AUTHOR QUERIES

AUTHOR PLEASE ANSWER ALL QUERIES

PLEASE NOTE: We cannot accept new source files as corrections for your article. If possible, please annotate the PDF proof we have sent you with your corrections and upload it via the Author Gateway. Alternatively, you may send us your corrections in list format. You may also upload revised graphics via the Author Gateway.

Carefully check the page proofs (and coordinate with all authors); additional changes or updates WILL NOT be accepted after the article is published online/print in its final form. Please check author names and affiliations, funding, as well as the overall article for any errors prior to sending in your author proof corrections. Your article has been peer reviewed, accepted as final, and sent in to IEEE. No text changes have been made to the main part of the article as dictated by the editorial level of service for your publication.

AQ:1 = Please confirm or add details for any funding or financial support for the research of this article.

AQ:2 = Please confirm whether the edits made in the current affiliation of the author Bin-Bin Gao is correct.

Learning to Discover Multi-Class Attentional Regions for Multi-Label Image Recognition

Bin-Bin Gao and Hong-Yu Zhou

***Abstract*— Multi-label image recognition is a practical and challenging task compared to single-label image classification. However, previous works may be suboptimal because of a great number of object proposals or complex attentional region generation modules. In this paper, we propose a simple but efficient two-stream framework to recognize multi-category objects from global image to local regions, similar to how human beings perceive objects. To bridge the gap between global and local streams, we propose a multi-class attentional region module which aims to make the number of attentional regions as small as possible and keep the diversity of these regions as high as possible. Our method can efficiently and effectively recognize multi-class objects with an affordable computation cost and a parameter-free region localization module. Over three benchmarks on multi-label image classification, our method achieves new state-of-the-art results with a single model only using image semantics without label dependency. In addition, the effectiveness of the proposed method is extensively demonstrated under different factors such as global pooling strategy, input size and network architecture. Code has been made available at https://github.com/gaobb/MCAR.**

***Index Terms*— Multi-label, multi-class, two-stream, image recognition, attentional region, global to local.**

I. Introduction

CONVOLUTIONAL Neural Networks (CNNs) have made revolutionary breakthroughs on various computer vision tasks. For example, single-label image recognition (SLR), as a fundamental vision task, has surpassed human-level performance [1] on large-scale ImageNet. Unlike SLR, multi-label image recognition (MLR) needs to predict a set of objects or attributes of interest present in a given image. Meanwhile, these objects or attributes usually have complex variations like spatial location, object scale and occlusion *etc*.. Nonetheless, MLR still has wide applications such as scene understanding [2], face or human attribute recognition [3], [4] and multi-object perception [5] *etc*.. These make MLR become a practical and challenging task. In recent years, a significant amount of learning approaches have been proposed to dealing with multi-label data [6].

MLR can be simply addressed by using SLR framework to predict whether each category object presents or not. Recently, there are many works using deep CNNs to improve the performance of MLR. These works can be roughly divided into three types: spatial information [5], [7], visual attention [8]–[11] and label dependency [12]–[15].

Since the goal of MLR is to predict a set of object categories instead of producing accurate spatial locations of all possible objects, we argue that it is not necessary to waste computation resource for hundreds of object proposals in HCP [5] or consume labor cost for the bounding box annotation of objects in Fev+Lv [7]. RARL [8] and RDAL [9] introduce a reinforcement learning module and a spatial transformer layer to localize attentional regions, respectively, and sequentially predict label distribution based on generated regions. The main problem of these two methods is that the generated attentional regions are always category-agnostic and it is also difficult to guarantee the diversity of these local regions. In fact, we should ask the number of attentional regions to be as small as possible while maintaining the high diversity. Recently, MLGCN [14], [16] and SSGRL [15] try to model the label dependency with graph CNN to boost the performance of MLR. However, in this paper, we aim to improve the performance of MLR with only image semantics.

In order to exploit the semantic information of image, let us recall how we humans recognize multiple objects appeared in an image. Firstly, people may have a glimpse of a given image to discover some possible object regions from a global view. Then, these possible object regions guide the eye movements and help to make decisions on specific object categories following a region-by-region manner. In other words, most of time we humans difficultly recognize multi-objects using a single glance but at least two steps from a global view to local regions. In fact, there actually exists evidence in cognitive science that global visual processing precedes local reaction in visual perception [17]. Also, such global-to-local mechanism is supported by studies in neurobiology [18] and psychology [19]. In this paper, we wonder if machines can acquire the learning ability like humans to recognize multi-objects.

Inspired by this observation, we propose a novel multi-label image recognition framework with Multi-Class Attentional Regions (MCAR) as illustrated in Fig. 1. This framework contains a global image stream, a local region stream, and a multi-class attentional region module. Firstly, the global image stream takes an image as the input for a deep CNN and learns global representations supervised by the corresponding labels. Then, the multi-class attentional region module is used to discover possible object regions with the information from

AQ:1 Manuscript received October 10, 2020; revised April 9, 2021 and May 29, 2021; accepted June 5, 2021. The associate editor coordinating the review of this manuscript and approving it for publication was Dr. Dong Tian. *(Corresponding author: Bin-Bin Gao.)*

AQ:2 Bin-Bin Gao is with the Tencent YouTu Laboratory, Shenzhen 518057, China (e-mail: gaobb@lamda.nju.edu.cn).

Hong-Yu Zhou is with the Department of Computer Science, The University of Hong Kong, Hong Kong (e-mail: whuzhouhongyu@gmail.com).

Digital Object Identifier 10.1109/TIP.2021.3088605

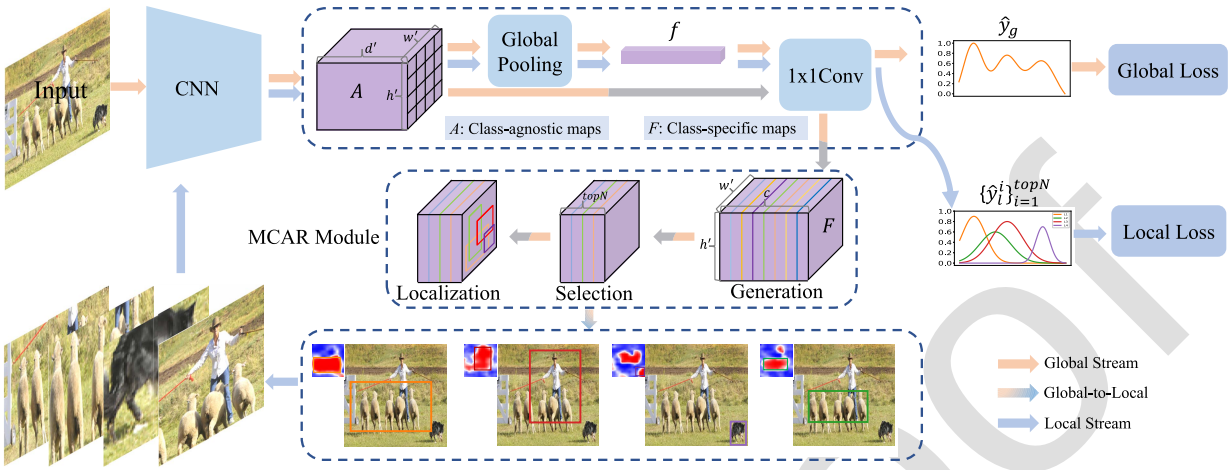


Fig. 1. The pipeline of our MCAR framework for multi-label image recognition. MCAR firstly feeds an input image into a deep CNN model to extract its global feature representation through the global image stream. Then, the multi-class attentional region module roughly localizes possible object regions by integrating that information from the global stream. Finally, these localized regions are fed to the shared CNN to obtain their predicted class distributions through the local region stream. At the inference stage, MCAR aggregates predictions from global and local streams with category-wise max-pooling and produces the final prediction.

the global stream, which is similar to the way we recognize multiple objects. Finally, these localized regions are fed to the *shared* CNN to obtain their predicted class distributions using the local region stream. The local region stream can recognize objects better since it flexibly focuses on details of each object which helps to alleviate the difficulty of recognition for these objects at different spatial locations and object scales.

The contributions of this paper can be summarized as follows.

- Firstly, we present a multi-label image recognition framework that can efficiently and effectively recognize multi-objects following a global to local manner. To the best of our knowledge, the learning mechanism of global to local in a unified model is the first time being proposed to find possible regions for multi-label images.
- Secondly, we propose a simple but effective multi-class attentional region module which includes three steps: generation, selection, and localization. In practice, it can dynamically generate a small number of attentional regions while keeping their diversity as high as possible.
- Thirdly, we achieve new state-of-the-art results on three widely used benchmarks with only a single model. Our method provides an affordable computation cost and needs no extra parameters.
- In addition, we also extensively demonstrate the effectiveness of the proposed method under different conditions like global pooling strategy, input size and network architecture.

The rest of this paper is organized as follows. We first review the related work in Section II. Then, Section III proposes our approach, including two-stream framework, MCAR module (from global to local) and two-stream learning. After that, the experiments are reported in Section IV. Finally, Section V presents discussions and the conclusion is given in Section VI.

II. Related Works

Recently, many efforts have been devoted into multi-label image recognition, using spatial information [5], [7], visual attention [8]–[11] and label dependency [12]–[15]. In this section, we briefly review these related approaches.

