PSCNet: Pyramidal Scale and Global Context Guided Network for Crowd Counting

Guangshuai Gao, Qingjie Liu, *Member, IEEE*, Qi Wen, Yunhong Wang, *Fellow, IEEE*

Abstract—Crowd counting, which towards to accurately count the number of the objects in images, has been attracted more and more attention by researchers recently. However, challenges from severely occlusion, large scale variation, complex background interference and non-uniform density distribution, limit the crowd number estimation accuracy. To mitigate above issues, this paper proposes a novel crowd counting approach based on pyramidal scale module (PSM) and global context module (GCM), dubbed PSCNet. Moreover, a reliable supervision manner combined Bayesian and counting loss (BCL) is utilized to learn the density probability and then computes the count exception at each annotation point. Specifically, PSM is used to adaptively capture multi-scale information, which can identify a fine boundary of crowds with different image scales. GCM is devised with low-complexity and lightweight manner, to make the interactive information across the channels of the feature maps more efficient, meanwhile guide the model to select more suitable scales generated from PSM. Furthermore, BL is leveraged to construct a credible density contribution probability supervision manner, which relieves non-uniform density distribution in crowds to a certain extent. Extensive experiments on four crowd counting datasets show the effectiveness and superiority of the proposed model. Additionally, some experiments extended on a remote sensing object counting (RSOC) dataset further validate the generalization ability of the model. Our resource code will be released upon the acceptance of this work.

Index Terms—Crowd Counting, pyramidal scale, global context, Bayesian loss

I. INTRODUCTION

CROWD counting, which is to estimate the accurate number of the objects in images or videos, has been attracted remarkable interest in recent years, and applied widely to traffic monitor [1], urban planning [2] and public safety [3], etc. Additionally, methods for crowd counting also can be extended to other domains including cell microscopy [4]–[6], animals [7], vehicles [8]–[11], leaves [12], [13], environment survey [14], [15] and remote sensing applications [16], [17], to name a few.

Recent prevalent crowd counting methods are following the pioneering work [6], which estimated the count number over a density map. Lately, with the powerful feature representation ability of Convolutional Neural Networks (CNNs), a lot of derivative CNN-based density estimation algorithms have been

Guangshuai Gao, Qingjie Liu and Yunhong Wang are with the State Key Laboratory of Virtual Reality Technology and Systems, Beihang University, Xueyuan Road, Haidian District, Beijing, 100191, China and Hangzhou Innovation Institute, Beihang University, Hangzhou, 310051, China. (Email: gaoguangshuai1990@buaa.edu.cn; qingjie.liu@buaa.edu.cn;yhwang@buaa.edu.cn.)

Qi Wen is with National Disaster Reduction Center of China, Beijing 100124, China (Email:whistlewen@aliyun.com).

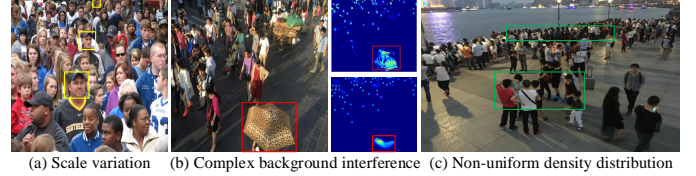(a) Scale variation   (b) Complex background interference   (c) Non-uniform density distribution

Fig. 1: Illustrations of large scale variation, complex background interference and non-uniform density distribution. In (a), person's head scales enclosed by the yellow boxes vary largely with the distance from the camera. In (b), the estimated density map always mistakenly predict the spot on the umbrella as people. In (c), the densities of different local regions are non-uniformly distributed.

presented. Although remarkable progress has been achieved, there still exist some challenges limiting the upper bound of their performance, such as large scale variation, complex background interference, non-uniform density distribution and countless others. To be specific, in Fig. 1(a), the sizes of human heads vary largely as the distance from the camera. In Fig. 1(b), background interferences (such as the spots on the umbrella), are easy to confuse the model making wrong predictions (as depicted in the red boxes in the density maps). Compared with the density map derived by a multi-column CNN (MCNN) model in [18] (top-right corner), result of our model is more robust to the interferences (bottom-right corner). Additionally, in Fig. 1(c), the density distribution is non-uniform in different local regions even in the same crowd scene.

Many efforts tackle the scale variation problem by designing multi-column architectures [18]–[21] or techniques such as dilated convolution [22], Spatial Pyramid Pooling (SPP) [23], Atrous Spatial Pyramid Pooling (ASPP) [24] and Inception blocks [25] to capture multi-scale information [26]–[29]. These models can deal with scale variation to a certain extent, but is inevasible to bring some drawbacks of them. 1) Multi-column architectures or Inception blocks adopt multiple branches with different kernel sizes, which introduce redundant and bloated structure [26]. In addition, different kernel sizes result in large number of parameters and huge computation burden [30]. 2) SPP assigns equal weights on each location of the feature map, and the pooling operation may lead to fine-detail information loss, which may degenerate the performance. 3) Dilated/Atrous convolution enlarges the receptive filed without increasing additional parameters and ASPP deploys multi-branch with different dilation rates. However, hand-crafted dilation rates cannot well match the range of scale variations. Besides, neighbor information will be lost due to the sparse sampling operation, and a larger

dilation rate may result in gridding artifacts [30]. Motivated by [31], we incorporate a pyramidal scale module (PSM) into our crowd counting framework, to effectively capture multi-scale information.

For the complex background interference, most works consider to incorporate the visual attention mechanism into their models to extract more informative features rather than all available information. Due to its prominent ability, this mechanism has been applied to many computer vision tasks such as semantic segmentation [32], visual pose estimation [33] and saliency detection [34], which is also suitable for the task of crowd counting. Recently, many outstanding crowd counting frameworks [35]–[39] have been presented with attention mechanism incorporated to address the issue of complex background interference. Although these works have achieved better performance, they always develop sophisticated attention modules, which have large computational cost and require many parameters. Although some lightweight and effective attention modules such as Squeeze-and-Excitation networks (SENet) [40] and its following works [41]–[44] developed, the fully connected (FC) layers still suffer from highly complexity. Meanwhile, the dimensionality reduction is inefficient and unnecessary to capture the interactive information across the channels [45]. Inspired by [45], [46], we provide an effective and efficient global context module (GCM), essentially a channel attention operation, to select more informative channels to make the objects more distinct from the backgrounds.

Current crowd counting datasets usually adopt point-annotations to represent the objects, and then convert them into a ground-truth density map with a Gaussian convolution operation. However, such a pixel-independent based "ground-truth" density map generation manner may be imperfect. As an alternation, Ma et al. [47] propose a Bayesian Loss (BL), which provides a more reliable supervision manner through learning the count expectation from the point annotations. In other words, the pixel values of the density map generated by this BL manner are closer to the ground-truth values than previous loss functions (i.e., mean square error (MSE) loss). This effective supervision manner can alleviate the problem of non-uniform density distribution. However, naive BL is applied to process the single scale density maps and assumes that scales of the instances are same. Besides, it is indeed a density map estimation based, which may exist an inconsistency between training phase (point-to-point loss) and the testing stage (the difference between the overall summation of estimated density maps and ground truth counts). Therefore, apart from Bayesian loss, we add a counting loss to mitigate this issue.

With the above analysis, we propose a **P**yramidal **S**cale and global **C**ontext **Net**work (**PSCNet**) for crowd counting, which is mainly used to tackle the aforementioned limitations. Specifically, for large scale variation problem, a pyramidal scale module is integrated to capture multi-scale information. For complex background interference, an effective and efficient global context module is incorporated to make the informative regions obtain more attention. Moreover, a supervision manner combined Bayesian and counting loss (BCL) is leveraged to obtain reliable on the count expectation at each annotation

point. Experiments on four crowd counting datasets and one remote sensing object counting (RSOC) dataset are developed to show the effectiveness of the proposed method.

In summary, the contributions in this paper are three folds:

- A novel crowd counting framework termed PSCNet is presented, which incorporates two efficient and lightweight modules to effectively tackle large scale variation and complex background interference. Then, a loss function combine Bayesian learning and counting property is employed to further improve the supervised training performance of the network.
- A lightweight and flexible pyramid scale module is designed to effectively extract multi-scale feature information of the crowd scenes with different image scales.
- A lightweight global context module is incorporated to make the interaction information across channels of feature map richer, and meanwhile, to guide the model to select more suitable scales.
- Extensive experiments conducted on four crowd counting datasets demonstrate the effectiveness and superiority of the proposed approach, and the extension to one remote sensing object counting dataset further reveals the generalization ability and robustness.

