-FPN	FPN		38.3	60.1	41.7	22.8	42.8	49.8
		✓	**40.0**	**62.3**	**43.4**	**24.0**	**44.1**	**51.9**
	Mask R-CNN		39.4	60.9	43.0	23.3	43.7	51.5
		✓	**40.5**	**62.6**	**44.0**	**24.4**	**44.5**	**53.4**
	Cascade R-CNN		41.9	60.1	45.7	23.2	45.9	56.2
		✓	**43.0**	**61.6**	**46.9**	**24.6**	**46.6**	**57.4**
VGG16	SSD300		25.4	43.5	26.2	6.9	27.6	42.2
		✓	**28.0**	**47.7**	**28.8**	**9.9**	**30.0**	**44.6**
ResNet-50-FPN	RetinaNet		35.6	55.4	38.0	20.3	39.7	46.3
		✓	**36.7**	**56.8**	**38.9**	**20.7**	**40.2**	**48.5**
ResNet-101-FPN	RetinaNet		37.7	57.7	40.3	21.7	42.1	49.9
		✓	**38.4**	**58.6**	**41.4**	**22.2**	**42.7**	**51.3**
ResNet-50-FPN	PA-FPN		37.5	58.4	40.7	21.1	41.7	48.2
		✓	**39.2**	**60.7**	**42.4**	**23.2**	**43.1**	**51.0**
	NAS-FPN		37.5	55.4	40.4	19.6	41.6	51.4
		✓	**39.9**	**58.5**	**43.1**	**20.7**	**44.3**	**55.4**
ResNet-101-FPN	PA-FPN		39.5	60.2	42.7	22.5	43.9	51.6
		✓	**41.1**	**62.4**	**44.7**	**24.7**	**44.8**	**53.7**
	NAS-FPN		37.8	56.0	40.6	18.6	42.4	52.9
		✓	**40.2**	**58.4**	**43.0**	**20.7**	**44.4**	**56.1**
EfficientDet-D0	EfficientDet		33.4	51.7	34.8	12.8	39.1	52.5
		✓	**34.6**	**53.4**	**36.5**	**14.0**	**40.2**	**53.0**

We conjecture that this is because Mask R-CNN and Cascade R-CNN themselves integrate mechanisms for better feature learning, which might overlap with the performance gains with our method. Specifically, Mask R-CNN benefits from extra accurate instance-level mask supervisions, while Cascade R-CNN enjoys IoU-specific multi-stage training to progressively refine object proposals and learn discriminative features for IoU-specific proposals. However, even in these cases, our method can also obtain $+1\%$ AP improvement over these competing baseline methods.

Additionally, we also conduct experiments that apply our HCE (with minor modification, see Section III-E) to one-stage detectors including SSD [14] and RetinaNet [17]. The comparision results are shown in the bottom of Table I. Our HCE framework can significantly improve the detection performance of SSD by 2.6% AP using VGG16 as backbone. We believe the behind reason lies in the fact: while the representational power of VGG16 (compared to ResNet) largely limits the classification ability of the detection head of SSD, the additional contextual information provided by our HCE framework to some extent makes up for the representational power of VGG. While for RetinaNet with ResNet-50-FPN and ResNet-101-FPN as backbones, our HCE method can still achieve about 1.0% AP improvements.

Finally, we also integrate our HCE framework with three state-of-the-art detectors, including PA-FPN [12], NAS-FPN [13] and EfficientDet [15]. As shown in Table I, our HCE framework achieves consistent improvements over these methods. In particular, our HCE improves NAS-FPN (both with ResNet-50-FPN and ResNet-101-FPN as backbones) by 2.4 AP on MS-COCO. Our experiments demonstrate that even state-of-the-art detectors have not fully exploited the potential of hierarchical context embedding in learning better representations for object detection, and thus can directly benefit from our HCE framework. We have added these experimental results and analysis in our revised manuscript.

C. Error Analyses

In the following, we perform error analyses to further understand in what aspects our HCE framework improves the region-based object detectors. Following the settings of [22], we choose the top N predictions for each category during inference time. Each prediction is classified based on the type of error:

- Correct: correct class and IOU $>$ 0.5
- Location Error: correct class and 0.1 $<$ IOU $<$ 0.5
- Background Error: IOU $<$ 0.1 for any object
- Classification Error: class is wrong and IOU $>$ 0.5
- Other: class is wrong and 0.1 $<$ IOU $<$ 0.5

We compare different error types between the FPN baseline and our method with ResNet-50 as backbone on MS-COCO 2017 `val`. Fig. 6 shows the results of each error type averaged across all 80 categories, and each error type for "`hot dog`", "`snowboard`" and "`baseball glove`" which are highly dependent on context information. Obviously, our method can effectively improve the classification ability of region-based detector and reduce the background errors to a large extent, without compromising the localization performance