A. Spatial Information

How to utilize the spatial information of image is very crucial for almost all visual recognition tasks such as image recognition [20], [21], object detection [22] and semantic segmentation [23], [24]. It is closely related to how to design (or learn) effective features. The reason is that objects usually present with different scales at different spatial locations. HCP [5] uses EdgeBox [25] or BING [26] to generate hundreds of object proposals for each image using a like RCNN [22] method, and aggregates prediction scores of these proposals to obtain the final prediction. However, a large number of proposals usually bring a huge computation cost. Fev+Lv [7] generates proposals using bounding box annotations. Their approach combined the local proposal features and global CNN features to produce the final feature representations. It reduces the number of proposals but introduces the labor cost of annotation.

B. Visual Attention

Attention mechanism has been widely used in many vision tasks, such as visual tracking [27], fine-grained image recognition [28], image captioning [29], image question answering [30], and semantic segmentation [31]. RARL [8] uses a recurrent attention reinforcement learning module [32] to localize a sequence of attention regions and further predict label scores conditioned on these regions. Instead of reinforcement learning in RARL, RDAL [9] introduces a spatial transformer layer [33], [34] for localizing attentional regions from an image and an LSTM unit to sequentially predict

the category distribution based on features of these localized regions. Unlike RARL and RDAL, SRN [10] and ACfs [11] combine attention regularization loss and multi-label loss to improve performance. Specifically, SRN [10] captures both spatial semantic and label correlations based on the weighted attention map, while ACfs [11] enforces the network to learn attention consistency that the classification attention map should follow the same transformation when input image is spatially transformed.

C. Label Dependency

In order to exploit label dependency, CNN-RNN [12] jointly learns image feature and label correlation in a unified framework composed of a CNN module and an LSTM layer. The limitation is that it requires a pre-defined label order for model training. Similar to [12], [35] also jointly learn multi-label classifiers with both spatial object relationships and semantic label correlations. Order-Free RNN [13] relaxes the label order constraint via learning visual attention model and a confidence-ranked LSTM. But it requires an explicit module for removing duplicate prediction labels and needs a threshold for stopping the sequence outputs. In order to alleviate the issues, PLA [36] proposes two alternative losses which dynamically order the labels based on the prediction label sequence of an LSTM model. Recently, SSGRL [15] directly uses a graph convolutional network to model the label dependency among all labels.

There have been some other attempts on multi-label researches, such as multi-label image retrieval [37], multi-label dictionary learning [38], zero-shot [39], [39] and few-shot [40] multi-label classification. While in this paper, we deliberately avoid using any information from label dependency and aim to improve the performance of multi-label recognition with only image semantic information. We leave them as future works to further boost recognition performance or extend application fields by integrating the label correlation and other paradigms to our framework.

III. MCAR Framework

In this section, we firstly present a two-stream framework which contains a global image stream and a local region stream. Then, we elaborate the multi-class attentional region module, which tries to bridge the gap between global and local views. Finally, we present the optimization details of our framework.

A. Two-Stream Framework

1) Global Image Stream: Given an input image $I \in \mathbb{R}^{h\times w\times 3}$, where h, w are the image's height and width. Let's denote its corresponding label as $\boldsymbol{y} = [y^1, y^2, \cdots, y^C]^T$, where y^i is a binary indicator. $y^i = 1$ if image I is tagged with label i, otherwise $y^i = 0$. C is the number of all possible categories in this dataset.

We assume that $A = \mathcal{F}(I;\boldsymbol{\theta})$ is the activation map of the last convolutional layer of a CNN, where θ denotes the parameters of the CNN and $A \in \mathbb{R}^{h'\times w'\times d'}$. Then, a global pooling function $\mathcal{P}(\cdot)$ encodes the activation map A to a single vector $\boldsymbol{f} \in \mathbb{R}^{1\times 1\times d'}$, *i.e.*, $\boldsymbol{f} = \mathcal{P}(A)$. Here $\boldsymbol{f}$ can be considered as a global feature representation of the image I. In order to get it's prediction score, a 1×1 fully convolutional layer transfers $\boldsymbol{f}$ to $\boldsymbol{x} \in \mathbb{R}^C$ by

$$\boldsymbol{x} = W^T\boldsymbol{f} + \boldsymbol{b}. \tag{1}$$

We then use a sigmoid function $\sigma(\cdot)$ to turn $\boldsymbol{x}$ into a range [0, 1], that is

$$\hat{\boldsymbol{y}}_{\boldsymbol{g}} = \frac{1}{1+\exp(-\boldsymbol{x})}, \tag{2}$$

where $\hat{\boldsymbol{y}}_{\boldsymbol{g}}$ stands for the global prediction distribution.

2) Local Regions Stream: Local stream is, in fact, to perform a multi-instance multi-label learning [41]. By decomposing an image into object regions, each image becomes a bag containing several positive instances, *i.e.*, regions with the target objects, and negative instances, *i.e.*, regions with background or other objects. We assume that $\{L_1, L_2, \cdots, L_N\}$ is a set of N local regions cropped from input image I. These local regions are firstly resized to the input size by bilinear upsampling. Then, they are fed to the shared CNN (with the global stream) to get prediction distributions $\{\hat{\boldsymbol{y}}_{\boldsymbol{L_1}}, \hat{\boldsymbol{y}}_{\boldsymbol{L_2}}, \cdots, \hat{\boldsymbol{y}}_{\boldsymbol{L_N}}\}$ with Eq. 1 and 2. Finally, these local region distributions are aggregated by a category-wise max-pooling operation:

$$\hat{y}^i_l = \max\left(\hat{y}^i_{L_1}, \hat{y}^i_{L_2}, \cdots, \hat{y}^i_{L_N}\right), \tag{3}$$

where $\hat{y}^i_l$ is the i-th category score of the local prediction $\hat{\boldsymbol{y}}_{\boldsymbol{l}}$. The subscript l means the distribution is from N local regions.

Note that *the global and local streams share the same network without introducing additional parameters*. It is obviously different from the classical two-stream architecture which usually contains two parallel subnetworks. The inputs of our two-stream are the whole image and local regions from it, respectively. These local regions are dynamically generated by using the information of the global stream. Therefore, it is also different from the existing methods whose inputs are always two parallel views like video frame and optical flow in video classification [42].

During the training stage, we jointly train these two streams. At the early stage of learning, there may be little difference between the number of positive and negative instances (local regions). With the gradual convergence of the global stream, positive instances will dominate the local stream and thus also tend to converge. At the inference stage, we fuse the predictions from global stream ($\hat{\boldsymbol{y}}_{\boldsymbol{g}}$) and local stream ($\hat{\boldsymbol{y}}_{\boldsymbol{l}}$) with a category-wise max-pooling operation to generate the final predicted distribution of image I.

B. From Global to Local

Potential object regions are not available in image-level labels, which must be generated in an efficient manner. The desirable generation module and candidate regions should satisfy some basic principles. First, the diversity of candidate regions should be as high as possible such that they can cover all possible objects of a given multi-label image. Second, the number of these candidate regions should be as small

as possible in order to ensure efficiency. In contrast, more candidate regions require more computation resources since these regions need to be fed to the shared CNN simultaneously. Last but not least, the candidate regions generation module should have a simple network architecture and few parameters to alleviate the computation cost and storage overhead.

1) Attentional Maps Generation: The class activation mapping method [43] intuitively shows the discriminative image regions and helps us understand how to identify a particular category with a CNN. To obtain class-specific activation maps, we directly apply the 1×1 convolutional layer to the class-agnostic activation maps A from the global stream, that is

$$F = W^T A + \boldsymbol{b}, \tag{4}$$

where $F \in \mathbb{R}^{h' \times w' \times c}$. The class-specific activation map of the i-th category is denoted as $F^i \in \mathbb{R}^{h' \times w'}$ and it directly indicates the importance of the activation map at spatial leading to the classification of an image to class i.

The discriminative class regions of a specific F^i are significantly different among all possible class maps $\{F^i\}_{i=1}^C$. If we employ class maps $\{F^i\}_{i=1}^C$ to localize the potential object regions then it is easy to satisfy the first principle: to increase the diversity of different proposals.

2) Attentional Maps Selection: The number of class activation maps is equal to that of all categories associated with a dataset. For example, there are 20 and 80 categories on PASCAL VOC and MS-COCO datasets, respectively. If we use all class maps, it leads to two problems. First, the generated regions are too many to ensure efficiency. Second, a majority of regions will be redundant or meaningless because an image usually consists of a few instances.

A fact is that the predicted distribution will be close to the ground-truth distribution with the learning of the network which is supervised by ground-truth labels. It is a reasonable assumption that the high category confidence means that the corresponding object presents on the image with a high probability. Therefore, we sort the predicted scores $\hat{\boldsymbol{y}}_{\boldsymbol{g}}$ (whose dimension is equal to the number of classes) following a descending order and select the $topN$ class attentional maps. In experiments, we can see that a satisfied performance can be achieved when the $topN$ is a small number (such as 2 or 4) which is *far less than the number of all categories*. Another benefit is that the proposed method may force network to implicitly learn label correlation if selective attentional maps don't fully cover all object categories. This is because the local stream is also supervised by the ground-truth label distribution.