The remainder of this paper is organized as follows. The related work of crowd counting algorithms especially for addressing scale variation and complex background interference are briefly surveyed in Section II. The details of our proposed method is introduced in Section III, followed by experimental results and analysis are presented in Section IV. Finally, the conclusion is concluded in Section V.

II. RELATED WORK

As a hot topic in computer vision, crowd counting has been extensively studied in recent years. Existing methods range from detection based, regression-based to recently prevalent CNN-based density estimation based, interested readers can refer to recent surveys [48], [49]. In this section, we only review some representative works related to this paper, specifically for addressing large scale variation and complex background interference.

Methods for large scale variation. The primary challenge of crowd counting is the large scale variation, which is usually tackled by capturing multi-scale features with customized multi-column network architectures or with the help of effective techniques such as dilated convolution, Spatial Pyramid Pooling (SPP) and Inception module. MCNN [18] is a simple and direct multi-column structure, which adopts multiple parallel branches with different filter kernel sizes corresponding to different receptive fields. CMTL [50] incorporates high-level features within crowd classification and density estimation into a cascaded framework. Switching-CNN [51] utilizes a specialized classifier to select suitable regressors for image patches. CP-CNN [19] incorporates local and global contextual information in a unified contextual pyramid CNN to generate high fidelity density map. McML [21] utilizes mutual information with a statistical network to measure the scale correlation between features from different columns. In another
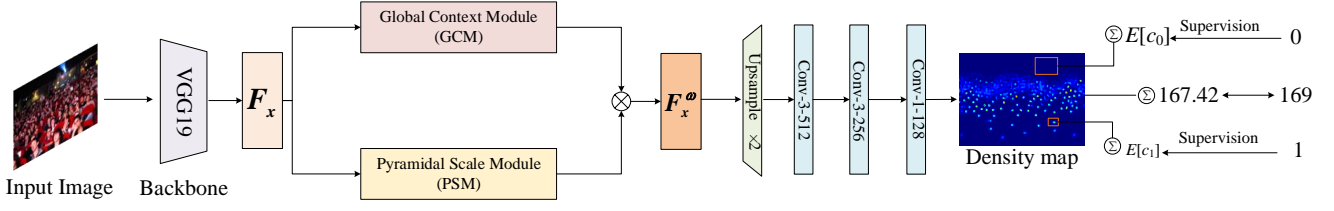
Fig. 2: The architecture of PSCNet for crowd counting. The parameters of the convolution layers are denoted as "Conv-(kernel_size)-(number of filters)". "⊗" indicates the element multiplication operation.

way, CSRNet [26] takes advantage of dilation convolution to enlarge the receptive fields of feature maps without increasing computation cost. SPN [27] utilizes a scale pyramid structure with different dilation rates for the same convolution layer. CAN [28] combines scale-aware and context-aware feature information to boost the counting performance. SANet [29] captures multi-scale features built on the shoulder of Inception module [25]. DSNet [52] cascades multiple dense dilated convolution blocks and link them with dense residual connections. SCNet [53] leverages nested dilation convolutional layer, which incorporates kernels with different dilation rates, and SPP layer with different stride. ADSCNet [54] adopts adaptive dilated convolution to learn dynamic and continuous dilated rates for each pixel location. Albeit great progress has been achieved by these models, bloated structure in the multi-column networks may lead to excessive information redundancy and huge computation burden. Additionally, the discrete receptive fields of dilated convolution limit the adaptive ability across the range of large scale changes.

Methods for complex background interference. Attention mechanism, abstracting more informative features from feature maps, performs strong and extraordinary capability in various computer vision tasks. It is often incorporated into the crowd counting frameworks to address the problem of complex background interference. Among the standout crowd counting methods with attention mechanism, SAANet [35] learns a set of gate attention masks in the intermediate convolution layers. ADCrowdNet [36] combines visual attention and deformable convolution [55] into a unified framework. HA-CNN [37] designs a hierarchical attention based network to selectively enhance the features at various levels. RANet [38] and ANF [39] incorporate self-attention to capture long-range dependencies of the feature maps. SDANet [56] builds a dense attention network based on shallow features. ASNet [57] learns attention scaling factors and automatically adjust the density regions by multiple density attention masks on them. These methods have gained significant performance, nevertheless, the sophisticated structures of the attention modules incorporated in these models always suffer from huge computation burden and large number of parameters. Although some lightweight attention modules such as Squeeze-and-Excitation networks (SENet) [40] and convolution block attention module (CBAM) [41] are developed to alleviate this problem, the fully connected (FC) layers still have many parameters. In addition, the channel dimensionality reduction in these models limits the upper bound of the performance.

Different from previous methods, our proposed PSCNet

takes advantage of a pyramidal scale module to capture multi-scale features, which can flexibly cover various scales and enlarge the receptive field without increasing any computation cost. Additionally, we devise an effective global context module, essentially a light-weight channel attention operation. It can not only reduce the computation burden and parameters in previous attention modules, but also makes the cross-channel interaction more efficient by avoiding dimensionality reduction. Finally, we train our model with a reliable supervision manner on the count expectation at each annotation point.

III. OUR PROPOSED METHOD

A. PSCNet Overview

The architecture of PSCNet is illustrated in Fig. 2, which mainly consists of four stages including a backbone network as feature extractor, a pyramidal scale module to capture multi-scale information, and an effective global context module for complex background interference, meanwhile multi-scale features selection generated from the pyramidal scale module. Then an upsample layer is adopted with bilinear interpolation operation. Finally, three successive convolutional layers are used, two of which are 3×3 with 256 and 128 channels, and one 1×1 to generate the final estimated density map.

Specifically, we adopt the same backbone network as [47], i.e., VGG19 [58] with the final three fully connected layers and one pooling layer removed. Hence, the output of the backbone is 1/16 times of resolution relative to the original input image. Then we upsample the feature maps by 2 times with bilinear interpolation. Afterwards, to deal with the scale variation problem, a pyramidal scale module is introduced to efficiently capture multi-scale information around crowd scenes. Furthermore, an effective global context module (GCM) is leveraged to alleviate the complex background inference, and select more suitable channels of the feature maps. Finally, a reliable supervision manner, i.e., Bayesian Loss [47] combined with counting loss (BCL) is adopted, which constructs a density contribution probability model by computing the count expectation at each annotation point. This operation can alleviate the non-uniform density distribution to a certain extent, and guarantee reliable and accurate counting numbers for the final regression.

B. Pyramidal Scale Module (PSM)

For the scale variation problem in crowd counting, motivated by PyConv module in [31], we introduce a pyramidal scale module (PSM) to capture multi-scale information from

local and global perspective. PSM deploys increasing types of kernels from bottom to up in a pyramidal manner, and decreasing kernel depths (connectivity) with grouped convolution. The double-oriented pyramid operation allows the model capture richer multi-scale information, from larger receptive fields of kernels with lower connectivity to smaller receptive fields with higher connectivity. This operation is efficient, flexible and computation cost economical, which has achieved promising performance in various application including image classification, semantic segmentation and object detection [31]. To the best of our knowledge, it is the first applied in the field of crowd counting.
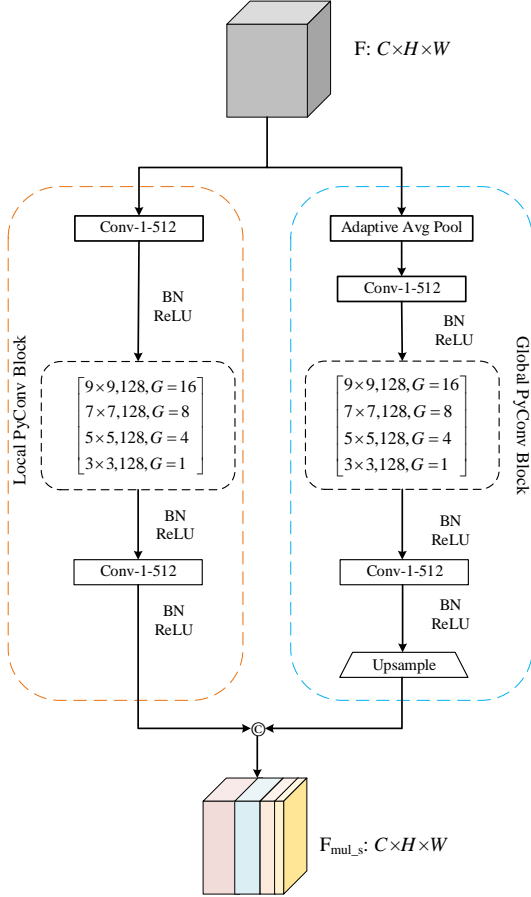

Fig. 3: The detail architecture and parameters setting of PSM. "©" indicates the concatenation operation.