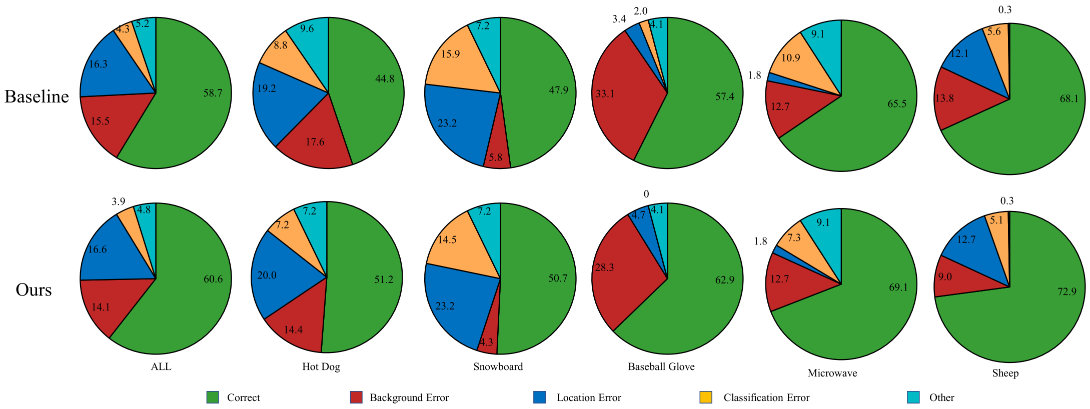


Fig. 6. Error analyses: These illustrations show the percentage of different error types in the top N detections (N = # objects in that category).

TABLE II

IMPACTS OF DIFFERENT CONTEXT EMBEDDING OPERATIONS ON MS-COCO 2017 VAL. "MLL" MEANS WE LEVERAGE THE IMAGE-LEVEL CATEGORICAL EMBEDDING MODULE TO ADVANCE THE LEARNING OF CONTEXT-DEPENDENT CATEGORIES. "INSTANCE" AND "GLOBAL" DENOTES THAT WE UTILIZE INSTANCE-LEVEL (CF. EQ (3)) OR GLOBAL-LEVEL (CF. EQ (4)) CONTEXTUAL FEATURES TO FURTHER IMPROVE THE REGION-WISE DETECTION HEAD

Method	MLL	Instance	Global	AP	AP^{50}	AP^{75}	AP^{S}	AP^{M}	AP^{L}
FPN				36.3	58.3	39.1	21.6	40.2	46.9
	✓			36.8	58.9	39.7	21.9	40.5	47.2
	✓	✓		37.8	59.9	40.9	22.2	41.4	48.9
	✓	✓	✓	**38.4**	**61.0**	**41.8**	**22.9**	**42.5**	**49.1**

or increasing other type of errors. Our improvements are particularly noticeable for context-dependent object categories. For example, the (normalized) correctly recognized instances of "`hot dog`" increase from 44.8% to 51.2%, while the background false positive detections reduce from 17.6% to 14.4%. These observations validate that our HCE framework can indeed improve the classification ability.

D. Ablation Studies

In this section, we conduct three series of ablation experiments to analyze the proposed method, using ResNet-50 as backbone on MS-COCO 2017 `val`.

1) Context Embedding Operations: We first investigate the impacts of different context embedding operations in our HCE framework. Specifically, there are three context embedding operations involved in our framework. Firstly, the image-level categorical embedding module employs multi-label learning (denoted as "MLL") to embed global-level context to advance the learning of context-dependent categories. Then, for further improving region-based classification, both the context-embedded instance-level feature (denoted as "Instance") and the context-aggregated global-level feature (denoted as "Global") are combined to generate hierarchical contextual RoI feature.

Table II shows the performance improvements by progressively integrating more context embedding operations. Solely applying MLL on the detection backbone gives 0.5% AP improvement. This verifies that image-level categorical embedding advances the feature learning for context-dependent object categories. Then, the context-embedded instance-level feature which can be directly utilized by the detection head brings another 1.0% AP improvement. Finally, global-level context embedding for contextual RoI feature improves 0.6% AP. These results suggest that the context embedding operations in our framework are complementary with each other.

2) Non-Local Module vs. HCE Framework: Wang *et al.* [18] proposed a non-local module which can be applied as a plug-and-play module on the backbone networks for various task. The non-local module enhances the capability of backbone network for capturing long-term contextual information, and achieves promising performance for many downstream tasks including object detection. However, our HCE framework significantly differs from the non-local module in two aspects. First, HCE explicitly imposes image-level categorical supervision signals upon the backbone networks for learning image-level concepts, while the non-local module is only trained by the backward signals from the downstream tasks. Secondly, HCE conveys the contextual cues into RoI-level representations, which has not been considered by the non-local module.

Due to above reasons, we believe that our HCE framework is complementary to the non-local module, while applied for region-based object detection. This is confirmed by the results in Table III, where our method improves over the non-local FPN baseline by 1.7% AP on COCO detection.