3) Local Regions Localization: We still denote $topN$ class attentional maps as $\{F^i\}_{i=1}^{topN}$ for notation simplification. Each F^i is normalized to the range $[0, 1]$ by a sigmoid function (Eq. 2). Furthermore, we simply upsample F^i to the input size to align the spatial semantics between F^i and the input image I.

The value of $F^i(x, y)$ represents a probability that it belongs to the i-th category at spatial location (x, y). In order to efficiently localize regions of interest, we decompose each selective attentional map F^i into a row and a column marginal distribution, which represents a probability distribution

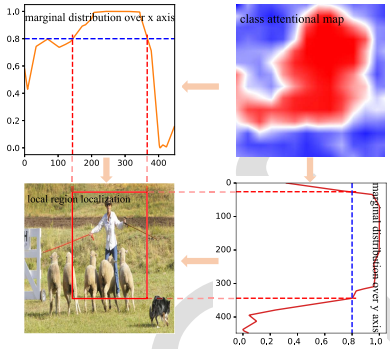


Fig. 2. The visualization of local region localization with class attentional map. We firstly decompose the class attentional map into two marginal distributions along row and column. Then, the class attentional region is localized by these two marginal distributions.

of objects present at the corresponding location (as shown in Fig. 2). We compute the marginal distribution based on the class attentional map F^i over x and y axis, respectively, which is

$$\begin{aligned} \boldsymbol{p}'_x &= \max_{1 \le y \le h} F^i(x, y), \\ \boldsymbol{p}'_y &= \max_{1 \le x \le w} F^i(x, y). \end{aligned} \tag{5}$$

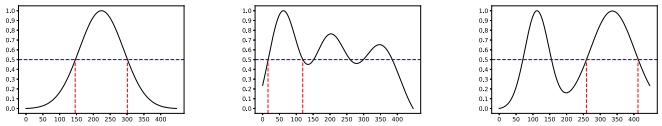
Then, $\boldsymbol{p}'_x$ and $\boldsymbol{p}'_y$ are normalized by min-max normalization such that the distribution is scaled to the range in $[0, 1]$, that is

$$\begin{aligned} \boldsymbol{p}_x &= \big(\boldsymbol{p}'_x - \min_i(p'^i_x)\big) / \big(\max_i(p'^i_x) - \min_i(p'^i_x)\big), \\ \boldsymbol{p}_y &= \big(\boldsymbol{p}'_y - \min_j(p'^j_y)\big) / \big(\max_j(p'^j_y) - \min_j(p'^j_y)\big), \end{aligned} \tag{6}$$

where p'^i_x represents the i-th element of $\boldsymbol{p}'_x$. In order to localize one discriminative region, we need to solve the following integer inequalities:

$$\begin{aligned} p_x{}^i &\ge \tau, \quad s.t.\ i = \{1, 2, \cdots, w\}, \\ p_y{}^j &\ge \tau, \quad s.t.\ j = \{1, 2, \cdots, h\}, \end{aligned} \tag{7}$$

where $\tau \in (0, 1)$ is a constant threshold. The solution of Eq. 7 may be a single interval or a union of multiple ones, and each interval corresponds to the spatial location of a specific object region. The fact is that $\boldsymbol{p}_x$ or $\boldsymbol{p}_y$ may have one peak when input image only contains an object in Fig. 3a and also may have multiple peaks when input image consists of multiple objects of the same category at different spatial locations in Fig. 3b and 3c. However, our objective is to recognize multi-class objects in a given image, and only one discriminative region needs to be selected for each category. Therefore, some constraints have to be added such that a unique interval among multiple feasible intervals can be chosen. To achieve this goal, we pick the interval contained in the global maximum peak for the case of multiple local maximum peaks as shown in Fig. 3b and choose the widest interval for multiple global maximum peaks as shown in Fig. 3c. For all selected $topN$ class attentional maps, $topN$ discriminative regions would be generated by solving the Eq. 7 conditioned on the above constraints.



(a) single peak (b) multiple peaks (c) multiple peaks

Fig. 3. Some examples of margin distribution. Black curves represent the margin distribution, and blue dash is the threshold τ, and the best interval between two red dashes is the desirable localization.

C. Two-Stream Learning

Given a training dataset $\{I_i, \boldsymbol{y_i}\}_{i=1}^{M}$, which I_i is the i-th image and $\boldsymbol{y_i} = [y_i^1, \cdots, y_i^C]^T$ represents the corresponding labels. The learning goal of our framework is to find $\boldsymbol{\theta}$, W and $\boldsymbol{b}$ via jointly learning global and local streams in an end-to-end manner. Thus, our overall loss function is formulated as the weighted sum of two streams,

$$\mathcal{L} = \mathcal{L}_g + \mathcal{L}_l, \tag{8}$$

where $\mathcal{L}_g$ and $\mathcal{L}_l$ represent the global and the local loss, respectively. Specifically, we adopt the binary cross entropy loss for global and local stream,

$$\begin{aligned} \mathcal{L}_g &= \sum_{i=1}^{M}\sum_{j=1}^{C} y_i^j \log(\hat{y_g}_i^j) + (1 - y_i^j)\log(1 - \hat{y_g}_i^j) \\ \mathcal{L}_l &= \sum_{i=1}^{M}\sum_{j=1}^{C} y_i^j \log(\hat{y_l}_i^j) + (1 - y_i^j)\log(1 - \hat{y_l}_i^j), \end{aligned} \tag{9}$$

where $\hat{y_g}_i^j$ and $\hat{y_l}_i^j$ are the prediction scores of the j-th category of the i-th image from global and local streams, respectively. Optimization is performed using SGD and standard back propagation.

IV. Experiments

In this section, we firstly report extensive experimental results and comparisons that demonstrate the effectiveness of the proposed method. Then, we present ablation studies to carefully evaluate and discuss the contribution of the crucial components in our MCAR.

A. Experiment Setting

1) Implementation Details: We perform experiments to validate the effectiveness of the proposed MCAR on three benchmarks in multi-label classification: MS-COCO [46], PASCAL VOC 2007 and 2012 [47], using the open-source framework PyTorch.

Following recent MLR works, we compare the proposed method with state-of-the-arts using the powerful ResNet-50 and ResNet-101 [48] models. Some popular and lightweight models, such as MobileNet-v2 [49], are also used to further evaluate our method. In general, for each of these networks we remove the fully-connected layers before the final output and replace them with global pooling followed by a 1×1 convolutional layer and a sigmoid layer. These models are all pre-trained on ImageNet and we train them using image-level labels only. The stochastic gradient descent (SGD) optimizer is used with the momentum of 0.9 and the weight decay of 0.0001. The initial learning rate is set to 0.001 for all layers but 0.01 for the 1×1 convolution, and they are decreased by a factor of 10 in the 30^{th} and 50^{th} epoch and the network is trained for 60 epochs in total.

During training, all input images are resized into a fixed size (*i.e.*, 256×256 or 448×448) with random horizontal flips and color jittering for data augmentation. In order to speed up the convergence of the network, we don't use the random crop although it can bring performance improvement but need more training time. Unless otherwise stated, we set $topN$ as 4 and τ as 0.5 in our experiments. The effects of hyper-parameters ($topN$ and τ) is discussed in Section IV-C.

2) Evaluation Metrics: The performance of MLR mainly employ two metrics which are the average precision (AP) for each category and the mean average precision (mAP) overall categories. We first employ AP and mAP to evaluate all the methods. Following conventional setting [5], [14], [15], we also compute the precision, recall and F1-measure for comparison performance on MS-COCO dataset. For each image, we assign a positive label if its prediction probability is greater than a threshold (0.6) and compare them with the ground-truth labels. The overall precision (OP), recall (OR), F1-measure (OF1) and per-category precision (CP), recall (CR), F1-measure (CF1) are computed as follows:

$$\begin{aligned} \mathrm{OP} &= \frac{\sum_i M_c^i}{\sum_i M_p^i}, \quad \mathrm{OR} = \frac{\sum_i M_c^i}{\sum_i M_g^i}, \\ \mathrm{CP} &= \frac{1}{C}\sum_i \frac{M_c^i}{M_p^i}, \quad \mathrm{CR} = \frac{1}{C}\sum_i \frac{M_c^i}{M_g^i}, \\ \mathrm{OF1} &= \frac{2 * \mathrm{OP} * \mathrm{OR}}{\mathrm{OP} + \mathrm{OR}}, \quad \mathrm{CF1} = \frac{2 * \mathrm{CP} * \mathrm{CR}}{\mathrm{CP} + \mathrm{CR}}, \end{aligned} \tag{10}$$

where M_c^i is the number of images correctly predicted for the i-th category, M_p^i is the number of predicted images for the i-th category, M_g^i is the number of ground truth images for the i-th category. We also compute these above metrics via another way that each image is assigned labels with top3 highest score. It is worthy to notice that these metrics may be affected by the threshold. Among these metrics, OF1 and CF1 are more stable than OP, CP, OR and CR. AP and mAP are the most important metrics which can provide a more comprehensive comparison.

B. Comparisons With State-of-the-Arts

To verify the effectiveness of our method, we compare the proposed method with state-of-the-arts on MS-COCO [46] and PASCAL VOC 2007 & 2012 [47].

1) MS-COCO: MS-COCO [46] is a widely used dataset to evaluate multiple tasks such as object detection, semantic segmentation and image caption, and it has been adopted to evaluate multi-label image recognition recently. It contains 82,081 images as the training set and 40,137 images as validation set and covers 80 object categories. Compared to VOC 2007 & 2012 [47], both the size of training set and the number of object categories are increased.