The PSM used in our work is illustrated in Fig. 3, from which we can observe that the module mainly comprises of two parts: a local PyConv block and a global PyConv block. Local PyConv block is to capture multi-scale features at the local view, which is responsible for the smaller objects. It firstly applies a 1×1 convolution to make the channel of the given feature map to 512, and then an aggregation of four layers with different kernel sizes (i.e., 9×9, 7×7, 5×5, and 3×3) is adopted. Besides, the number of groups (G) enables the kernels to have different connectivity. Finally, a 1×1 convolution is used to fuse the information from different kernel sizes and depths. Noting that each convolution block is followed by a batch normalization [59] and ReLU [60] activation layer.

Global PyConv block is to capture larger objects in global view, which uses the same structure as the local Pyconv block. But the difference between them is that, global PyConv block uses an adaptive average pooling operation at the beginning to reduce the spatial size of the feature maps to 9×9, and adopts an upsample operation to resize the feature maps to the same resolution with the input through bilinear interpolation at the end.

Finally, the features extracted from both local and global PyConv blocks are concatenated. Thus we can obtain multi-scale features without adding computation burden, which can largely boost the robustness of the model to scale variation.

C. Global Context Module (GCM)

Visual attention has been claimed as a promising solution for overcoming the interference of complex background, and some previous works also use it to improve crowd counting performance [35], [37]–[39], [57]. These models have achieved higher accuracy, however, they usually leverage self-attention [61] or non-local modules [62], which always bring higher model complexity and suffer from heavier computational burden. Albeit some lightweight and effective attention models such as SENet [40] and CBAM [41] are presented, the FC layers in them still have many parameters. In addition, dimensionality reduction brings side effect on channel attention prediction, it is unnecessary and inefficient to capture the relationships between the channels [45]. Drawing the inspiration from [45] and [46], we propose an efficient global context module , essentially a channel attention operation, which can avoid dimensionality reduction and model the dependencies across the channels. The global context module designed in our work is depicted in Fig. 4.

Concretely, given an intermediate feature map, denoted as $x \in \mathbb{R}^{C \times H \times W}$, where C, H and W represent the number of channels, height and width of the feature map, respectively. Assume that x_c be the feature map corresponding to each channel, i.e., $x_c = \left[x_c^{i,j}\right]_{H \times W} \in \mathbb{R}^{H \times W}, c \in \{1, 2, \cdots, C\}$. Larger receptive fields would capture more information than smaller ones to avoid local ambiguities [43]. Therefore, a global context module is embedded to capture global context information of each channel. The module is formulated as:

$$s_c = \alpha_c \left\| x_c \right\|_2 = \alpha_c \left\{ \left[\sum_{i=1}^{H} \sum_{j=1}^{W} \left(x_c^{i,j} \right)^2 \right] + \epsilon \right\}^{\frac{1}{2}} \quad (1)$$

where α_c denotes the embedding weight, and ϵ is a small constant to avoid the deviation at zero point. This global context module is somewhat similar as global average pooling (GAP) but more robust than it [46].

Generally, to reduce the parameters overhead, some attention models such as SENet [40] and CBAM [41], always reduce the dimension of the channels. Although this way makes the model complexity lower, it destroys the direct correspondence between channel and the weight [45]. Therefore, to learn cross-channel interaction effectively, we adopt the strategy in [45], which first adaptively determines the
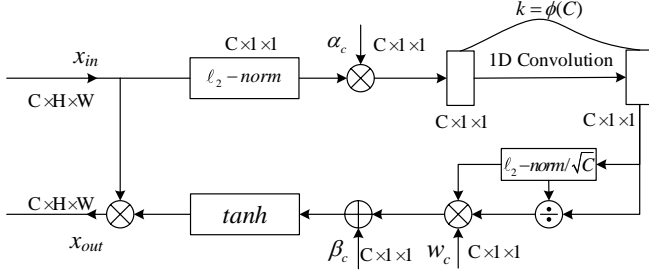
Fig. 4: The illustration of global context module.

kernel sizes k ($k = 3$ in this paper), and then performs a 1D convolution operation, i.e.,

$$\hat{s}_c = C1D(s_c) \qquad (2)$$

where $C1D$ means $1D$ convolution.

In addition, normalization methods can make the relationships between neurons (or channels) more competitive [63]. Therefore, a channel normalization method is necessary to adopt. Define $\mathbf{s} = \begin{bmatrix} \hat{s}_1, \hat{s}_2, ..., \hat{s}_C \end{bmatrix}$, the channel normalization can be formulated as:

$$\tilde{s}_c = \frac{\sqrt{C}\hat{s}_c}{\|\mathbf{s}\|_2} = \frac{\sqrt{C}\hat{s}_c}{\sqrt{\sum_{c=1}^{C} \hat{s}_c^2 + \epsilon}} \qquad (3)$$

Eventually, the final global context attention map $\tilde{x}_c^{att} \in \mathbb{R}^{C \times 1 \times 1}$ can be obtained by a $tanh$ activation layer:

$$\tilde{x}_c^{att} = \tanh(w_c \tilde{s}_c + \beta_c) \qquad (4)$$

where w_c and β_c represent the trainable weight and bias, which are both initialized to 0 in the training stage.

We multiply the obtained attention map with the multi-scale features generated from PSM, which can guide PSM select more suitable scales information meanwhile highlight the informative object regions and alleviate the complex background interference in crowd scenes.

D. Bayesian and counting loss function (BCL)

Current available crowd counting datasets mostly provide point annotations at the center of heads in each training image, and then convert them to a ground truth density map by using a Gaussian convolution operation. Then the overall network os trained with Euclidean loss to measure the distance between the estimated density map and ground truth [18], [26]. Such a pixel-independent ground truth density map generation manner may be suboptimal due to severe occlusion, large scale variation, complex background interference and non-uniform density distribution. To tackle this, Ma et al. [47] propose a more reliable and stronger supervision manner, Bayesian Loss (BL). It constructs a density contribution probability model from point annotations, which trains the loss by computing the difference between the count expectation and the ground truth crowd number (i.e., one) at each annotated point. The Bayesian loss function is defined as follows:

$$\mathcal{L}^{Bayesian} = \sum_{n=1}^{N} \mathcal{F}(1 - E[c_n]) + \mathcal{F}(0 - E[c_0]) \qquad (5)$$

where N is the total crowd number, $E[c_n]$ and $E[c_0]$ indicate the expected counts for each person and the entire background, respectively. The first term denotes that impelling the foreground count at each annotation point equal to 1, while the second term means that enforcing the background count to be zero. $\mathcal{F}(\cdot)$ is a distance function, we adopt ℓ_1 distance metric as suggested in [47].

Although Bayesian loss [47] is more reliable and effective than traditional Euclidean loss, it is indeed a density map estimation based, which may exist an inconsistency between training phase (point-to-point loss) and the testing stage (the difference between the overall summation of estimated density maps and ground truth counts). Therefore, apart from Bayesian loss, we add a counting loss to mitigate this issue. The counting loss is defined as:

$$\mathcal{L}^{Count} = \frac{1}{N} \sum_{i=1}^{N} \|F(X_i; \Theta) - Y_i\|_1 \qquad (6)$$

where $F(X_i; \Theta)$ and Y_i represent the count integrated by the estimated density map and ground truth count of the ith image. Θ denotes training parameters and $\|\cdot\|_1$ means ℓ_1-norm.