3) Fusion Strategies in Training: We consider the proposed two fusion strategies, feature fusion and confidence fusion, are complementary to each other. To verify this, we evaluate the performance by training the model with feature fusion and confidence fusion individually, as well as both of them. Table IV shows the results of different fusion strategies. Specifically, "FF Train" means that we apply feature fusion (FF) for training, while "CF Train" means confidence fusion (CF) are applied for training. Utilizing feature fusion and confidence fusion individually for training can outperform the baseline (FPN with MLL) by 0.8% and 0.6%

TABLE III

COMPARISONS OF NON-LOCAL MODULE [18] AND OUR HCE FOR CONTEXT EMBEDDING

Method	Non Local	HCE	AP	AP^{50}	AP^{75}	AP^{S}	AP^{M}	AP^{L}
FPN			36.8	58.9	39.7	21.9	40.5	47.2
	✓		37.2	59.3	40.1	22.1	41.3	47.3
		✓	38.4	61.0	41.8	22.9	42.5	49.1
	✓	✓	**38.9**	**61.7**	**41.9**	**23.5**	**42.9**	**50.0**

TABLE IV

EFFECTS OF DIFFERENT FUSION STRATEGIES DURING *TRAINING*, EVALUATED BY DETECTION PERFORMANCE ON MS-COCO 2017 VAL. THE MODELS SHARE THE SAME BACKBONE NETWORK RESNET50-FPN. "FF TRAIN" MEANS THAT WE APPLY FEATURE FUSION (FF) FOR TRAINING, WHILE "CF TRAIN" MEANS CONFIDENCE FUSION (CF) ARE APPLIED FOR TRAINING

Method	FF Train	CF Train	AP	AP^{50}	AP^{75}	AP^{S}	AP^{M}	AP^{L}
FPN			36.8	58.9	39.7	21.9	40.5	47.2
	✓		37.6	60.3	40.7	22.5	41.4	48.2
		✓	37.4	60.2	40.1	23.0	41.1	47.6
	✓	✓	**38.4**	**61.0**	**41.8**	**22.9**	**42.5**	**49.1**

TABLE V

EFFECTS OF DIFFERENT FUSION STRATEGIES IN TESTING, WHICH ARE EVALUATED BY THE INFERENCE TIME AND DETECTION PERFORMANCE ON MS-COCO 2017 VAL. NOTE THAT ALL MODELS ARE TRAINED WITH BOTH FUSION STRATEGIES. "FF TEST" DENOTES THAT WE EVALUATE THE FEATURE FUSION (FF) STRATEGY DURING INFERENCE, WHILE "CF TEST" MEANS THE RESULTS ARE EVALUATED BY CONFIDENCE FUSION (CF) STRATEGY. INFERENCE SPEED IS EVALUATED ON A SINGLE 1080TI GPU

Method	FF Test	CF Test	Speed	GFLOPs	AP	AP^{50}	AP^{75}	AP^{S}	AP^{M}	AP^{L}
FPN			0.087s	414.14	36.3	58.3	39.1	21.6	40.2	46.9
	✓		0.090s	428.06	38.2	60.8	41.5	22.6	42.2	49.0
		✓	0.094s	456.68	38.3	60.8	41.6	22.8	42.3	49.0
	✓	✓	0.100s	471.30	**38.4**	**61.0**	**41.8**	**22.9**	**42.5**	**49.1**

AP, respectively. Training with both fusion strategies achieves the best result, and is clearly better than using each individual fusion strategy separately.

4) Fusion Strategies in Testing: We also evaluate each fusion strategy independently during inference, with all HCE models trained with both fusion strategies. Table V shows the results of each fusion strategy and the combined fusion strategies. "FF Test" denotes that we evaluate the feature fusion (FF) strategy during inference, while "CF Test" means that the results are evaluated by confidence fusion (CF) strategy. We can see that once the model is trained with both fusion strategies, using only one fusion branch for inference will not cause obvious accuracy drop, but brings computational economy. For example, using the feature fusion branch for inference adds very minimal time cost (0.003s) to the baseline, but increases the AP from 36.3% to 38.2%. These results also prove the complementarity of the proposed two fusion strategies.

In Figure 7, we show the difference between the confusion matrix of confidence fused scores and original scores, by directly subtracting the confusion matrix of the original scores from the confusion matrix of confidence fused scores. The resulting confusion matrix clearly shows that the confidence fusion strategy can improve the true positive rate of the classification results. In Figure 8, we show some classification results with and without confidence fusion, where we observe that our fusion strategy can improve the classification accuracy under difficult conditions like extreme occlusion. In Figure 9, we show some activation maps with and without feature fusion, where we observe that our feature fusion strategy can absorb the context information and produce more accurate and wide activations on objects of interests.

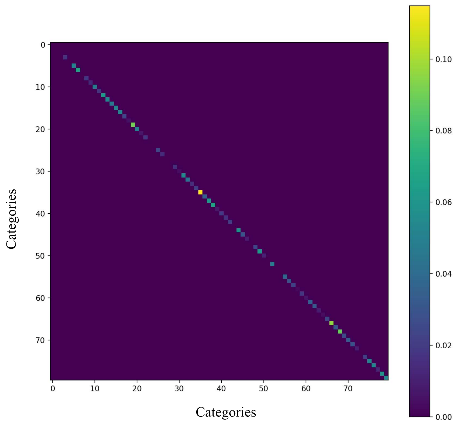


Fig. 7. Confusion matrix of the difference between confidence fusion scores and original scores.

5) Speed and Computational Cost: As shown in Table V, our feature fusion and confidence fusion components add extra computational cost to the baseline detectors. We note that our extra computational burden are only added to the detection head, and because the head has much less computation than the backbone network, our HCE has small computational overhead. An analogy can be observed in the famous Cascade R-CNN detector, which only brings marginal computational overhead to the detection head.

