PSGCNet: A Pyramidal Scale and Global Context Guided Network for Dense Object Counting in Remote-Sensing Images

Guangshuai Gao, Qingjie Liu, Member, IEEE, Zhenghui Hu, Lu Li, Qi Wen, and Yunhong Wang, Fellow, IEEE

Abstract—Object counting, which aims to count the accurate number of object instances in images, has been attracting more and more attention. However, challenges such as large-scale variation, complex background interference, and nonuniform density distribution greatly limit the counting accuracy, particularly striking in remote-sensing imagery. To mitigate the above issues, this article proposes a novel framework for dense object counting in remote-sensing images, which incorporates a pyramidal scale module (PSM) and a global context module (GCM), dubbed PSGCNet, where PSM is used to adaptively capture multi-scale information and GCM is to guide the model to select suitable scales generated from PSM. Moreover, a reliable supervision manner improved from Bayesian and counting loss (BCL) is utilized to learn the density probability and then compute the count expectation at each annotation. It can relieve nonuniform density distribution to a certain extent. Extensive experiments on four remote-sensing counting datasets demonstrate the effectiveness of the proposed method and its superiority compared with state of the arts. Additionally, experiments extended on four commonly used crowd counting datasets further validate the generalization ability of the model. Code is available at https://github.com/gaoguangshuai/psgcnet.

Index Terms—Bayesian loss (BL), global context, object counting, pyramidal scale, remote sensing.

I. INTRODUCTION

OBJ ect counting, which is to estimate the accurate number of object instances in images or videos, has been attracting remarkable interest in recent years owing to its potential value in traffic monitor [1], urban planning [2], public safety [3], and crowd behavior understanding [4]. Additionally, object counting has been applied in many practical applications, such as cell microscopy [5], animals [6], and remote-sensing applications [7]–[10].

Recent prevalent object counting methods have been following the pioneering work [5], which estimates the count number over a density map. Lately, driven by the powerful feature representation ability of convolutional neural networks (CNNs), a lot of CNN-based density estimation algorithms have been presented. Although remarkable progress has been achieved, there still exist challenges limiting the counting performance, such as large-scale variation, complex background interference, nonuniform density distribution, which are much tougher in remote-sensing images. Taking Fig. 1(a) as an example, the scale variation of different ship instances is large because of different types of ships. In Fig. 1(b), complex background interferences (such as the green plants) are easy to fool models to make wrong predictions. Furthermore, in Fig. 1(c), the spatial distribution is nonuniform varying from sparse to congested even in the same scene.

Many efforts tackle the scale variation problem by designing multi-column architectures [11]–[13] or employing techniques, such as dilated convolution [14], [15], spatial pyramid pooling (SPP) [15], [16], atrous SPP (ASPP) [17], [18], and inception blocks [19] to capture multi-scale information [20]–[23]. These models relieve scale variation problems, yet still have some limitations. 1) multi-column architectures or inception blocks have multiple branches built with different kernel sizes, which introduce a large number of parameters and huge computation burdens [24]; 2) the pooling operation in these models (e.g., SPP) may lead to fine-detail information loss, thus degenerating the performance; and 3) hand-crafted dilation rates are hard to match the range of scale variations. To alleviate these issues, motivated by [25], we embed a pyramidal scale module (PSM) into our framework to effectively capture multi-scale information.

Fig. 1. Illustrations of large-scale variation, complex background interference, and nonuniform density distribution. (a) Scales of ships enclosed by the red bounding boxes vary largely. (b) Objects (i.e., small vehicles) are shaded by the plants. (c) Ships at the harbor are unevenly distributed.
To suppress background distractors, visual attention has been successfully applied to the counting task [7], [8], [26] and achieves good performance. However, these attention modules suffer from heavy computation cost and high complexity, for instance, the fashionable squeeze-and-excitation network (SENet) [27] and its followers [28], [29] employ multiple fully connected (FC) layers to compute attention weights. Such designs are inefficient and not helpful for capturing interactive information across channels. Inspired by [30] and [31], we introduce an effective and efficient global context module (GCM) to select more suitable scales generated from PSM.