TABLE I

COMPARISONS OF MAP, CP, CR, CF1 AND OP, OR, OF1 IN % OF OUR MODEL AND STATE-OF-THE-ART METHODS ON THE MS-COCO DATASET. * INDICATES THAT THE RESULTS ARE REPRODUCED BY USING THE OPEN-SOURCE CODE [15], AND - DENOTES THE CORRESPONDING RESULT IS NOT PROVIDED

Methods	Input Size	Backbone	mAP	All						Top3					
				CP	CR	CF1	OP	OR	OF1	CP	CR	CF1	OP	OR	OF1
CNN-RNN [12]	-	VGG16	61.2	–	–	–	–	–	–	66.0	55.6	60.4	69.2	66.4	67.8
RDAL [9]	-	VGG16	–	–	–	–	–	–	–	79.1	58.7	67.4	84.0	63.0	72.0
Order-Free RNN [13]	-	ResNet-152	–	–	–	–	–	–	–	71.6	54.8	62.1	74.2	62.2	67.7
ML-ZSL [44]	-	ResNet-152	–	–	–	–	–	–	–	74.1	64.5	69.0	–	–	–
SRN [10]	224×224	ResNet-101	77.1	81.6	65.4	71.2	82.7	69.9	75.8	85.2	58.8	67.4	87.4	62.5	72.9
ACfs [11]	288×288	ResNet-101	77.5	77.4	68.3	72.2	79.8	73.1	76.3	85.2	59.4	68.0	86.6	63.3	73.1
PLA [36]	288×288	ResNet-101	–	80.4	68.9	74.2	81.5	73.3	77.1	–	–	–	–	–	–
ResNet-101 [45]	448×448	ResNet-101	–	73.8	72.9	72.8	77.5	75.1	76.3	78.3	63.7	69.5	83.8	64.9	73.1
Multi-Evidence [45]	448×448	ResNet-101	–	80.4	70.2	74.9	85.2	72.5	78.4	84.5	62.2	70.6	89.1	64.3	74.7
SSGRL* [15]	448×448	ResNet-101	81.9	84.2	70.3	76.6	85.8	72.4	78.6	88.0	63.1	73.5	90.2	64.5	75.2
MCAR	288×288	ResNet-101	80.5	81.8	69.2	75.0	84.9	72.2	78.0	85.8	62.6	72.4	88.9	64.7	74.9
Baseline	448×448	ResNet-101	77.1	72.7	72.3	72.5	77.4	75.5	76.5	77.8	63.5	69.9	84.0	65.5	73.6
MCAR	448×448	ResNet-101	**83.8**	**85.0**	**72.1**	**78.0**	**88.0**	**73.9**	**80.3**	**88.1**	**65.5**	**75.1**	**91.0**	**66.3**	**76.7**
SSGRL [15]	576×576	ResNet-101	83.8	**89.9**	68.5	76.8	**91.3**	70.8	79.7	**91.9**	62.5	72.7	**93.8**	64.1	76.2
MCAR	576×576	ResNet-101	**84.5**	84.3	**73.9**	**78.7**	86.9	**76.1**	**81.1**	87.8	**65.9**	**75.3**	90.4	**67.1**	**77.0**

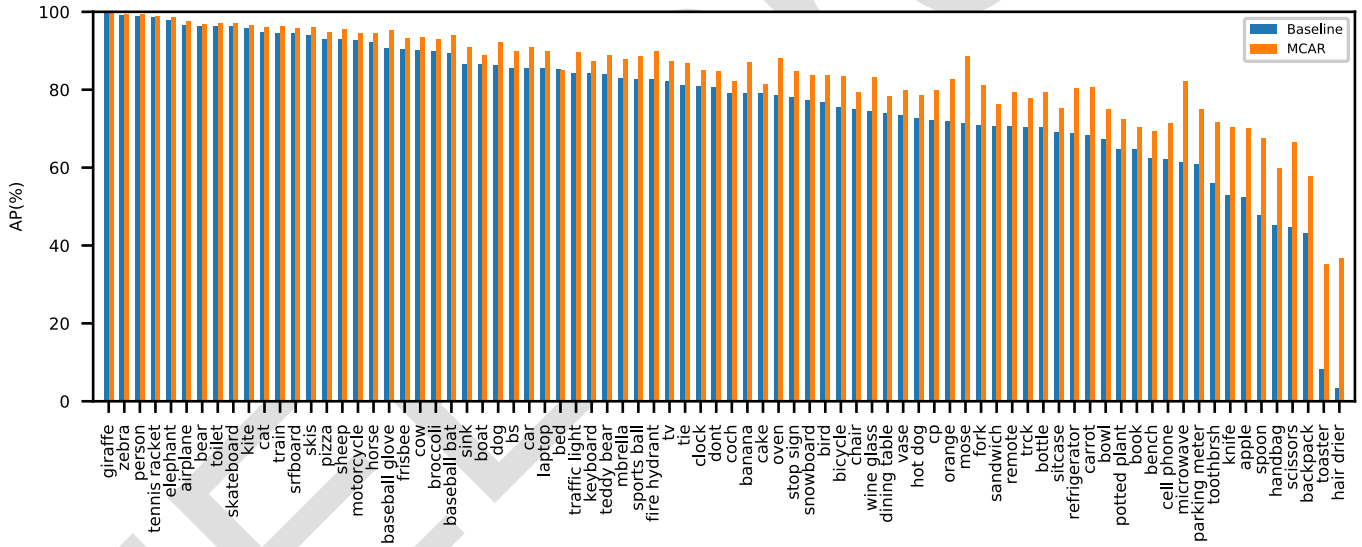


Fig. 4. AP (in %) of each category of our proposed framework and the ResNet-101 baseline on MS-COCO dataset. Our MCAR has significant improvements on almost all categories, especially for some difficult categories such as "toaster" and "hair drier".

Meanwhile, the number of labels of different images, the scale of different objects and the number of images in each category vary considerably, which makes it more challenging.

2) Results on MS-COCO: The results on MS-COCO are reported in Table I. When the input size is 448 × 448 (the most common setting in MLR), our method is already comparable to the state-of-the-art SSGRL [15] which uses additional label dependency and larger input to boost performance. Moreover, if we simply resize the input image to 576 × 576 during the testing stage while still using the model weights trained with 448 × 448 inputs, our method achieves 84.5% mAP which outperforms the SSGRL by 0.7%. In order to fairly compare with the SSGRL, we re-implement the experiment with 448×448 input following the same setting as described in the SSGRL. In Table I, we can see that our method significantly beats the SSGRL and improves it by 1.9 points (83.8% vs. 81.9%). Note that PLA [36] models label correlation through exploiting LSTM model. Using the same input size (288 × 288), our method gets higher F1 scores than PLA, which further indicates that it is very important to exploit image semantics for multi-label image recognition.

The performance of our method is also significantly better than that of Multi-Evidence [45], and it improves CF1 by 3.1%, OF1 by 1.9%, CF1-top3 by 4.5%, OF1-top3 by 2.0%. Note that our baseline ResNet-101 model achieves 77.1% mAP, which should be close to that of the baseline of Multi-Evidence [45] because of nearly the same F1-measures. In comparison to the baseline, our method is 6.7% higher in mAP (83.8% vs. 77.1%).

Meanwhile, we show the AP performance of each class for further comparison with the baseline model in Fig 4. It is obvious that our method has significant improvements on almost all categories, especially for some difficult categories such as "toaster" and "hair drier". In short, MCAR outperforms all state-of-the-art methods and significantly surpasses

TABLE II
COMPARISONS OF AP AND MAP IN % OF OUR MODEL AND STATE-OF-THE-ART METHODS ON THE PASCAL VOC 2007. ∗ INDICATES METHODS USING LARGER INPUT SIZE (576 × 576)

Methods	Backbone	aero	bike	bird	boat	bottle	bus	car	cat	chair	cow	table	dog	horse	mbike	person	plant	sheep	sofa	train	tv	mAP
CNN-RNN [12]	VGG16	96.7	83.1	94.2	92.8	61.2	82.1	89.1	94.2	64.2	83.6	70.0	92.4	91.7	84.2	93.7	59.8	93.2	75.3	**99.7**	78.6	84.0
VGG+SVM [50]	VGG16&19	98.9	95.0	96.8	95.4	69.7	90.4	93.5	96.0	74.2	86.6	87.8	96.0	96.3	93.1	97.2	70.0	92.1	80.3	98.1	87.0	89.7
Fev+Lv [7]	VGG16	97.9	97.0	96.6	94.6	73.6	93.9	96.5	95.5	73.7	90.3	82.8	95.4	97.7	95.9	98.6	77.6	88.7	78.0	98.3	89.0	90.6
HCP [5]	VGG16	98.6	97.1	98.0	95.6	75.3	94.7	95.8	97.3	73.1	90.2	80.0	97.3	96.1	94.9	96.3	78.3	94.7	76.2	97.9	91.5	90.9
RDAL [9]	VGG16	98.6	97.4	96.3	96.2	75.2	92.4	96.5	97.1	76.5	92.0	87.7	96.8	97.5	93.8	98.5	81.6	93.7	82.8	98.6	89.3	91.9
RARL [8]	VGG16	98.6	97.1	97.1	95.5	75.6	92.8	96.8	97.3	78.3	92.2	87.6	96.9	96.5	93.6	98.5	81.6	93.1	83.2	98.5	89.3	92.0
SSGRL* [15]	ResNet-101	99.5	97.1	97.6	97.8	82.6	94.8	96.7	98.1	78.0	**97.0**	85.6	97.8	98.3	96.4	98.1	**84.9**	96.5	79.8	98.4	92.8	93.4
Baseline	ResNet-101	99.0	97.9	97.2	97.6	80.2	93.6	96.0	98.0	81.8	92.0	84.6	97.5	97.2	95.3	97.9	81.8	94.6	84.1	98.2	93.6	92.9
MCAR	ResNet-101	**99.7**	**99.0**	98.5	**98.2**	**85.4**	**96.9**	**97.4**	**98.9**	**83.7**	95.5	**88.8**	**99.1**	98.2	95.1	**99.1**	84.8	**97.1**	**87.8**	98.3	**94.8**	**94.8**