6) Global Pooling Strategies: We evaluate the effects of different global pooling strategies in our image-level categorical embedding module (cf. Section III-B). Table VI shows the results, where "GAP" means Global Average Pooling and "GMP" denotes Global Max Pooling. It is obvious to see that our HCE framework utilizing GAP or GMP individually for training can still outperform the baseline by 1.6% and 1.8% AP, respectively. Besides, utilizing both global pooling strategies for training achieves the best result. Similar observations

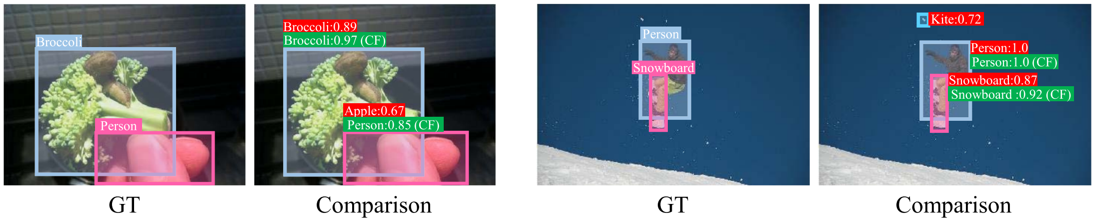


Fig. 8. Visualization of the classification results *without* (red box) and *with* (green box) confidence fusion.

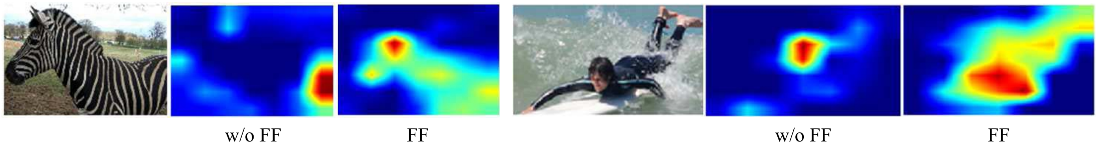


Fig. 9. Visualization of the activation maps *without* and *with* feature fusion.

TABLE VI

EFFECTS OF DIFFERENT GLOBAL POOLINGS ON MS-COCO 2017 VAL. "GAP" MEANS GLOBAL AVERAGE POOLING, WHILE "GMP" DENOTES GLOBAL MAX POOLING

Method	GAP	GMP	ML mAP	AP	AP^{50}	AP^{75}	AP^{S}	AP^{M}	AP^{L}
FPN			–	36.3	58.3	39.1	21.6	40.2	46.9
HCE FPN (Ours)	✓		78.1	37.9	60.4	40.7	22.6	41.8	48.5
		✓	80.2	38.1	60.7	41.2	**22.9**	42.1	48.6
	✓	✓	**83.3**	**38.4**	**61.0**	**41.8**	**22.9**	**42.5**	**49.1**

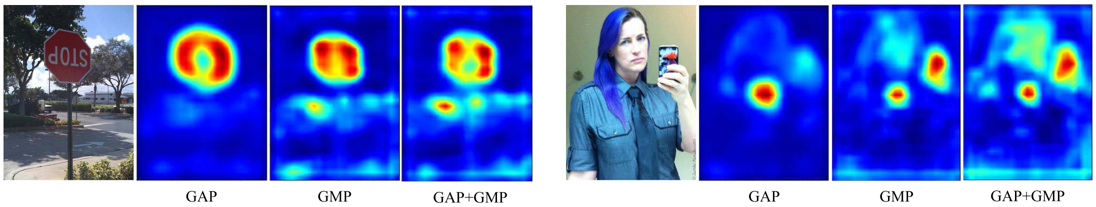


Fig. 10. Visualization of the activation maps from global average pooling (GAP), global max pooling (GMP), and GAP + GMP.

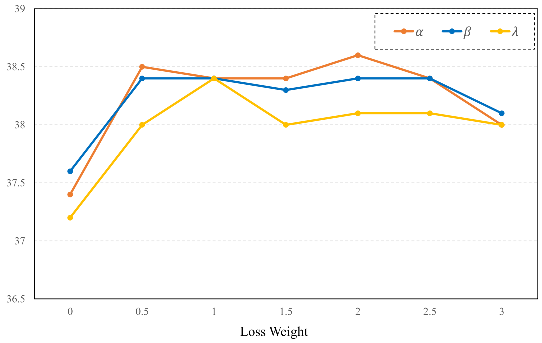


Fig. 11. Effects by setting different weights to the loss in our HCE framework.

are consistent with the observations in [36], which indicates that the GAP and GMP operations are complementary for feature aggregation.

As shown in Figure 10, we find that the activation by GAP typically concentrates on the main object in an image, while the activation by GMP may scatter across multiple objects. As a result, we believe that these two feature aggregation strategies are complementary to each other, and this can be verified by the activation maps that combine both GAP and GMP.