Most counting methods convert annotated points into density maps using Gaussian filters and then train CNN models using $L_2$ loss. Consequently, the counting performance highly depends on the quality of the “ground-truth” density map. However, such a pixel-independent-based “ground-truth” density map generation manner may be suboptimal, especially in nonuniform distributed regions. As an alternation, Ma et al. [32] propose a reliable supervision manner through learning the count expectation from the point annotations, named Bayesian loss (BL). This effective supervision manner could alleviate the problem of nonuniform density distribution. However, there may exist an inconsistency between the 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 the BL, we add a counting loss to mitigate this issue.

In summary, the contributions of this work are threefold.

1) A novel Pyramidal Scale and Global Context-based framework for dense object counting in remote-sensing images, termed PSGCNet, is presented.

2) A flexible pyramid scale module is designed to effectively extract multi-scale features of dense scenes. A lightweight GCM is embedded to make use of the rich interaction information across channels of feature maps to guide the model to select more suitable scales.

3) Extensive experiments conducted on four remote-sensing object counting (RSOC) datasets demonstrate the effectiveness and superiority of the proposed approach, and the extension to four commonly used crowd counting datasets further validate the generalization ability and robustness of our proposed method.

The remainder of this article is organized as follows. The related work of object counting algorithms is briefly surveyed in Section II. The details of our proposed method are introduced in Section III, following which experimental results and analysis are presented in Section IV. Finally, the conclusion is concluded in Section V.

II. RELATED WORK

A. Object Counting in Congested Scenes

Early object counting methods are mainly detection based [33], [34], they first detect the interested object instances and then count the number of the bounding boxes. These methods obtain satisfactory performance in sparse scenarios, thanks to the powerful detectors. However, they may fail in highly congested scenes, since the object instances are usually with small sizes and easily confused with background distractors. Another mainline is regression-based methods, which map the high-dimensional image space to natural numbers [35], [36]. As a highly nonlinear regression task, it is very hard to optimize models and the performance is far from satisfactory. Lempitsky and Zisserman [5] rekindled the counting task as a density map generation problem, which estimates the counting number of object instances by integrating all the pixels of the density map. Entering the deep learning era, the performance of object counting has been significantly improved. Many deep neural networks have been designed for tackling the counting task. The performances on several representative benchmark datasets such as ShanghaiTech [11], University of Central Florida-Qatar National Research Fund (UCF-QNRF) [37], and University of Central Florida_Crowd Counting_50 (UCF_CC_50) [38] have reached promising results. For a comprehensive review of the counting task, refer to [39] and [40].

B. Object Counting From the Remote-Sensing View

Capturing from a remote distance, aerial images or videos provide a wider field of view and thus with much more complex scene contents, which brings great challenges to existing counting models. To facilitate research in this field, Bahmanyar et al. [41] introduced a drone-based crowd dataset and developed a multi-resolution network for estimating the number of pedestrians in aerial images. Layout proposal network (LPN) [9] takes advantage of the regular spatial layout of cars and proposes a spatial LPN for car counting and localization, simultaneously. Inclusion-exclusion principle (IEP) [42] proposes to predict the image-level count by dividing the image into a set of divisions. It achieves good performance on several drone-based counting datasets. Li et al. [43] draw inspiration from the object detectors and proposed to detect and count cars simultaneously using a unified framework. Space-time neighbor-aware network (STNNet) [44] takes a step further and performs the density map estimation, localization, and tracking tasks in one network. Attention, scale pyramid, deformable network (ASPDNet) [8] builds a new benchmark for aerial image counting. It employs recently developed techniques such as dilated convolution, attention, and deformable convolution to achieve better performance.