TABLE III
COMPARISONS OF AP AND MAP IN % OF OUR MODEL AND STATE-OF-THE-ART METHODS ON THE PASCAL VOC 2012. ∗ INDICATES METHODS USING LARGER INPUT SIZE (576 × 576)

Methods	Backbone	aero	bike	bird	boat	bottle	bus	car	cat	chair	cow	table	dog	horse	mbike	person	plant	sheep	sofa	train	tv	mAP
VGG+SVM [50]	VGG16&19	99.0	89.1	96.0	94.1	74.1	92.2	85.3	97.9	79.9	92.0	83.7	97.5	96.5	94.7	97.1	63.7	93.6	75.2	97.4	87.8	89.3
Fev+Lv [7]	VGG16	98.4	92.8	93.4	90.7	74.9	93.2	90.2	96.1	78.2	89.8	80.6	95.7	96.1	95.3	97.5	73.1	91.2	75.4	97.0	88.2	89.4
HCP [5]	VGG16	99.1	92.8	97.4	94.4	79.9	93.6	89.8	98.2	78.2	94.9	79.8	97.8	97.0	93.8	96.4	74.3	94.7	71.9	96.7	88.6	90.5
SSGRL* [15]	ResNet-101	99.5	95.1	97.4	96.4	85.8	94.5	93.7	**98.9**	86.7	96.3	84.6	**98.9**	**98.6**	96.2	98.7	82.2	**98.2**	**84.2**	98.1	93.5	93.9
MCAR	MobileNet-v2	98.6	92.3	95.4	93.3	77.7	93.8	92.6	97.6	80.8	90.9	82.3	96.5	96.6	95.5	98.3	78.4	92.6	78.7	96.8	90.9	91.0
MCAR	ResNet-50	99.6	95.6	97.5	95.2	85.1	95.5	94.3	98.6	85.2	95.8	83.9	98.4	98.0	97.2	98.8	81.6	95.5	81.8	98.3	**93.6**	93.5
MCAR	ResNet-101	**99.6**	**97.1**	**98.3**	**96.6**	**87.0**	**95.5**	**94.4**	98.8	**87.0**	**96.9**	**85.0**	98.7	98.3	**97.3**	**99.0**	**83.8**	96.8	83.7	**98.3**	93.5	**94.3**

the baseline by a large margin even though it does not need a large number of proposals or label dependency information. This further demonstrates the effectiveness of the proposed method for large-scale multi-label image recognition.

3) PASCAL VOC 2007 and 2012: PASCAL VOC 2007 and 2012 [47] are the most widely used datasets for MLR. There are 9,963 and 22,531 images in VOC 2007 and 2012, respectively. Each image contains one or several labels, corresponding to 20 object categories. These images are divided into three parts including *train*, *val* and *test* sets. In order to fairly compare with other competitors, we follow the common setting to train our model on the *train-val* sets, and then evaluate produced models on the *test* set. VOC 2007 contains a *train-val* set of 5,011 images and a *test* set of 4,952 images. VOC 2012 consists of 11,540 images as *train-val* set and 10,991 images as the *test* set.

4) Results on VOC 2007: We first report the AP for each category and the mAP for all categories on VOC 2007 *test* set in Table II. The current state-of-the-art is SSGRL [15] which uses GCN to model label dependency to boost the performance. We can see that our method achieves the best mAP performance among all methods. It largely outperforms the SSGRL [14] by 1.4 points (94.8% vs. 93.4%) when SSGRL uses a larger input size 576 × 576. Moreover, the proposed method improves the baseline ResNet-101 model by 1.9% under the same setting such as data augmentation and hyper-parameters of optimization. Last but not least, our framework shows good performance for some difficult categories such as "bottle", "table" and "sofa". This shows that exploiting global and local vision information is very crucial for multi-label recognition.

5) Results on VOC 2012: We report the results on VOC 2012 *test* set with PASCAL VOC evaluation server in Table III. We compare state-of-the-arts with our method on several backbone networks. First, we still win the best mAP performance with a smaller input size compared to SSGRL [15] when ResNet-101 is considered as a backbone. Second, our method achieves better performance using light-weight networks, *i.e.*MobileNet-v2 and ResNet-50, than that of VGG. This implies that it may be easy to extend our method to resource-restricted devices such as mobile phones.

C. Ablation Study

In order to comprehend how MCAR works, we perform exhaustive experiments to analyze the components in MCAR. We firstly analyze the contribution of each component in our two-stream architecture and demonstrate its effectiveness. The training details are exactly the same as those described in Section IV-B. Then, the effect of the attentional maps selection criteria and learning strategy is analyzed. Next, we also present the effects of MCAR in different hyper-parameters ($topN$ and τ) appearing in the local region localization module. The experiment is conducted on VOC 2007 and MS-COCO using different backbone networks, *e.g.*MobileNet-v2, ResNet-50 and ResNet-101, and we set the input size to 256 × 256. Finally, we extensively analyze the effects of our method under different conditions such as different global pooling strategies, various input sizes, and different network architectures.

1) Contributions of Proposed Global-to-Local Framework: To explore the effectiveness of two streams, we jointly train the global and local streams in MCAR, and during the inference stage, we report the influence of using each stream in Table IV. Firstly, thanks to the joint training strategy, our MCAR significantly outperforms the baseline method even

TABLE IV
ABLATIVE STUDY OF TWO STREAMS IN MCAR WITH RESNET-101 BACKBONE AND THE INPUT SIZE OF 448 × 448

Line No.	Methods	Global	Local	VOC 2007	MS-COCO
0	Baseline	√		92.9	77.1
1	MCAR	√		93.4 ↑0.5	81.9 ↑4.8
2			√	94.2 ↑1.3	82.9 ↑5.8
3		√	√	94.8 ↑1.9	83.8 ↑6.7

TABLE V
ABLATIVE STUDY OF ATTENTIONAL MAPS SELECTION STRATEGY IN MCAR WITH RESNET-101 BACKBONE AND THE INPUT SIZE OF 448 × 448

Methods	Selection criteria	VOC 2007	MS-COCO
MCAR	$bottom4$	93.8	81.2
	$random4$	94.2	82.1
	$top4$	**94.8**	**83.8**

when the same global image is taken as input (line 1 vs. line 0). Such improvement is very intuitive because MCAR is more robust and generalized by learning on not only global image but also various scales of local regions. Secondly, we can see that using local stream alone performs better than only using global stream (line 2 vs. line 1), which is because the local stream is able to flexibly focus on the details of each object. Nonetheless, we want to emphasize that the global stream plays an important role in guiding the learning of local stream. Last but not least, it is obvious that employing both global and local streams achieves the best results (line 3). This is similar to humans perception because we usually make a final decision after our brain gathers information from different spatial locations and object scales.

2) Importances of Attentional Maps Selection: In our method, all attentional maps are firstly sorted by global stream score following a descending order and then the $topN$ attentional maps are chosen. In order to further verify the selection strategy, we conduct other criterions to see if the performance is sensitive to the score ranking. Specifically, we design two criterions to compare with our $topN$ strategy. The first one is that we still sort global stream scores but pick $bottom\ N$ feature maps. The second one is that we randomly $sample\ N$ maps among all attentional maps.

For simplicity, we test MCAR with $top4$, $random4$ and $bottom4$ local regions while using the weights trained on MCAR with $top4$ setting in Table V. We can see that the performance of MCAR using high-confidence local regions ($top4$) is significantly better than that of using low-confidence ones ($bottom4$) or random manner ($random4$). This indicates the effectiveness of the local region selection strategy based on the ranking of the global scores.