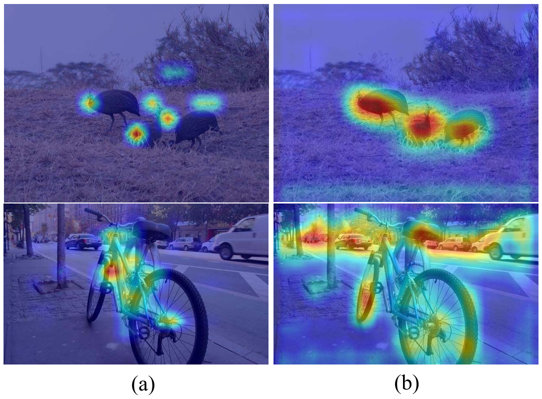


Fig. 12. Visualization of the activation maps from **(a)** detection backbone and **(b)** the multi-label classification features (*i.e.*, context-embedded image feature $\boldsymbol{X}$). It can be observed that the activation maps from the multi-label classification features generated by our image-level categorical embedding module typically have more proper activations on the whole objects of interests, while the activation maps produced by detection backbone may only have high responses on object parts.

7) Effects of Multi-Label Classification Accuracy: Table VI, we compare three different multi-label classifiers, by using GAP, GMP, and GAP + GMP for feature aggregation respectively. Multi-label classifier using GMP performs significantly better than using GAP (80.2 *VS.* 78.1), and combining GAP

TABLE VII

COMPARISONS WITH THE STATE-OF-THE-ART SINGLE-MODEL DETECTORS ON MS-COCO 2017 TEST-DEV. "*" DENOTES USING TRICKS (WITH BELLS AND WHISTLES) DURING INFERENCE

Method	Backbone	AP	AP^{50}	AP^{75}	AP^{S}	AP^{M}	AP^{L}
YOLOv3 [41]	Darknet-53	33.0	57.9	34.4	18.3	35.4	41.9
SSD513 [14]	Res101	31.2	50.4	33.3	10.2	34.5	49.8
RetinaNet* [17]	Res101-FPN	39.1	59.1	42.3	21.8	42.7	50.2
FCOS [42]	Res101-FPN	41.5	60.7	45.0	24.4	44.8	51.6
FPN [4]	Res101-FPN	36.2	59.1	39.0	18.2	39.0	48.2
Mask R-CNN [8]	Res101-FPN	38.2	60.3	41.7	20.1	41.1	50.2
Cascade R-CNN [5]	Res101-FPN	42.8	62.1	46.3	23.7	45.5	55.2
Deformable R-FCN* [43]	Aligned-Inception-ResNet	37.5	58.0	40.8	19.4	40.1	52.5
DCNv2* [44]	Res101-DeformableV2	46.0	**67.9**	**50.8**	**27.8**	49.1	**59.5**
IoU-Net [45]	Res101-FPN	40.6	59.0	–	–	–	–
TridentNet [46]	Res101	42.7	63.6	46.5	23.9	46.6	56.6
Cascade +Rank-NMS [47]	Res101-FPN	43.2	61.8	47.0	24.6	46.2	55.4
HCE RetinaNet	Res101-FPN	38.8	59.1	41.7	20.9	41.3	50.2
HCE FPN	Res101-FPN	41.0	63.5	44.7	23.4	44.2	52.2
HCE Mask R-CNN	Res101-FPN	41.6	63.9	45.4	23.7	44.7	53.1
HCE Cascade R-CNN	Res101-FPN	44.1	63.2	47.9	25.2	46.9	57.0
HCE Cascade R-CNN*	Res101-FPN	**46.5**	65.6	50.6	27.4	**49.9**	59.4

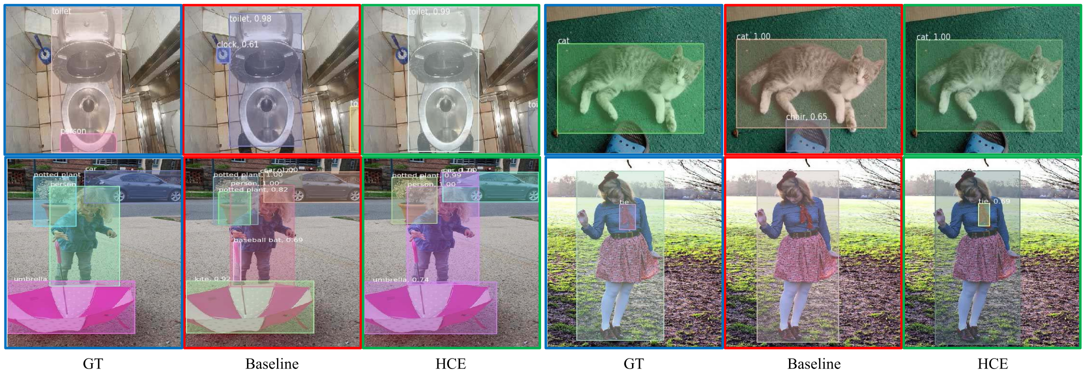


Fig. 13. Visualization and comparisons of the output results produced by the ground truth (*left with blue border*), the baseline FPN (*middle with red border*) and our HCE FPN (*right with green border*), respectively. These visualizations demonstrate that our HCE can effectively filter out false positive detections or aid in correctly classifying context-dependent objects. Best viewed in color.

and GMP achieves the best performance. From Table VI, we can draw two conclusions. Firstly, in general, more accurate multi-label classifier leads to better detection performance. Secondly, the final detection performance, to some extent, is robust to different classifiers, since the detection AP with the best classifier is 38.4, while the AP with the worst classifier is 37.9.