Therefore, the overall loss function is the combination of Bayesian loss $\mathcal{L}^{textBayesian}$ and counting loss $\mathcal{L}^{Count}$:

$$\mathcal{L}^{Overall} = \mathcal{L}^{Bayesian} + \lambda \mathcal{L}^{Count} \qquad (7)$$

where λ indicates trade-off weight, which is a tunable positive hyper parameter.

IV. EXPERIMENT RESULTS AND DISCUSSION

In this section, the datasets, evaluation protocols and implementation details are first introduced. Then ablation studies and comparison experiments with state-of-the-art methods are provided to demonstrate the effectiveness and superiority of the proposed approach. Furthermore, some extension experiments to other object counting applications are conducted to validate the generalization ability and robustness of the model.

A. Datasets and evaluation protocol

Datasets: Extensive experiments are conducted on four widely used crowd counting datasets, i.e., ShanghaiTech [18] Part_A and Part_B, as well as two very dense datasets, UCF-QNRF [64] and UCF_CC_50 [65]. Moreover, to further validate the generalization ability and robustness of the model, we also conduct the experiments on a remote sensing object counting (RSOC) [16], [17] dataset. The statistics of the used object counting datasets are presented in Table I.

• **ShanghaiTech** [18] includes two parts, i.e., Part_A and Part_B, with a total number of 1,198 images. The images of Part_A are randomly crawled from Internet, which are across diverse scenes and largely varied densities. Part_A has 482 images, of which 300 are served as training set and the remaining 182 for testing. The images of Part_B are taken from the metropolis in Shanghai, which consists of 400 images for training and 316 for testing.

• **UCF-QNRF** [64] is a recently released large and challenging dataset, which has a wide range of image resolutions, counts,

TABLE I: Statistics of the object counting datasets, including four crowd counting datasets and one remote sensing object counting (RSOC) dataset. Total-, min-, average- and max represent the total number, the minimum, average number and maximum number of instances in the datasets, respectively.

Dataset	Attributes	Number of Images	Training/Test	Average Resolution	Count Statistics			
					Total	Min	Average	Max
SHT_A [18]	Crowd	482	300/182	589×868	241,677	33	501.4	3,139
SHT_B [18]	Crowd	716	400/316	768×1024	88,488	9	123.6	578
UCF-QNRF [64]	Crowd	1,535	1201/334	2013×2902	1,251,642	49	815	12,865
UCF_CC_50 [65]	Crowd	50	–	2101×2888	63,974	94	1,280	4,543
RSOC_building [16]	Remote sensing	2468	1205/1263	512×512	76,215	15	30.88	142
RSOC_small-vehicle [16]	Remote sensing	280	222/58	2473×2339	148,838	17	531.56	8531
RSOC_large-vehicle [16]	Remote sensing	172	108/64	1552×1573	16,594	12	96.48	1336
RSOC_ship [16]	Remote sensing	137	97/40	2558×2668	44,892	50	327.68	1661

scale variations and diversely density distribution. The images of this dataset are crawled from Flickr, Web Search and Hajj footage, containing 1,535 images with over 125 million point annotations, where 1,201 images are used for training and the remaining 334 images for testing.

• **UCF_CC_50** [65] is composed of 50 images with various resolutions. The dataset is small-scale yet challenging since the average count is up to 1,280. Following [65], five-fold cross validations are preformed to obtained the final test result.

• **RSOC** [16], [17] is a remote sensing object counting dataset, which has four categories, including buildings, small vehicles, large vehicles and ships. The dataset consists of 3,057 images with 286,539 annotations in total, and images are collected from GoogleEarth platform (*resp.* building subset) and one off-the-shell large-scale Dataset for Object deTection in Aerial images (DOTA) [66] (*resp.* the other three subsets). In which 2,468 building images, 1,205 and 1,263 are used for training and testing; 280 small vehicle images, 222 images for training and 58 for testing; 172 large vehicle images, 108 for training and 64 for testing; 137 ship images, 97 images for training and 40 images for testing, respectively.

Evaluation protocol: The most widely used evaluation metrics including Mean Average Error (MAE) and Root Mean Squared Error (RMSE), are employed to evaluate the performance of the proposed method. The two metrics are defined as follows:

$$MAE = \frac{1}{K} \sum_{i=1}^{K} \left| \hat{C}_i - C_i \right| \tag{8}$$

$$RMSE = \sqrt{\frac{1}{K} \sum_{i=1}^{K} \left| \hat{C}_i - C_i \right|^2} \tag{9}$$

where K is the number of test images, $\hat{C}_i$ denotes the predicted count and C_i indicates the ground truth count for the i-th image, respectively.

B. Implementation details

We implement our proposed PSCNet in PyTorch [67] with an ene-to-end manner. All the experiments are implemented on an NVIDIA 2080Ti GPU. A truncated VGG19 [58] pre-trained on ImageNet [63] is taken as backbone, with the fully connected layers and the last pooling layer are removed. During training, the initial learning rate is 1e-5 and Adam optimizer [68] is used. For better training

TABLE II: Different settings on UCF_QNRF crowd counting dataset.

Baseline	PSM	GCM	MAE	RMSE
✓	–	–	98.62	169.73
✓	✓	–	89.06	155.25
✓	–	✓	91.34	158.81
✓	✓	✓	86.31	149.57

and avoiding overfitting, random crop and horizontal flipping are applied for augmentation. Specifically, the crop size is 256×256 for ShanghaiTech Part_A, UCF_CC_50 and RSOC_building datasets with smaller sizes images, and 512×512 for shanghaiTech Part_B, UCF_QNRF, RSOC_small-vehicle, RSOC_large-vehicle and RSOC_ship, since they have large image resolutions which make regular CNNs unable to deal with various scale variation, therefore, we limit the shorter side of each image within 2048 pixels for them. In addition, for all the datasets, 10% images are random sampled for validation from each training set. The batch size is set 1 for all the datasets.

C. Ablation studies

To validate the effectiveness of different setting of our proposed approach, we make some ablation studies on the challenging dataset, UCF_QNRF. The baseline method we adopt in this paper is BL-based model [47], which is trained with the pretrained models provided by the authors at https://github.com/ZhihengCV/Bayesian-Crowd-Counting. The results are somewhat different from, slightly worse than the results reported in [47]. The specific setting are shown in Table II.

• **Effectiveness of PSM.** We demonstrate the effectiveness of the pyramidal scale module in this subsection. From Table II we can observe that when PSM is introduced, the performance can achieve significant improvement. Specifically, there are relative improvements of 9.69% and 8.53% w.r.t MAE and RMSE, which means the robustness of the proposed PSM to the problem of large scale variation.

• **Effectiveness of GCM.** In order to validate the robustness of the model to the complex background interference, we adopt a global context module. From Table II we can find that the GCM can boost the baseline method with a considerable elevation. In particular, the performance will gain by 7.38% and 6.43% w.r.t. MAE and RMSE, which clearly proves that it
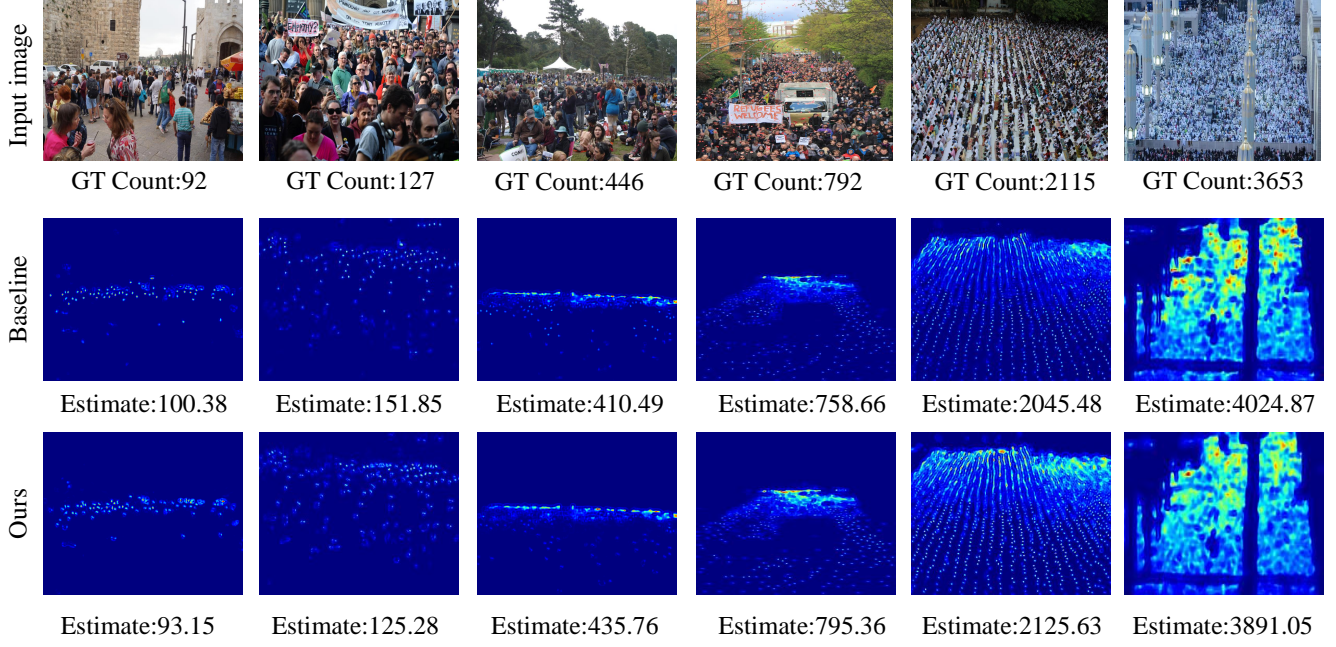
Fig. 5: Density maps generated by Baseline (the middle row) and our method (the last row). The ground truth and estimated count are put at the bottom of each image. Compared with the baseline, our proposed model can obtain more accurate estimation from sparse to highly congested scenes.