3) Single or Pair loss?: Instead of two-stream learning using a pair of parallel losses in Eq. 8, we design a simple way to train the network utilizing a single loss. Specifically, we firstly fuse global and local prediction scores with a max-wised aggregation

$$\hat{y}^i = \max\left(\hat{y}_g^i, \hat{y}_l^i\right), \tag{11}$$

TABLE VI
ABLATIVE STUDY OF LEARNING STRATEGY IN MCAR WITH RESNET-101 BACKBONE AND THE INPUT SIZE OF 448 × 448

Methods	Learning Strategy	VOC 2007	MS-COCO
Baseline	single	92.9	77.1
MCAR	single	94.4	82.6
	pair	**94.8**	**83.8**

and then train the network with a single BCE loss as the same in Eq. 8,

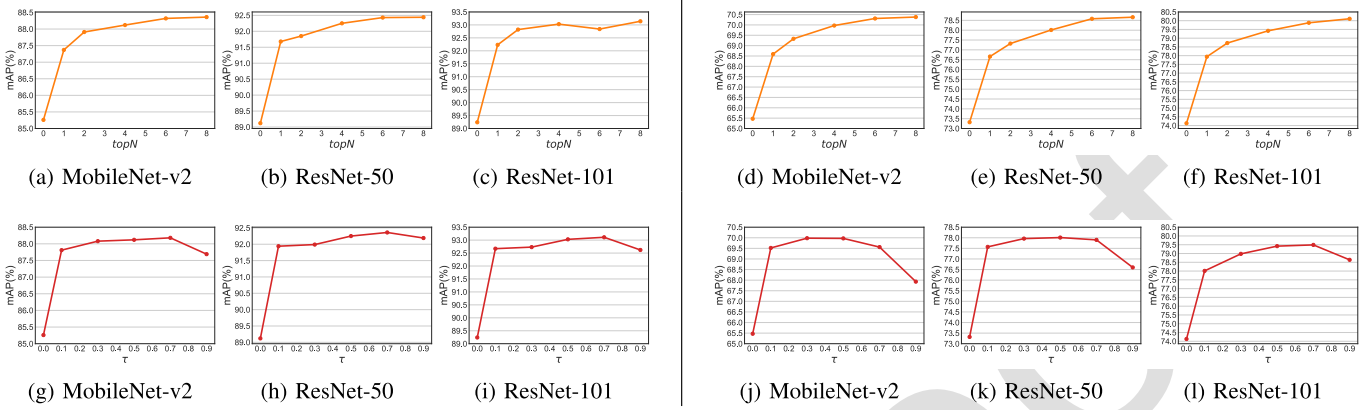
$$\mathcal{L} = \sum_{i=1}^{M}\sum_{j=1}^{C} y_i^j \log(\hat{y}_i^j) + (1 - y_i^j)\log(1 - \hat{y}_i^j) \tag{12}$$

Using ResNet-101 backbone and keeping the rest settings the same, the experimental results are reported in Table VI. We can see that MCAR with single loss obtains 94.4 mAP on VOC2007 and 82.6 mAP on MS-COCO which improves the baseline by 1.5 and 5.5 mAP, but is 0.4 and 1.2 mAP worse than our main method (with pair loss).

Why is MCAR equipped with a pair of losses better than that of a single loss? The main insight is that global visual processing usually precedes the local one. The pair loss may ensure that after the global stream has been converged fast, then it guides the local stream to find possible local regions. Indeed, we find that the single loss setting usually needs more epochs when arriving at the similar performance. This indicates that the convergence of single loss is slower than that of our pair loss.

4) Number of Local Regions: We fix τ to 0.5 and choose the value $topN$ from a given set $\{0, 1, 2, 4, 6, 8\}$. Note that, $topN = 0$ implies we train the model using global stream only, which is equal to our baseline. In the first row of Fig. 5, we show the mAP performance curves when $topN$ is set to different numbers. First, the mAP performance shows an upward trend with the number of $topN$ gradually being increased. This means that it is useful to improve the multi-label classification performance using more local regions. Second, the performance tends to be stable when $topN$ is set to 4 or 6, which implies that the improvements will be not significant when applying a large $topN$. Third, the performance of a small $topN$, (*e.g.*, 1, 2, or 4) is significantly better than that of a pure global stream (*i.e.*, $topN = 0$). This further verifies the effectiveness of the proposed selection strategy of generated high-confidence local regions. Another benefit of the region selection strategy is to help reduce the cost of computation resources.

5) Threshold of Localization: To explore the sensitivity of the τ in Eq. 7, we fix $topN$ to 4 and test different τ values from $\{0, 0.1, 0.3, 0.5, 0.7, 0.9\}$. The whole image will be considered as a local region when τ equals to 0, and it is also equivalent to the baseline method. We show the mAP performances as the function of τ in the second row of Fig. 5. First, we observe that the performance is better when τ is greater than 0. Second, the performance drops when τ is either too small or too large. We argue that if τ is too small, local regions may contain more context information and lack discriminative features because all local regions are close to the original input image. When



(a) MobileNet-v2 (b) ResNet-50 (c) ResNet-101 (d) MobileNet-v2 (e) ResNet-50 (f) ResNet-101

(g) MobileNet-v2 (h) ResNet-50 (i) ResNet-101 (j) MobileNet-v2 (k) ResNet-50 (l) ResNet-101

Fig. 5. mAP comparisons of our MCAR with different values of $topN$ and τ. The left three columns are based on PASCAL-VOC 2007 and the right three columns are based on MS-COCO dataset.

TABLE VII

COMPARISONS OF MAP IN % OF OUR METHODS AND BASELINE ON THE MS-COCO DATASET. COMPARED TO THE BASELINE METHOD, THE IMPROVEMENTS OF OUR METHOD ARE HIGHLIGHTED IN RED

Methods	MobileNet-v2		ResNet-50		ResNet-101	
Input Size	256	448	256	448	256	448
Baseline	61.5	67.8	70.1	75.4	71.2	77.1
MCAR (GAP)	66.6 ↑5.1	74.3 ↑6.5	75.9 ↑5.8	78.0 ↑2.6	77.4 ↑6.2	80.5 ↑3.4
MCAR (GWP)	69.8 ↑8.3	75.0 ↑7.2	78.0 ↑7.9	82.1 ↑6.7	79.4 ↑8.2	83.8 ↑6.7

TABLE VIII

COMPARISONS OF MAP IN % OF OUR METHODS AND BASELINE ON THE PASCAL VOC 2007 DATASET. COMPARED TO THE BASELINE METHOD, THE IMPROVEMENTS OF OUR METHOD ARE HIGHLIGHTED IN RED

Backbone	MobileNet-v2		ResNet-50		ResNet-101	
Input Size	256	448	256	448	256	448
Baseline	85.5	89.5	89.1	91.8	89.2	92.9
MCAR (GAP)	88.1 ↑2.6	91.3 ↑1.8	92.3 ↑3.2	94.1 ↑2.3	93.0 ↑3.8	94.8 ↑1.9
MCAR (GWP)	88.5 ↑3.0	91.7 ↑2.2	92.0 ↑2.9	93.7 ↑1.9	92.6 ↑3.4	94.3 ↑1.4

τ is too large, it makes local regions only contain the most discriminative parts of an object and easily leads to over-fitting. It is a good choice when the value τ is in the interval between 0.3 and 0.7.

6) Global Pooling Strategy: Encoding spatial feature descriptors to a single vector is a necessary step in state-of-the-art CNNs. The early works, *e.g.*, AlexNet and VGGNet, use a fully connected layer, and the recent ResNet usually employs global average pooling (GAP) which outputs the spatial average of each feature map. Specifically, considering class-agnostic feature map A from the top block of a backbone network. The GAP operation outputs the spatial average of the A, returning a vector $\boldsymbol{f}^a \in \mathbf{R}^{d'}$ with the k-th element being

$$f_k^a = \frac{1}{h'w'} \sum_{i=1}^{h'} \sum_{j=1}^{w'} A_{i,j,k}. \tag{13}$$

We denote the output of global maximum pooling (GMP) as $\boldsymbol{f}^m \in \mathbf{R}^{d'}$, whose the k-th element is

$$f_k^m = \max\{A_{i,j,k}\}_{i=1}^{h'}{}_{j=1}^{w'}. \tag{14}$$

The GMP easily falls into over-fitting because it enforces the network to learn the most discriminative feature. Generally, GAP usually has a better generalization ability than GMP. However, GAP may lead to under-fitting and slow convergence because it equally gives the same importance for all spatial feature descriptors. Our local region localization needs to discover the discriminative region which seems to be opposite to the objective of GAP. In order to alleviate this conflict, we propose a simple solution termed as *Global Weighted Pooling* (GWP) which is an average of $\boldsymbol{f}^a$ and $\boldsymbol{f}^m$, as

$$\boldsymbol{f} = \lambda \boldsymbol{f}^a + (1-\lambda)\boldsymbol{f}^m, \tag{15}$$

where $\lambda \in [0, 1]$ is a weight which balances the importance between GAP and GMP. In our paper, the weight λ is empirically set to 0.5.

In Table VII, we can see that MCAR with GWP further boosts performance on MS-COCO dataset. It improves the mAP by 4.1 points and 3.3 points compared to the common GAP on ResNet-50 and ResNet-101 when input size is 448×448. Nevertheless, the overall performance of GWP is

comparable to that of GAP on the PASCAL-VOC dataset as reported in Table VIII. This may be associated with a specific dataset that the task of PASCAL-VOC is relatively simpler than that of MS-COCO because of small-scale samples, fewer classes and fewer instances per image in PASCAL-VOC. Generally, MCAR equipped with GWP is better than GAP, especially on more challenging tasks.

7) Network Architecture: The recent state-of-the-art methods usually take ResNet-101 as a backbone to report their performance. However, in real applications, light-weight networks have been widely adopted. To meet such requirements, we extensively evaluate the proposed method with MobileNet-v2 and ResNet-50 besides ResNet-101 on PASCAL-VOC and MS-COCO and report their results in Tables VII and VIII. The deeper network tends to obtain better performance. This is not surprised because the big network has more parameters and a deeper structure to ensure strong capacity and transferability. Note that our method still has good performance using the lightweight MobileNet-v2. In addition, the proposed method has significant improvements for all backbones. On the MS-COCO dataset, our MCAR with GWP improves the baseline by about 7% using the input size of 448×448.