8) Effects of Different Loss Weights: We note that simplicity is central to our design, and we keep most of the hyper-parameters and loss terms of baseline detectors unchanged. Take the FPN detector for example, we do not modify the loss weights for RPN and the detection head, but we add the multi-label classification loss, and extend the loss of original detection head (a simplified version of Fast R-CNN) into two separate losses for our feature fusion detection head and confidence fusion head, respectively.

In Figure 11, we show the effects by setting different weights to the loss of our HCE framework. On one hand, our framework is robust to the variations of hyper-parameters in general. And on the other hand, we find that when carefully adjusting these hyper-parameters, we can achieve slightly better detection performance. However, to keep the simplicity of our method, we still set all hyper-parameters to 1 by default in our implementation.

E. Comparisons With State-of-the-Art

We compare our proposed method with state-of-the-art on MS-COCO 2017 `test-dev`. For fair comparisons, we report the performance of all methods with single-model inference. Specifically, we apply our method on RetinaNet, FPN, Mask R-CNN and Cascade R-CNN in 2× training scheme without bells and whistles. Table VII shows all comparison results.

Our hierarchical context embedding framework, when integrated with FPN, Mask R-CNN and Cascade R-CNN object detectors, consistently outperforms state-of-the-art object detectors using the same backbone network. When equipped with HCE, RetinaNet achieves comparable result with that trained and tested with bells and whistles. For fairly comparisons with Deformable R-FCN* and DCNv2* which adopt multi-scale 3× training scheme and multi-scale testing, we follow the same experimental setting to train our HCE Cascade R-CNN*. It gives an AP of 46.5%, which surpasses R-FCN* and DCNv2*. These results demonstrate the superior performance of the proposed context embedding framework.

F. Visualization and Comparisons

We first visualize the learned concepts for objects of interests from both the detection backbone and our image-level

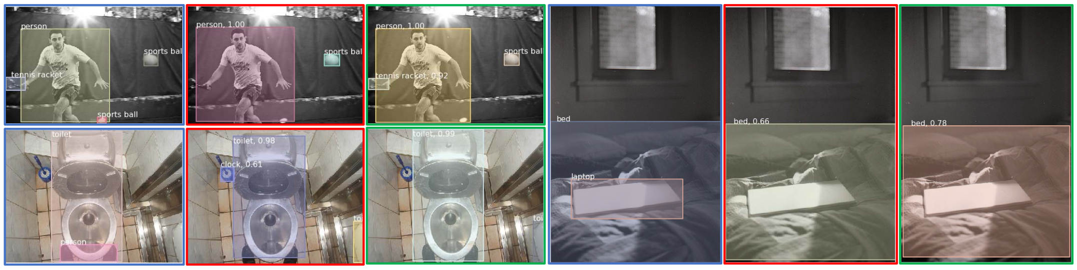


Fig. 14. Visualization and comparisons of the fail case output results produced by the ground truth (*left with blue border*), the baseline FPN (*left with red border*) and our HCE FPN (*right with green border*), respectively. Best viewed in color.

categorical embedding module. To this end, we show the activation maps generated by the last convolutional layer of FPN with ResNet-50 as backbone, and the context-embedded image feature generated by our image-level categorical embedding module. As shown in Fig. 12, the activation maps of our context-embedded image features typically have more proper activations on the whole objects of interests, while the activation maps produced by detection backbone may only have high responses on object parts.

We also give some detection example images in Fig. 13, produced by the ground truth (*left with blue border*), the baseline FPN (*middle with red border*) and our HCE FPN (*right with green border*), respectively. The "clock" in the first example image shows that our HCE framework can effectively filter out false positive background detection, while the "kite" and "umbrella" in the third example image shows that our method can assist in correctly classifying context-dependent objects. This visualization and comparisons further demonstrates the motivation of our HCE framework.

Besides, we give some failure cases in Figure 14. From this figure, we can draw two root causes for the failure of our HCE method. Firstly, as shown in the cases on the left of the Figure 14, when the object stands on the edge of the image, and is small or partially occluded, HCE can hardly extract discriminative features for such objects and may miss the detection of these objects. Secondly, as shown by the cases on the right of Figure 14, when the object of interest seldom appears in current context (e.g., the `laptop` on the `bed`) or the object itself lacks distinctive appearance, HCE can hardly extract useful context information to aid the recognition of such objects.

V. Conclusion

In this paper, we investigated the limitation of context information on conventional region-based detectors, and proposed a novel and effective Hierarchical Context Embedding (HCE) framework to facilitate the classification ability of current region-based detectors. Our HCE framework can also be conveniently adapted to benefit one-stage detectors. Comprehensive experiments demonstrated the consistent outperforming accuracy on almost all existing mainstream region-based detectors, include FPN, Mask R-CNN and Cascade R-CNN. In the future, we will concentrate in extending the usage scope of our HCE framework and adapting it to other vision tasks (*e.g,* semantic segmentation) that rely on contextual clues.