TABLE III: Different backbones on UCF_QNRF crowd counting dataset.

Backbone	MAE	RMSE
Baseline	98.62	169.73
VGG16	95.51	171.99
VGG19	86.31	149.57

TABLE IV: Different λ on UCF_QNRF crowd counting dataset.

Backbone	MAE	RMSE
Baseline	98.62	169.73
λ=0	89.36	158.64
λ=10	90.23	157.14
λ=1	87.30	155.30
λ=0.1	86.31	149.57
λ=0.01	86.98	150.93
λ=0.001	86.47	151.20

has made a significant impact on highlighting objects regions while diminishing background noise.

• **Different backbones.** Since VGG16 has been demonstrated the best backbone for crowd counting [69], thus we consider to compare VGG16 backbone with our adopted VGG19. From Table III we can observe that with the VGG16 backbone, our proposed model can still achieve relatively comparable results with the baseline method, which demonstrate strong adaptation ability. By the way, this comparison result also shows that the importance of pre-trained model selection.

• **The selection of hyper parameter** λ. To verify the influence of trade-off parameter, we set a series of different λ. As can be observed from Table IV, when $\lambda = 0.1$ we can obtain the best performance.

D. Comparisons on crowd counting datasets

1) Quantitative comparison.: Table V reports the results on four challenging crowd counting datasets, which demonstrates that our proposed approach can achieve consistent improvements compared with 30 currently reported state-of-the-art method [18]–[20], [26]–[29], [38], [39], [47], [50], [51], [56], [57], [70]–[85]. Specifically, on ShanghaiTech dataset, our proposed model increases relative improvements of 12.4%/5.9% on Part_A and 15.4%/28.1% on Part_B, w.r.t, MAE/RMSE. Even on the more dense UCF_QNRF dataset and UCF_CC_50, we can still improve the baseline with relative improvements of 12.5%/11.9% and 20.9%/14.5% w.r.t MAE/RMSE. It indicates that our proposed method performs superior performance not only for sparse but also highly congested crowd scenes.

In consideration of some methods only perform good performance on one dataset while relative poor on the other ones, for fairness, we adopt the average ranking evaluation strategy [57] to make comprehensive evaluation (denoted by avg. R. in Tabel V). The average ranking value is obtained by summing of all ranks that one method gains to divide to divide the number of datasets it utilizes. Lower value indicates higher rank. Therefore, as can be seen our proposed method obtains the best average ranking, which reveals the powerful ability to dealt with diverse crowd scenes of our proposed PSCNet model.

2) Qualitative comparison.: We visualize the estimated density maps of the proposed method compared with the baseline in Fig. 5, from which we can observe that our proposed method obtains more accurate estimations. Benefiting from the proposed PSM and GCM, our method can better reflect the

TABLE V: Comparisons of our proposed PSCNet with 30 state-of-the-art methods on four crowd counting datasets. The average ranking (denoted by avg.R.) is obtained by using the sum of all rankings that one method gains to divide the number of datasets it utilizes.

Datasets Methods	Year&Venue	ShanghaiTech_A			ShanghaiTech_B			UCF_QNRF			UCF_CC_50			avg.R.
		MAE	RMSE	R.	MAE	RMSE	R.	MAE	RMSE	R.	MAE	RMSE	R.	
Cross-scene [70]	2015 CVPR	181.8	277.7	28	32.0	49.8	27	–	–	–	467.0	498.5	29.5	28.1
MCNN [18]	2016 CVPR	110.2	173.2	27	26.4	41.3	26	277	426	19	377.6	509.1	29.5	25.3
MSCNN [71]	2017 ICIP	83.8	127.4	24.5	17.7	30.2	22	–	–	–	363.7	468.4	28	24.8
CP-CNN [19]	2017 ICCV	73.6	106.4	19	20.1	30.1	23	–	–	–	295.8	320.9	19.5	20.5
CMTL [50]	2017 AVSS	101.3	152.4	26	20.0	31.1	23.5	252	514	19.5	322.8	397.9	20.5	22.3
Switching-CNN [51]	2017 CVPR	90.4	135	25.5	21.6	33.4	25	228	445	18.5	318.1	439.2	26.5	23.8
CSRNet [26]	2018 CVPR	68.2	115	20	10.6	16	17.5	–	–	–	266.1	397.5	21.5	19.6
D-ConvNet [72]	2018 CVPR	73.5	112.3	21	18.7	26.0	21	–	–	–	288.4	404.7	24	22
IG-CNN [20]	2018 CVPR	72.5	118.2	22	13.6	21.1	19	–	–	–	291.4	349.4	21.5	20.8
ACSCP [73]	2018 CVPR	75.7	102.7	18.5	17.2	27.4	20.5	–	–	–	291.0	404.6	24	21
IC-CNN [74]	2018 ECCV	68.5	116.2	21	10.7	16.0	18	–	–	–	260.9	365.5	20	19.6
SANet [29]	2018 ECCV	67.0	104.5	16	8.4	13.6	14	–	–	–	258.4	334.9	16.5	14.8
SPN [27]	2019 WACV	61.7	99.5	8.5	9.4	14.4	16	–	–	–	259.2	335.9	17.5	14
SFCN [75]	2019 CVPR	64.8	107.5	17	7.6	13.0	9	102.0	171.4	6.5	214.2	318.2	9.5	10.5
CAN [28]	2019 CVPR	62.3	100.0	10	7.8	12.2	7.5	107	183	11.5	212.2	**243.7**	3.5	8.1
RAZNet [76]	2019 CVPR	65.1	106.7	16.5	8.4	14.1	14.5	116	195	17	–	–	–	16
TEDNet [77]	2019 CVPR	64.2	109.1	16.5	8.2	12.8	10.5	113	188	14.5	249.4	354.5	17.5	14.7
DSSINet [78]	2019 ICCV	60.6	96.0	6	6.9	10.3	3	99.1	159.2	4	216.9	302.4	7.5	5.1
RANet [38]	2019 ICCV	59.4	102.0	8.5	7.9	12.9	10	111	190	14.5	239.8	319.4	13	11.5
ANF [39]	2019 ICCV	63.9	99.4	10.5	8.3	13.2	12.5	110	174	10	250.2	340.0	17	12.5
BL [47]	2019 ICCV	62.8	101.8	10	7.7	12.7	8	88.7	154.8	2	229.3	308.2	9	7.2
LS2M [79]	2019 ICCV	64.2	98.4	10.5	7.2	11.1	4.5	104.7	173.6	8	188.4	315.3	7	7.5
S-DCNet [80]	2019 ICCV	58.3	95.0	3.5	6.7	10.7	3	104.4	176.1	8.5	204.2	301.3	5.5	5.1
MBTTBF [81]	2019 ICCV	60.2	94.1	3.5	8.0	15.5	14	97.5	165.2	4.5	233.1	300.9	8.5	7.6
DADNet [82]	2019 MM	64.2	99.9	12	8.8	13.5	14	113.2	189.4	4.5	285.5	389.7	8.5	7.6
MRL [83]	2019 MM	63.3	97.8	8.5	7.5	11.5	5.5	111.1	182.8	12.5	232.3	314.8	10	9.1
DUBNet [84]	2020 AAAI	64.6	106.8	16	7.7	12.5	7.5	105.6	180.5	10	243.8	329.3	14.5	12
HYGNN [85]	2020 AAAI	60.2	94.5	4	7.5	12.7	7	100.8	185.3	9.5	184.4	270.1	3.5	6
SDANet [56]	2020 AAAI	63.6	101.8	11	7.8	10.2	5	–	–	–	227.6	316.4	10	8.6
ASNet [57]	2020 CVPR	57.78	**90.13**	1.5	–	–	–	91.59	159.71	3.5	**174.84**	251.63	1.5	2.1
Baseline	–	64.5	101.6	–	7.8	13.5	–	98.6	169.7	–	229.3	308.2	–	–
PSCNet(Ours)	–	**56.1**	95.6	3	**6.6**	**9.7**	1	**86.3**	**149.5**	1	181.3	263.5	2.5	1.8