8) Input Size: The performance of multi-label recognition is sensitive to the choice of input size. Generally, the larger size tends to get the better performance as reported in Tables VII and VIII. However, it is more practicable to employ small-sized input on resource-restrict devices. Somewhat surprisingly, MCAR performs better using small inputs. In Table VII and VIII, we can see that our method always tends to produce more improvements when a smaller input size is employed. This advantage comes from the two-stream architecture which can look at an image in a comprehensive manner (global to local). This indicates that our method is more friendly for low-resolution inputs.

V. Discussion

In this section, we try to understand how the network recognizes multi-objects for a multi-label image via visualizing the produced local regions and discuss why MCAR is a simple and efficient multi-label framework.

A. Visualization

To analyze where our model focuses on an image, we show the class-specific attentional regions generated by a multi-class attentional region module in Fig. 6. It can be seen that these attentional regions cover almost all possible objects in each image which is consistent with our initial intention. Furthermore, we can find that global prediction scores of some small-scale objects are low, *e.g.* the train in (1,1), the car in (1,3), the bird in (1,5), the chair in (1,1), the cat in (2,4) and the sofa in (2,5) on the PASCAL VOC 2012 testing set and the snowboard in (3,1), the car in (3,3), the dog in (3,6), the cell phone in (4,1), the train in (4,2) and the mouse in (4,5) on the MS-COCO validation images, where (i, j) is the image at i-th row and j-th column in Fig. 6. This indicates that it is suboptimal to use global image stream solely, especially for small-scaled and partly occluded objects. This limitation would be improved by our two-stream network because it recognizes this type of object from a closer view (high score of two-stream). Compared to the baseline method, our method significantly improves the multi-label image recognition performance. Note that MCAR may produce incorrect or incomplete predictions when local regions are too small or too blurry such as the bench in (3,4), the book in (4,6) and the couch in (4,5) on MS-COCO testing images.

Furthermore, the local region stream is hardly ensured to cover all target objects even if we use a larger number ($topN$). However, the local stream is able to contain a majority of target objects because of the high diversity of local regions. Moreover, our two streams can complement each other by finding missing discriminative regions. Considering this is a weakly supervised problem and the computation efficiency, we think such this situation can be acceptable.

TABLE IX

Comparisons of Average Inference Time of per-Image Between Our MCAR (Including Each Component) and Baselines With Different Backbones and Input Sizes. The Time Is Measured in Milliseconds (ms) on One P40 GPU

Methods	ImgSize	Baseline	MCAR ($topN$=4)			
			Total	Global	G-to-L	Local
MobileNet-v2	256×256	6.7	22.7	6.6	9.2	6.9
	448×448	6.9	34.3	6.9	13.0	14.4
ResNet-50	256×256	8.5	31.7	8.2	9.1	14.3
	448×448	11.2	55.4	11.1	13.2	31.2
ResNet-101	256×256	15.7	46.8	16.1	8.6	22.1
	448×448	18.4	82.8	18.8	13.5	50.6

B. Simplicity

Our framework aims at proposing a simple and efficient method that puts forward to learn global and local image semantics in a single unified model. On one hand, we generate object proposals only using the network itself while HCP utilizes external tools such as EdgeBox [25] or BING [26]. On the one hand, our method can efficiently obtain multi-class regions with a parameter-free region localization module because of the parameter share mechanism in Eq. 1 and 4. Unlike some existing attention-based methods, they always need a slightly complex module such as LSTM unit in [33], [34] or reinforcement learning module in RARL [8].

C. Complexity

The computation complexity linearly grows with the region number (*i.e.*, $topN$). However, it is worth noting that the number of local regions has been significantly reduced by using our framework compared to region-based methods such as HCP (*e.g.*, 500). Our method works well when a small $topN$ (*e.g.*, 4) is used and thus the complexity is controllable and the computation cost is affordable. For example, the number of object proposals in HCP is 500 while we reduce this number to 4. So about 100-time speedup is obtained.

We test the forward running time of each model using the input sizes of 256×256 and 448×448. This evaluation is conducted on one P40 GPU accelerated by cuDNN v7.4.1.

Fig. 6. Selected examples of region localization and classification results on PASCAL VOC 2012 testing images (first two rows) and MS-COCO validation images (last two rows). Our MCAR achieves 94.3% mAP on the VOC 2012 testing set and 83.8% mAP on the MS-COCO validation set by using ResNet-101 backbone and the input size of 448×448. Note that these attentional regions are generated by using the model trained on image-level labels only (without bounding box annotations). Each region box is associated with a category label (c), a global stream score ($\hat{y}_g^c$) and a two-stream score ($\max\{\hat{y}_g^c, \hat{y}_l^c\}$), organized as "category name:global score/two-stream score", *e.g.*, "train: 0.14/0.99" in the image at the fourth row and second column. These region boxes are displayed with conditions on $\hat{y}_l^c > 0.1$, $\max\{\hat{y}_l^c, \hat{y}_g^c\} > 0.6$, $topN = 4$ and $\tau = 0.5$. For each image, one color represents one object category in that image. The proposed two-stream MCAR framework recognizes objects of a wide range of scales, especially for those small-sized or occluded objects, such as the car in (1,3), the bird in (1,5), the cat in (2,4) and the sofa in (2,5) on VOC 2012 testing images and the snowboard in (3,1), the car in (3,3), the dog in (3,6), the cell phone in (4,1), the train in (4,2) and the mouse in (4,5) on MS-COCO validation images, where (i, j) represents the image at i-th row and j-th column. It is noteworthy that MCAR may produce incorrect or incomplete predictions when the local region is too small or too blurry such as as the bench in (3,4), the book in (4,6) and the couch in (4,5) on the MS-COCO testing images. (Best view in color and zoom in.)

The actual inference time is reported in Table IX. We can see that the total time of our MCAR ($topN = 4$) is about 4 to 5 times compared to baselines. This is not surprising because there is at least $topN + 1$ times computation cost with our method. We also report the inference time of each component (global stream, global-to-local and local stream) of our MCAR. It can be seen that those local regions' generation and their forward inference dominate the computation cost of our MCAR, reducing the number of local regions would accelerate inference speed greatly. In addition, our MCAR significantly outperforms the baseline method (81.9% vs. 77.1% mAP on MS-COCO Table IV) even if only global image (without local regions) is taken as input, which implies our method still better than the baseline under the same inference time. Meanwhile, our method needs no additional parameters for generating local regions because of the parameter sharing mechanism between global and local streams.

VI. Conclusion

We observe that humans recognize multiple objects following two steps. In practice, they usually obey a rule of global to local. Through looking at the whole image at first, people can discover places that need to be focused with more attentions. These attentional regions are then checked closer

for a better decision. Inspired by this observation, we develop a two-stream framework to recognize multi-label images from global to local as human's perception system works. In order to localize object regions, we propose an efficient multi-class attentional region module which significantly reduces the number of regions and keeps their diversity. Our method can efficiently and effectively recognize multi-class objects with an affordable computation cost and a parameter-free region localization module. On three prevalent multi-label benchmarks, the proposed method achieves state-of-the-art results. In the future, we will try to integrate the label dependency into our method to further boost the performance. It is also an interesting direction to explore how to extend the proposed method for weakly supervised image detection and semantic segmentation.

REFERENCES

[1] K. He, X. Zhang, S. Ren, and J. Sun, "Delving deep into rectifiers: Surpassing human-level performance on ImageNet classification," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Dec. 2015, pp. 1026–1034.

[2] J. Shao, K. Kang, C. C. Loy, and X. Wang, "Deeply learned attributes for crowded scene understanding," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2015, pp. 4657–4666.

[3] Z. Liu, P. Luo, X. Wang, and X. Tang, "Deep learning face attributes in the wild," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Dec. 2015, pp. 3730–3738.

[4] Y. Li, C. Huang, C. C. Loy, and X. Tang, "Human attribute recognition by deep hierarchical contexts," in *Proc. Eur. Conf. Comput. Vis. (ECCV)*, 2016, pp. 684–700.

[5] Y. Wei *et al.*, "HCP: A flexible CNN framework for multi-label image classification," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 38, no. 9, pp. 1901–1907, Sep. 2016.

[6] M.-L. Zhang and Z.-H. Zhou, "A review on multi-label learning algorithms," *IEEE Trans. Knowl. Data Eng.*, vol. 26, no. 8, pp. 1819–1837, Aug. 2014.

[7] H. Yang, J. T. Zhou, Y. Zhang, B.-B. Gao, J. Wu, and J. Cai, "Exploit bounding box annotations for multi-label object recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 280–288.

[8] T. Chen, Z. Wang, G. Li, and L. Lin, "Recurrent attentional reinforcement learning for multi-label image recognition," in *Proc. 32nd AAAI Conf. Artif. Intell. (AAAI)*, 2018, pp. 6730–6737.

[9] 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.

[10] F. Zhu, H. Li, W. Ouyang, N. Yu, and X. Wang, "Learning spatial regularization with image-level supervisions for multi-label image classification," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 5513–5522.