References

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

[2] R. Girshick, "Fast R-CNN," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Dec. 2015, pp. 1440–1448.

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

[4] T.-Y. Lin, P. Dollár, R. Girshick, K. He, B. Hariharan, and S. Belongie, "Feature pyramid networks for object detection," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 2117–2125.

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

[6] C.-D. Xu, X.-R. Zhao, X. Jin, and X.-S. Wei, "Exploring categorical regularization for domain adaptive object detection," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2020, pp. 11724–11733.

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

[8] K. He, G. Gkioxari, P. Dollár, and R. Girshick, "Mask R-CNN," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Oct. 2017, pp. 2961–2969.

[9] W. Luo, "Understanding the effective receptive field in deep convolutional neural networks," in *Proc. NIPS*, 2016, pp. 4905–4913.

[10] Z. Chen, S. Huang, and D. Tao, "Context refinement for object detection," in *Proc. ECCV*, 2018, pp. 71–86.

[11] V. Kantorov, M. Oquab, M. Cho, and I. Laptev, "ContextLocNet: Context-aware deep network models for weakly supervised localization," in *Proc. ECCV*, 2016, pp. 350–365.

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

[13] G. Ghiasi, T.-Y. Lin, and Q. V. Le, "NAS-FPN: Learning scalable feature pyramid architecture for object detection," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2019, pp. 7036–7045.

[14] W. Liu *et al.*, "SSD: Single shot multibox detector," in *Proc. ECCV*. Cham, Switzerland: Springer, 2016, pp. 21–37.

[15] M. Tan, R. Pang, and Q. V. Le, "EfficientDet: Scalable and efficient object detection," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2020, pp. 10781–10790.

[16] Z.-M. Chen, X. Jin, B. Zhao, X.-S. Wei, and Y. Guo, "Hierarchical context embedding for region-based object detection," in *Proc. ECCV*, 2020, pp. 633–648.

[17] T.-Y. Lin, P. Goyal, R. Girshick, K. He, and P. Dollár, "Focal loss for dense object detection," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Oct. 2017, pp. 2980–2988.

[18] X. Wang, R. B. Girshick, A. Gupta, and K. He, "Non-local neural networks," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, vol. 2018, pp. 7794–7803.

[19] L. Liu *et al.*, "Deep learning for generic object detection: A survey," *Int. J. Comput. Vis.*, vol. 128, no. 2, pp. 1–30, 2019.

[20] J. R. R. Uijlings, K. E. A. van de Sande, T. Gevers, and A. W. M. Smeulders, "Selective search for object recognition," *Int. J. Comput. Vis.*, vol. 104, no. 2, pp. 154–171, Apr. 2013.

[21] J. Dai, Y. Li, K. He, and J. Sun, "R-FCN: Object detection via region-based fully convolutional networks," in *Proc. NIPS*, 2016, pp. 379–387.

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

[23] C. Galleguillos and S. Belongie, “Context based object categorization: A critical survey,” *Comput. Vis. Image Understand.*, vol. 114, no. 6, pp. 712–722, Jun. 2010.

[24] A. Rabinovich, A. Vedaldi, C. Galleguillos, E. Wiewiora, and S. Belongie, “Objects in context,” in *Proc. ICCV*, 2007, pp. 1–8.

[25] H. Harzallah, F. Jurie, and C. Schmid, “Combining efficient object localization and image classification,” in *Proc. IEEE 12th Int. Conf. Comput. Vis.*, Sep. 2009, pp. 237–244.

[26] R. G. Cinbis, J. Verbeek, and C. Schmid, “Weakly supervised object localization with multi-fold multiple instance learning,” *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 39, no. 1, pp. 189–203, Jan. 2017.

[27] H. Bilen and A. Vedaldi, “Weakly supervised deep detection networks,” in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 2846–2854.

[28] H. Bilen, M. Pedersoli, and T. Tuytelaars, “Weakly supervised object detection with convex clustering,” in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2015, pp. 1081–1089.

[29] Z. Wang, T. Chen, G. Li, R. Xu, and L. Lin, “Multi-label image recognition by recurrently discovering attentional regions,” in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Oct. 2017, pp. 464–472.

[30] J. He, Z. Deng, L. Zhou, Y. Wang, and Y. Qiao, “Adaptive pyramid context network for semantic segmentation,” in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2019, pp. 7519–7528.

[31] W. Byeon, Q. Wang, R. K. Srivastava, and P. Koumoutsakos, “ContextVP: Fully context-aware video prediction,” in *Proc. ECCV*, 2018, pp. 753–769.

[32] X. Li, J. Wu, Z. Lin, H. Liu, and H. Zha, “Recurrent squeeze-and-excitation context aggregation net for single image deraining,” in *Proc. ECCV*, 2018, pp. 254–269.

[33] X. Zhang, R. Jiang, T. Wang, and W. Luo, “Single image dehazing via dual-path recurrent network,” *IEEE Trans. Image Process.*, vol. 30, pp. 5211–5222, 2021.

[34] E. Zablocki, P. Bordes, L. Soulier, B. Piwowarski, and P. Gallinari, “Context-aware zero-shot learning for object recognition,” in *Proc. ICML*, 2019, pp. 7292–7303.