scale variation of the presidents. Compared with the baseline methods, our proposed model obtain more accurate estimations across diverse scenes from sparse to highly congested. Moreover,compared with baseline, our method has obtained more clearer density maps and performed stronger localization ability to a certain extent.

E. Comparisons on RSOC dataset

Beyond crowd counting, we also conduct experiments on RSOC dataset [16] to demonstrate the generalization ability and robustness of our proposed method. We compare our approach with several state-of-the-art methods and results are shown in Table VI and visualize the density maps in Fig. 6. Our model achieves substantial improvements on all the four subsets. Specifically, we improve the baseline with relative MAE improvement of 34.49%, 6.76%, 17.85% and 11.01% on buildings, small-vehicle, large-vehicle and ship subset, respectively. The improvements indicate that our proposed method has strong generalization ability.

It is worth noting that the results are reported in [16] except the baseline method. Especially, compared with AS-PDNet [16], which incorporates channel-attention, spatial attention, scale pyramid and deformable convolution into a unified framework, our proposed model still performs superior performance. The inferior results of the other comparison methods perform, we speculate that it may attribute to the down-sample all the images to fix sizes (i.e., 1024×768) in training and testing for pre-processing, which easily misleads

objects with smaller sizes. In contrast, our resizing scheme, as adopted in [47], which limits the image resolution to 2048 pixels is more effective.

V. CONCLUSION

In this paper, we have presented a novel supervised learning framework for crowd counting, called as PSCNet. Our PSCNet is characterized by three components: 1) capturing multi-scale features with an effective pyramidal scale module; 2) alleviating the interferences of complex background with a lightweight global context module incorporated, and 3) a reliable supervision manner combined with Bayesian loss and counting loss is utilized to trained the network, which learns the count expectation at each annotation point. Extensive experiments on four widely used crowd counting benchmark datasets demonstrate the effectiveness and superiority of the proposed approach. Moreover, extension experiments on a re-mote sensing object counting (RSOC) dataset further validate the generalization ability and robustness of the model.

REFERENCES

[1] D. Kang, Z. Ma, and A. B. Chan, "Beyond counting: comparisons of density maps for crowd analysis tasks—counting, detection, and tracking," *IEEE TCSVT*, vol. 29, no. 5, pp. 1408–1422, 2018. 1

TABLE VI: Performance comparison on RSOC [16] dataset.

| Datasets	RSOC_Building		RSOC_Small-vehicle		RSOC_Large-vehicle		RSOC_Ship	
Methods	MAE	RMSE	MAE	RMSE	MAE	RMSE	MAE	RMSE
MCNN [18]	13.65	16.56	488.65	1317.44	36.56	55.55	263.91	412.30
CMTL [50]	12.78	15.99	490.53	1321.11	61.02	78.25	251.17	403.07
CSRNet [26]	8.00	11.78	443.72	1252.22	34.10	46.42	240.01	394.81
SANet [29]	29.01	32.96	497.22	1276.66	62.78	79.65	302.37	436.91
SFCN [75]	8.94	12.87	440.70	1248.27	33.93	49.74	240.16	394.81
SPN [27]	7.74	11.48	445.16	1252.92	36.21	50.65	241.43	392.88
SCAR [86]	26.90	31.35	497.22	1276.65	62.78	79.64	302.37	436.92
CAN [28]	9.12	13.38	457.36	1260.39	34.56	49.63	282.69	423.44
SFANet [87]	8.18	11.75	435.29	1284.15	29.04	47.01	201.61	332.87
ASPDNet [16]	7.59	10.66	433.23	1238.61	18.76	31.06	193.83	318.95
BL [47]	11.51	15.96	168.62	280.50	13.39	35.24	84.18	136.21
PSCNet(Ours)	**7.54**	**10.52**	**157.55**	**245.31**	**11.00**	**17.65**	**74.91**	**112.11**
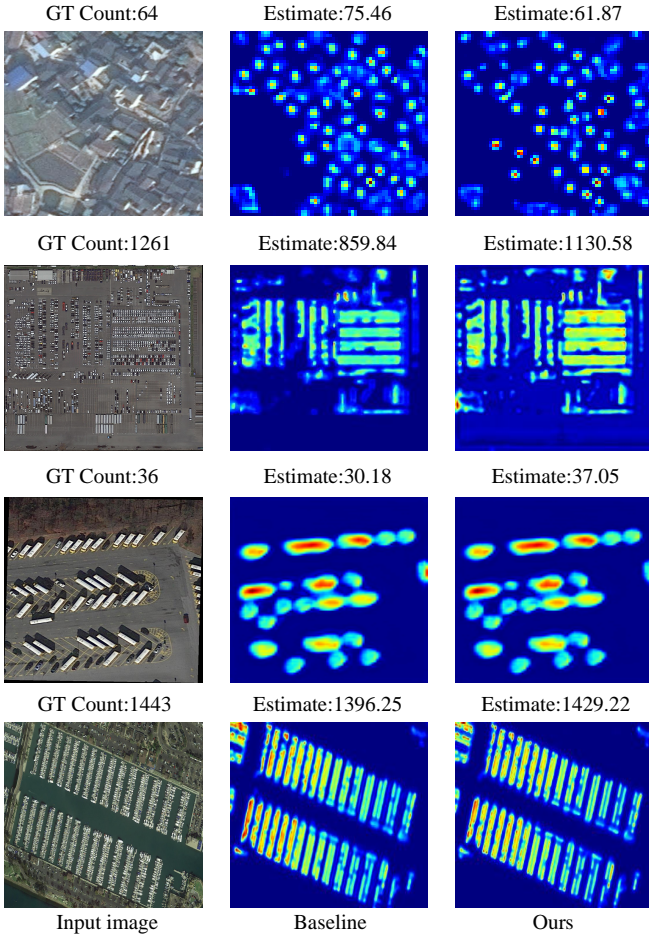
Fig. 6: Density maps generated by Baseline (the middle column) and our method (the last column). The ground truth and estimated count are put at the top of each image. Compared with the baseline, our proposed model can obtain more accurate estimation across diverse scenarios.