[11] H. Guo, K. Zheng, X. Fan, H. Yu, and S. Wang, "Visual attention consistency under image transforms for multi-label image classification," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2019, pp. 729–739.

[12] J. Wang, Y. Yang, J. Mao, Z. Huang, C. Huang, and W. Xu, "CNN-RNN: A unified framework for multi-label image classification," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 2285–2294.

[13] S.-F. Chen, Y.-C. Chen, C.-K. Yeh, and Y.-C. F. Wang, "Order-free RNN with visual attention for multi-label classification," in *Proc. 32nd AAAI Conf. Artif. Intell. (AAAI)*, 2018, pp. 6714–6721.

[14] Z.-M. Chen, X.-S. Wei, P. Wang, and Y. Guo, "Multi-label image recognition with graph convolutional networks," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2019, pp. 5177–5186.

[15] T. Chen, M. Xu, X. Hui, H. Wu, and L. Lin, "Learning semantic-specific graph representation for multi-label image recognition," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2019, pp. 522–531.

[16] Z. Chen, X.-S. Wei, P. Wang, and Y. Guo, "Learning graph convolutional networks for multi-label recognition and applications," *IEEE Trans. Pattern Anal. Mach. Intell.*, early access, Mar. 3, 2021, doi: 10.1109/TPAMI.2021.3063496.

[17] D. Navon, "Forest before trees: The precedence of global features in visual perception," *Perception Psychophys.*, vol. 5, no. 3, pp. 197–200, 1969.

[18] J. Hegdé, "Time course of visual perception: Coarse-to-fine processing and beyond," *Prog. Neurobiol.*, vol. 84, no. 4, pp. 405–439, Apr. 2008.

[19] A. V. Flevaris, A. Martínez, and S. A. Hillyard, "Attending to global versus local stimulus features modulates neural processing of low versus high spatial frequencies: An analysis with event-related brain potentials," *Frontiers Psychol.*, vol. 5, Apr. 2014, p. 277.

[20] S. Lazebnik, C. Schmid, and J. Ponce, "Beyond bags of features: Spatial pyramid matching for recognizing natural scene categories," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, vol. 2, Jun. 2006, pp. 2169–2178.

[21] K. He, X. Zhang, S. Ren, and J. Sun, "Spatial pyramid pooling in deep convolutional networks for visual recognition," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 37, no. 9, pp. 1904–1916, Sep. 2015.

[22] 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. (CVPR)*, Jun. 2014, pp. 580–587.

[23] H. Zhao, J. Shi, X. Qi, X. Wang, and J. Jia, "Pyramid scene parsing network," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 2881–2890.

[24] L.-C. Chen, G. Papandreou, F. Schroff, and H. Adam, "Rethinking atrous convolution for semantic image segmentation," 2017, *arXiv:1706.05587*. [Online]. Available: http://arxiv.org/abs/1706.05587

[25] C. L. Zitnick and P. Dollár, "Edge boxes: Locating object proposals from edges," in *Proc. Eur. Conf. Comput. Vis. (ECCV)*, 2014, pp. 391–405.

[26] M.-M. Cheng, Z. Zhang, W.-Y. Lin, and P. Torr, "BING: Binarized normed gradients for objectness estimation at 300fps," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2014, pp. 3286–3293.

[27] L. Bazzani, H. Larochelle, V. Murino, J.-A. Ting, and N. D. Freitas, "Learning attentional policies for tracking and recognition in video with deep networks," in *Proc. 28th Int. Conf. Mach. Learn. (ICML)*, 2011, pp. 937–944.

[28] J. Fu, H. Zheng, and T. Mei, "Look closer to see better: Recurrent attention convolutional neural network for fine-grained image recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 4438–4446.

[29] K. Xu *et al.*, "Show, attend and tell: Neural image caption generation with visual attention," in *Proc. Int. Conf. Mach. Learn. (ICLR)*, 2015, pp. 2048–2057.

[30] P. Anderson *et al.*, "Bottom-up and top-down attention for image captioning and visual question answering," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2018, pp. 6077–6086.

[31] S. Hong, J. Oh, H. Lee, and B. Han, "Learning transferrable knowledge for semantic segmentation with deep convolutional neural network," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 3204–3212.

[32] V. Mnih *et al.*, "Recurrent models of visual attention," in *Proc. Adv. Neural Inf. Process. Syst. (NIPS)*, 2014, pp. 2204–2212.

[33] M. Jaderberg, K. Simonyan, A. Zisserman, and K. Kavukcuoglu, "Spatial transformer networks," in *Proc. Adv. Neural Inf. Process. Syst. (NIPS)*, 2015, pp. 2017–2025.

[34] W.-J. Yu, Z.-D. Chen, X. Luo, W. Liu, and X.-S. Xu, "DELTA: A deep dual-stream network for multi-label image classification," *Pattern Recognit.*, vol. 91, pp. 322–331, Jul. 2019.

[35] J. Xu, H. Tian, Z. Wang, Y. Wang, W. Kang, and F. Chen, "Joint input and output space learning for multi-label image classification," *IEEE Trans. Multimedia*, vol. 23, pp. 1696–1707, 2021.

[36] V. O. Yazici, A. Gonzalez-Garcia, A. Ramisa, B. Twardowski, and J. V. D. Weijer, "Orderless recurrent models for multi-label classification," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2020, pp. 13440–13449.

[37] H. Lai, P. Yan, X. Shu, Y. Wei, and S. Yan, "Instance-aware hashing for multi-label image retrieval," *IEEE Trans. Image Process.*, vol. 25, no. 6, pp. 2469–2479, Jun. 2016.

[38] X.-Y. Jing, F. Wu, Z. Li, R. Hu, and D. Zhang, "Multi-label dictionary learning for image annotation," *IEEE Trans. Image Process.*, vol. 25, no. 6, pp. 2712–2725, Jun. 2016.

[39] Z. Ji *et al.*, "Deep ranking for image zero-shot multi-label classification," *IEEE Trans. Image Process.*, vol. 29, pp. 6549–6560, 2020.

[40] T. Chen, L. Lin, X. Hui, R. Chen, and H. Wu, "Knowledge-guided multi-label few-shot learning for general image recognition," *IEEE Trans. Pattern Anal. Mach. Intell.*, early access, Sep. 28, 2020, doi: 10.1109/TPAMI.2020.3025814.

[41] Z.-H. Zhou, M.-L. Zhang, S.-J. Huang, and Y.-F. Li, "Multi-instance multi-label learning," *Artif. Intell.*, vol. 176, no. 1, pp. 2291–2320, 2012.
[42] K. Simonyan and A. Zisserman, "Two-stream convolutional networks for action recognition in videos," in *Proc. Adv. Neural Inf. Process. Syst. (NIPS)*, 2014, pp. 568–576.
[43] B. Zhou, A. Khosla, A. Lapedriza, A. Oliva, and A. Torralba, "Learning deep features for discriminative localization," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 2921–2929.
[44] C.-W. Lee, W. Fang, C.-K. Yeh, and Y.-C.-F. Wang, "Multi-label zero-shot learning with structured knowledge graphs," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2018, pp. 1576–1585.
[45] W. Ge, S. Yang, and Y. Yu, "Multi-evidence filtering and fusion for multi-label classification, object detection and semantic segmentation based on weakly supervised learning," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2018, pp. 1277–1286.
[46] T.-Y. Lin *et al.*, "Microsoft COCO: Common objects in context," in *Proc. Eur. Conf. Comput. Vis. (ECCV)*, 2014, pp. 740–755.
[47] M. Everingham, L. Van Gool, C. K. I. Williams, J. Winn, and A. Zisserman, "The Pascal visual object classes (VOC) challenge," *Int. J. Comput. Vis.*, vol. 88, no. 2, pp. 303–338, Jun. 2010.
[48] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 770–778.
[49] M. Sandler, A. Howard, M. Zhu, A. Zhmoginov, and L.-C. Chen, "MobileNetV2: Inverted residuals and linear bottlenecks," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2018, pp. 4510–4520.
[50] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," in *Proc. Int. Conf. Learn. Represent. (ICLR)*, 2015.

Bin-Bin Gao received the B.S. and M.S. degrees in applied mathematics in 2010 and 2013, respectively, and the Ph.D. degree in computer science from Nanjing University, China, in 2018. He is currently a Senior Researcher with the Tencent YouTu Laboratory. His research interests include computer vision and machine learning. He has severed as a Reviewer for international journals, such as the IEEE TRANSACTIONS ON IMAGE PROCESSING (TIP), the IEEE TRANSACTIONS ON NEURAL NETWORKS AND LEARNING SYSTEMS (TNNLS), the IEEE TRANSACTIONS ON INDUSTRIAL INFORMATICS (TII), the IEEE TRANSACTIONS ON KNOWLEDGE AND DATA ENGINEERING (TKDE), and *Neural Networks* and a Program Committee Member for international conferences, such as CVPR, ICCV, ECCV, and AAAI.

Hong-Yu Zhou received the B.S. degree from Wuhan University, China, in 2015, and the M.S. degree from the Department of Computer Science and Technology, Nanjing University, China, in 2018. He is currently pursuing the Ph.D. degree with the Department of Computer Science, The University of Hong Kong. His research interests include computer vision and machine learning.