[35] L. Qu, J. Tian, S. He, Y. Tang, and R. W. H. Lau, “DeshadowNet: A multi-context embedding deep network for shadow removal,” in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 4067–4075.

[36] S. Woo, J. Park, J.-Y. Lee, and I. S. Kweon, “CBAM: Convolutional block attention module,” in *Proc. ECCV*, 2018, pp. 3–19.

[37] H. Gunes and M. Piccardi, “Affect recognition from face and body: Early fusion vs. late fusion,” in *Proc. SMC*, vol. 4, 2005, pp. 3437–3443.

[38] M. Ebersbach, R. Herms, and M. Eibl, “Fusion methods for ICD10 code classification of death certificates in multilingual corpora,” in *Proc. CLEF*, 2017, pp. 1–8.

[39] K. Chen *et al.*, “MMDetection: Open MMLab detection toolbox and benchmark,” 2019, *arXiv:1906.07155*. [Online]. Available: http://arxiv.org/abs/1906.07155

[40] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei, “ImageNet: A large-scale hierarchical image database,” in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2009, pp. 248–255.

[41] J. Redmon and A. Farhadi, “YOLOv3: An incremental improvement,” 2018, *arXiv:1804.02767*. [Online]. Available: http://arxiv.org/abs/1804.02767

[42] Z. Tian, C. Shen, H. Chen, and T. He, “FCOS: Fully convolutional one-stage object detection,” in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2019, pp. 9627–9636.

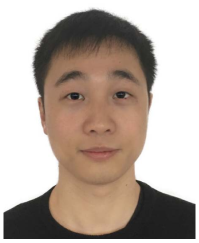
[43] J. Dai *et al.*, “Deformable convolutional networks,” in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Oct. 2017, pp. 764–773.

[44] X. Zhu, H. Hu, S. Lin, and J. Dai, “Deformable ConvNets v2: More deformable, better results,” in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2019, pp. 9308–9316.

[45] B. Jiang, R. Luo, J. Mao, T. Xiao, and Y. Jiang, “Acquisition of localization confidence for accurate object detection,” in *Proc. ECCV*, 2018, pp. 784–799.

[46] Y. Li, Y. Chen, N. Wang, and Z.-X. Zhang, “Scale-aware trident networks for object detection,” in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2019, pp. 6054–6063.

[47] Z. Tan, X. Nie, Q. Qian, N. Li, and H. Li, “Learning to rank proposals for object detection,” in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2019, pp. 8273–8281.

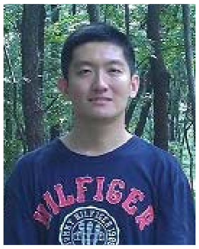


Zhao-Min Chen received the B.S. degree from Hunan University, China, in 2016, and the Ph.D. degree from the Department of Computer Science and Technology, Nanjing University, Nanjing, China, in 2021. He is currently an Associate Professor with Wenzhou University, China. His research interests include deep learning, computer vision, general object detection, and multi-label image recognition.

Xin Jin received the B.S., M.S., and Ph.D. degrees from the Department of Computer Science and Technology, Nanjing University of Aeronautics and Astronautics, Nanjing, China, in 2009, 2012, and 2017, respectively. He is currently a Researcher with Samsung Research Nanjing. His research interests include computer vision and deep learning, especially focusing on face landmark detection and general object detection.

Bo-Rui Zhao received the B.S. and M.S. degrees from the Department of Electronic Science and Engineering, Nanjing University, China, in 2016 and 2019, respectively. He is currently a Researcher with Megvii Research Nanjing. His research interests include computer vision, deep learning, and general object detection.

Xiaoqin Zhang received the B.Sc. degree in electronic information science and technology from Central South University, China, in 2005, and the Ph.D. degree in pattern recognition and intelligent system from the National Laboratory of Pattern Recognition, Institute of Automation, Chinese Academy of Sciences, China, in 2010. He is currently a Professor with Wenzhou University, China. He has published more than 100 papers in international and national journals and international conferences, including IEEE Transactions on Pattern Analysis and Machine Intelligence (T-PAMI), *IJCV*, IEEE Transactions on Image Processing (T-IP), IEEE Transactions on Neural Networks and Learning Systems (T-NNLS), IEEE Transactions on Computers (T-C), ICCV, CVPR, NIPS, IJCAI, AAAI, and among others. His research interests are in pattern recognition, computer vision, and machine learning.



Yanwen Guo (Member, IEEE) received the Ph.D. degree in applied mathematics from the State Key Laboratory of CAD&CG, Zhejiang University, China, in 2006. He worked as a Visiting Professor with the Department of Computer Science and Engineering, The Chinese University of Hong Kong, in 2006 and 2009, respectively, and the Department of Computer Science, The University of Hong Kong, in 2008, 2012, and 2013, respectively. From 2013 to 2015, he was a Visiting Scholar with the Department of Electrical and Computer Engineering, University of Illinois at Urbana-Champaign. He is currently a Professor with the National Key Laboratory for Novel Software Technology, Department of Computer Science and Technology, Nanjing University, Jiangsu, China. His research interests include image and video processing, vision, and computer graphics.