[2] T. Li, H. Chang, M. Wang, B. Ni, R. Hong, and S. Yan, "Crowded scene analysis: A survey," *IEEE TCSVT*, vol. 25, no. 3, pp. 367–386, 2015. 1

[3] S. Zhang, G. Wu, J. P. Costeira, and J. M. F. Moura, "Understanding traffic density from large-scale web camera data," in *CVPR*, 2017, pp. 4264–4273. 1

[4] Y. Wang and Y. Zou, "Fast visual object counting via example-based density estimation," in *ICIP*, 2016, pp. 3653–3657. 1

[5] E. Walach and L. Wolf, "Learning to count with cnn boosting," in *ECCV*, 2016, pp. 660–676. 1

[6] V. Lempitsky and A. Zisserman, "Learning to count objects in images," in *NeurIPS*, 2010, pp. 1324–1332. 1

[7] C. Arteta, V. Lempitsky, and A. Zisserman, "Counting in the wild," in *ECCV*, 2016, pp. 483–498. 1

[8] H. Zhang, Z. Kyaw, S.-F. Chang, and T.-S. Chua, "Visual translation embedding network for visual relation detection," in *CVPR*, 2017, pp. 5532–5540. 1

[9] S. Zhang, G. Wu, J. P. Costeira, and J. M. Moura, "Fcn-rlstm: Deep spatio-temporal neural networks for vehicle counting in city cameras," in *ICCV*, 2017, pp. 3667–3676. 1

[10] R. Guerrero-Gómez-Olmedo, B. Torre-Jiménez, R. López-Sastre, S. Maldonado-Bascón, and D. Onoro-Rubio, "Extremely overlapping vehicle counting," in *PRIA*, 2015, pp. 423–431. 1

[11] D. Onoro-Rubio and R. J. López-Sastre, "Towards perspective-free object counting with deep learning," in *ECCV*, 2016, pp. 615–629. 1

[12] S. Aich and I. Stavness, "Leaf counting with deep convolutional and deconvolutional networks," in *ICCV*, 2017, pp. 2080–2089. 1

[13] M. V. Giuffrida, M. Minervini, and S. A. Tsaftaris, "Learning to count leaves in rosette plants," in *CVPPP*, 2016. 1

[14] G. French, M. Fisher, M. Mackiewicz, and C. Needle, "Convolutional neural networks for counting fish in fisheries surveillance video," in *BMVCW*, 2015. 1

[15] B. Zhan, D. N. Monekosso, P. Remagnino, S. A. Velastin, and L.-Q. Xu, "Crowd analysis: a survey," *MVA*, vol. 19, no. 5-6, pp. 345–357, 2008. 1

[16] G. Gao, Q. Liu, and Y. Wang, "Counting dense objects in remote sensing images," in *ICASSP*, 2020, pp. 4137–4141. 1, 5, 6, 8, 9

[17] ——, "Counting from sky: A large-scale data set for remote sensing object counting and a benchmark method," *TGRS*, 2020. 1, 5, 6

[18] Y. Zhang, D. Zhou, S. Chen, S. Gao, and Y. Ma, "Single-image crowd counting via multi-column convolutional neural network," in *CVPR*, 2016, pp. 589–597. 1, 2, 5, 6, 7, 8, 9

[19] V. A. Sindagi and V. M. Patel, "Generating high-quality crowd density maps using contextual pyramid cnns," in *ICCV*, 2017, pp. 1861–1870. 1, 2, 7, 8

[20] D. Babu Sam, N. N. Sajjan, R. Venkatesh Babu, and M. Srinivasan, "Divide and grow: capturing huge diversity in crowd images with incrementally growing cnn," in *CVPR*, 2018, pp. 3618–3626. 1, 7, 8

[21] Z.-Q. Cheng, J.-X. Li, Q. Dai, X. Wu, J.-Y. He, and A. G. Hauptmann, "Improving the learning of multi-column convolutional neural network for crowd counting," in *ACMMM*, 2019, pp. 1897–1906. 1, 2

[22] F. Yu and V. Koltun, "Multi-scale context aggregation by dilated convolutions," in *ICLR*, 2016. 1

[23] K. He, X. Zhang, S. Ren, and J. Sun, "Spatial pyramid pooling in deep convolutional networks for visual recognition," *IEEE TPAMI*, vol. 37, no. 9, pp. 1904–1916, 2015. 1

[24] L.-C. Chen, G. Papandreou, I. Kokkinos, K. Murphy, and A. L. Yuille, "Deeplab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected crfs," *IEEE TPAMI*, vol. 40, no. 4, pp. 834–848, 2017. 1

[25] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, and A. Rabinovich, "Going deeper with convolutions," in *CVPR*, 2015, pp. 1–9. 1, 3

[26] Y. Li, X. Zhang, and D. Chen, "Csrnet: Dilated convolutional neural networks for understanding the highly congested scenes," in *CVPR*, 2018, pp. 1091–1100. 1, 3, 5, 7, 8, 9

[27] X. Chen, Y. Bin, N. Sang, and C. Gao, "Scale pyramid network for crowd counting," in *WACV*, 2019, pp. 1941–1950. 1, 3, 7, 8, 9

[28] W. Liu, M. Salzmann, and P. Fua, "Context-aware crowd counting," in *CVPR*, 2019, pp. 5099–5108. 1, 3, 7, 8, 9

[29] X. Cao, Z. Wang, Y. Zhao, and F. Su, "Scale aggregation network for accurate and efficient crowd counting," in *ECCV*, 2018, pp. 734–750. 1, 3, 7, 8, 9

[30] J. He, Z. Deng, and Y. Qiao, "Dynamic multi-scale filters for semantic segmentation," in *ICCV*, 2019, pp. 3562–3572. 1, 2

[31] I. C. Duta, L. Liu, F. Zhu, and L. Shao, "Pyramidal convolution: Rethinking convolutional neural networks for visual recognition," *arXiv preprint arXiv:2006.11538*, 2020. 2, 3, 4

[32] M. Ren and R. S. Zemel, "End-to-end instance segmentation with recurrent attention," in *CVPR*, 2017, pp. 6656–6664. 2

[33] X. Chu, W. Yang, W. Ouyang, C. Ma, A. L. Yuille, and X. Wang, "Multi-context attention for human pose estimation," in *CVPR*, 2017, pp. 1831–1840. 2

[34] L. Chen, H. Zhang, J. Xiao, L. Nie, J. Shao, W. Liu, and T.-S. Chua, "Sca-cnn: Spatial and channel-wise attention in convolutional networks for image captioning," in *CVPR*, 2017, pp. 5659–5667. 2

[35] R. R. Varior, B. Shuai, J. Tighe, and D. Modolo, "Scale-aware attention network for crowd counting," *arXiv preprint arXiv:1901.06026*, 2019. 2, 3, 4

[36] N. Liu, Y. Long, C. Zou, Q. Niu, L. Pan, and H. Wu, "Adcrowdnet: An attention-injective deformable convolutional network for crowd understanding," in *CVPR*, 2019, pp. 3225–3234. 2, 3

[37] V. A. Sindagi and V. M. Patel, "Ha-ccn: Hierarchical attention-based crowd counting network," *IEEE TIP*, vol. 29, pp. 323–335, 2019. 2, 3, 4

[38] A. Zhang, J. Shen, Z. Xiao, F. Zhu, X. Zhen, X. Cao, and L. Shao, "Relational attention network for crowd counting," in *ICCV*, 2019, pp. 6788–6797. 2, 3, 4, 7, 8

[39] A. Zhang, L. Yue, J. Shen, F. Zhu, X. Zhen, X. Cao, and L. Shao, "Attentional neural fields for crowd counting," in *ICCV*, 2019, pp. 5714–5723. 2, 3, 4, 7, 8

[40] J. Hu, L. Shen, and G. Sun, "Squeeze-and-excitation networks," in *CVPR*, 2018, pp. 7132–7141. 2, 3, 4

[41] S. Woo, J. Park, J.-Y. Lee, and I. So Kweon, "Cbam: Convolutional block attention module," in *ECCV*, 2018, pp. 3–19. 2, 3, 4

[42] Z. Gao, J. Xie, Q. Wang, and P. Li, "Global second-order pooling convolutional networks," in *CVPR*, 2019, pp. 3024–3033. 2

[43] J. Hu, L. Shen, S. Albanie, G. Sun, and A. Vedaldi, "Gather-excite: Exploiting feature context in convolutional neural networks," in *NeurIPS*, 2018, pp. 9401–9411. 2, 4

[44] X. Li, W. Wang, X. Hu, and J. Yang, "Selective kernel networks," in *CVPR*, 2019, pp. 510–519. 2

[45] Q. Wang, B. Wu, P. Zhu, P. Li, W. Zuo, and Q. Hu, "Eca-net: Efficient channel attention for deep convolutional neural networks," in *CVPR*, 2020, pp. 11 534–11 542. 2, 4

[46] Z. Yang, L. Zhu, Y. Wu, and Y. Yang, "Gated channel transformation for visual recognition," in *CVPR*, 2020, pp. 11 794–11 803. 2, 4

[47] Z. Ma, X. Wei, X. Hong, and Y. Gong, "Bayesian loss for crowd count estimation with point supervision," in *ICCV*, 2019, pp. 6142–6151. 2, 3, 5, 6, 7, 8, 9

[48] G. Gao, J. Gao, Q. Liu, Q. Wang, and Y. Wang, "Cnn-based density estimation and crowd counting: A survey," *arXiv preprint arXiv:2003.12783*, 2020. 2

[49] V. A. Sindagi and V. M. Patel, "A survey of recent advances in cnn-based single image crowd counting and density estimation," *PRL*, vol. 107, pp. 3–16, 2018. 2

[50] ——, "Cnn-based cascaded multi-task learning of high-level prior and density estimation for crowd counting," in *AVSS*, 2017, pp. 1–6. 2, 7, 8, 9

[51] D. B. Sam, S. Surya, and R. V. Babu, "Switching convolutional neural network for crowd counting," in *CVPR*, 2017, pp. 4031–4039. 2, 7, 8

[52] F. Dai, H. Liu, Y. Ma, J. Cao, Q. Zhao, and Y. Zhang, "Dense scale network for crowd counting," *arXiv preprint arXiv:1906.09707*, 2019. 3

[53] Z. Wang, Z. Xiao, K. Xie, Q. Qiu, X. Zhen, and X. Cao, "In defense of single-column networks for crowd counting," *arXiv preprint arXiv:1808.06133*, 2018. 3

[54] S. Bai, Z. He, Y. Qiao, H. Hu, W. Wu, and J. Yan, "Adaptive dilated network with self-correction supervision for counting," in *CVPR*, 2020, pp. 4594–4603. 3

[55] J. Dai, H. Qi, Y. Xiong, Y. Li, G. Zhang, H. Hu, and Y. Wei, "Deformable convolutional networks," in *ICCV*, 2017, pp. 764–773. 3

[56] Y. Miao, Z. Lin, G. Ding, and J. Han, "Shallow feature based dense attention network for crowd counting." in *AAAI*, 2020, pp. 11 765–11 772. 3, 7, 8

[57] X. Jiang, L. Zhang, M. Xu, T. Zhang, P. Lv, B. Zhou, X. Yang, and Y. Pang, "Attention scaling for crowd counting," in *CVPR*, 2020, pp. 4706–4715. 3, 4, 7, 8

[58] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," *arXiv preprint arXiv:1409.1556*, 2014. 3, 6

[59] S. Ioffe and C. Szegedy, "Batch normalization: Accelerating deep network training by reducing internal covariate shift," in *ICML*, 2015. 4

[60] V. Nair and G. E. Hinton, "Rectified linear units improve restricted boltzmann machines," in *ICML*, 2010. 4

[61] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. Kaiser, and I. Polosukhin, "Attention is all you need," in *NeurIPS*, 2017, pp. 5998–6008. 4

[62] X. Wang, R. Girshick, A. Gupta, and K. He, "Non-local neural networks," in *CVPR*, 2018, pp. 7794–7803. 4

[63] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," in *NeurIPS*, 2012, pp. 1097–1105. 5, 6

[64] H. Idrees, M. Tayyab, K. Athrey, D. Zhang, S. Al-Maadeed, N. Rajpoot, and M. Shah, "Composition loss for counting, density map estimation and localization in dense crowds," in *ECCV*, 2018, pp. 532–546. 5, 6

[65] H. Idrees, I. Saleemi, C. Seibert, and M. Shah, "Multi-source multi-scale counting in extremely dense crowd images," in *CVPR*, 2013, pp. 2547–2554. 5, 6

[66] G.-S. Xia, X. Bai, J. Ding, Z. Zhu, S. Belongie, J. Luo, M. Datcu, M. Pelillo, and L. Zhang, "Dota: A large-scale dataset for object detection in aerial images," in *CVPR*, 2018, pp. 3974–3983. 6

[67] A. Paszke, S. Gross, F. Massa, A. Lerer, J. Bradbury, G. Chanan, T. Killeen, Z. Lin, N. Gimelshein, L. Antiga *et al.*, "Pytorch: An imperative style, high-performance deep learning library," in *NeurIPS*, 2019, pp. 8024–8035. 6

[68] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," *arXiv preprint arXiv:1412.6980*, 2014. 6

[69] V. K. Valloli and K. Mehta, "W-net: Reinforced u-net for density map estimation," *arXiv preprint arXiv:1903.11249*, 2019. 7

[70] C. Zhang, H. Li, X. Wang, and X. Yang, "Cross-scene crowd counting via deep convolutional neural networks," in *CVPR*, 2015, pp. 833–841. 7, 8

[71] L. Zeng, X. Xu, B. Cai, S. Qiu, and T. Zhang, "Multi-scale convolutional neural networks for crowd counting," in *ICIP*. IEEE, 2017, pp. 465–469. 7, 8

[72] Z. Shi, L. Zhang, Y. Liu, X. Cao, Y. Ye, M.-M. Cheng, and G. Zheng, "Crowd counting with deep negative correlation learning," in *CVPR*, 2018, pp. 5382–5390. 7, 8

[73] Z. Shen, Y. Xu, B. Ni, M. Wang, J. Hu, and X. Yang, "Crowd counting via adversarial cross-scale consistency pursuit," in *CVPR*, 2018, pp. 5245–5254. 7, 8

[74] V. Ranjan, H. Le, and M. Hoai, "Iterative crowd counting," in *ECCV*, 2018, pp. 270–285. 7, 8

[75] Q. Wang, J. Gao, W. Lin, and Y. Yuan, "Learning from synthetic data for crowd counting in the wild," in *CVPR*, 2019, pp. 8198–8207. 7, 8, 9

[76] C. Liu, X. Weng, and Y. Mu, "Recurrent attentive zooming for joint crowd counting and precise localization," in *CVPR*. IEEE, 2019, pp. 1217–1226. 7, 8

[77] X. Jiang, Z. Xiao, B. Zhang, X. Zhen, X. Cao, D. Doermann, and L. Shao, "Crowd counting and density estimation by trellis encoder-decoder networks," in *CVPR*, 2019, pp. 6133–6142. 7, 8

[78] L. Liu, Z. Qiu, G. Li, S. Liu, W. Ouyang, and L. Lin, "Crowd counting with deep structured scale integration network," in *ICCV*, 2019, pp. 1774–1783. 7, 8

[79] C. Xu, K. Qiu, J. Fu, S. Bai, Y. Xu, and X. Bai, "Learn to scale: Generating multipolar normalized density maps for crowd counting," in *ICCV*, 2019, pp. 8382–8390. 7, 8

[80] H. Xiong, H. Lu, C. Liu, L. Liu, Z. Cao, and C. Shen, "From open set to closed set: Counting objects by spatial divide-and-conquer," in *ICCV*, 2019, pp. 8362–8371. 7, 8

[81] V. A. Sindagi and V. M. Patel, "Multi-level bottom-top and top-bottom feature fusion for crowd counting," *ICCV*, 2019. 7, 8

[82] D. Guo, K. Li, Z.-J. Zha, and M. Wang, "Dadnet: Dilated-attention-deformable convnet for crowd counting," in *ACM MM*, 2019, pp. 1823–1832. 7, 8

[83] X. Tan, C. Tao, T. Ren, J. Tang, and G. Wu, "Crowd counting via multi-layer regression," in *ACM MM*, 2019, pp. 1907–1915. 7, 8

[84] M.-h. Oh, P. A. Olsen, and K. N. Ramamurthy, "Crowd counting with decomposed uncertainty." in *AAAI*, 2020, pp. 11 799–11 806. 7, 8

[85] A. Luo, F. Yang, X. Li, D. Nie, Z. Jiao, S. Zhou, and H. Cheng, "Hybrid graph neural networks for crowd counting," *arXiv preprint arXiv:2002.00092*, 2020. 7, 8

[86] J. Gao, Q. Wang, and Y. Yuan, "Scar: Spatial-/channel-wise attention regression networks for crowd counting," *Neurocomputing*, vol. 363, pp. 1–8, 2019. 9

[87] L. Zhu, Z. Zhao, C. Lu, Y. Lin, Y. Peng, and T. Yao, "Dual path multi-scale fusion networks with attention for crowd counting," *arXiv preprint arXiv:1902.01115*, 2019. 9