C. Alleviating Large-Scale Variation

Scale variation is a great challenge for object counting. Four strategies are widely studied to address this problem: multi-column network architectures, dilated convolution, SPP, and inception module. For example, multi-column convolution neural network (MCNN) [11] is a simple multi-column network, in which each column is built with different filter kernels. Switch-CNN [45] adopts a frame structure similar to MCNN [11]. The difference is that a specialized classifier is applied to select a suitable column network for inputs. Congested scene recognition network (CSRNNet) [20] takes advantage of dilated convolution to enlarge the receptive fields without increasing computation cost. Context-aware network (CAN) [22] combines scale-aware and context-aware feature
information to boost the performance. Scale aggregation network (SANet) [23] captures multi-scale features built on the shoulder of the Inception module [19]. Dense scale network (DSNet) [46] cascades multiple dense dilated convolution blocks and links them with dense residual connections. Adaptive dilated network with self-correction supervision (ADSCNet) [47] adopts adaptive dilated convolution to learn dynamic and continuous dilated rates for each pixel location. Multi-resolution crowd network (MRCNet) [41] combines low-level and high-level features with lateral connections to learn contextual and detailed local information in aerial imagery. Scale-adaptive long-range context-aware network (SACANet) [48] utilizes a pyramid contextual module to extract long-range contextual information and enlarge the receptive fields of the objects in drone scenes. ASPDNet [7], [8] integrates a scale pyramid module to capture multi-scale information for counting in remote-sensing images.

D. Mitigating Cluttered Background Interferences

Attention mechanism has been widely used to suppress cluttered backgrounds and highlight foreground regions. For instance, scale-aware attention network (SAANet) [49] develops a soft attention mechanism to learn a set of gating masks to aggregate the multi-scale density maps. Attention-injective deformable convolutional network (ADCrowdNet) [26] combines visual attention and deformable convolution [50] into a unified framework. Hierarchical attention-based crowd counting network (HA-CCN) [51] designs a hierarchical attention-based network to selectively enhance the features at various levels. Relational attention network (RANet) [52] and ANF [53] incorporate self-attention to capture long-range dependencies of the feature maps. Shallow feature based dense attention network (SDANet) [54] builds a dense attention network based on shallow features. Attention-scaling network (ASNNet) [55] learns attention scaling factors and automatically adjusts the density regions by multiplying multiple density attention masks on them. SACANet [48] leverages a scale-adaptive self-attention multi-branch module to address isolated clusters in aerial images. ASPDNet [7], [8] cascades channel attention and spatial attention to relieve the impact of complex cluttered backgrounds in diverse remote-sensing scenarios. These methods have gained significant performance; nevertheless, the sophisticated structures of the attention modules incorporated in them introduce a large number of parameters, thus making them suffer from huge computation burdens. Although some lightweight attention modules such as SENets [27] and convolution block attention module (CBAM) [28] are developed to alleviate this problem, the FC layers still have many parameters. What is more, the channel dimensionality reduction in these models also limits the upper bound of the performance.

Different from the aforementioned methods, our proposed PSGCNet takes advantage of a PSM 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 GCM, essentially a lightweight channel attention operation. It can not only reduce the computation burden of attention modules, but also make 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. PROPOSED METHOD

A. PSGCNet Overview

The architecture of PSGCNet is illustrated in Fig. 2. It has four key components, including a backbone network as a feature extractor, a PSM capturing multi-scale information, a GCM suppressing cluttered backgrounds, and a decoder to estimate the final density map.

Specifically, we adopt a truncated visual geometry group 19 (VGG19) [56] same to [32] as the backbone network, in which the three FC layers and one pooling layer are removed. The output feature map’s resolution of the backbone is 1/16 of the original input image. Afterward, a pyramid scale module is built on top of the feature maps to capture multi-scale information. Then, an effective GCM followed is leveraged to restrain the complex backgrounds. Then feature maps are upsampled twice with bilinear interpolation operation. Finally, a decoder is equipped to produce the density map, in which three successive convolutional layers are used, including two 3 × 3 convolution layers with 256 and 128 channels, and one 1 × 1 convolution. To further improve the performance, we optimize the model using a modified BL.

B. Pyramidal Scale Module

Scale variation is a critical problem in remote-sensing image understanding. In this article, we attack this problem by introducing a PSM. The PSM deploys two paralleled network paths: a local PyConv path and a global PyConv path. The two paths have a dual-oriented pyramid architecture, enabling richer multi-scale information capturing.
PyConv has a pyramid structure, as shown in Fig. 3. It contains increasing kernel sizes from bottom to top in a pyramidal manner and decreasing kernel depths (connectivity) with grouped convolution. The double-oriented pyramid operation allows the model to capture richer multi-scale information, from larger receptive fields of kernels with lower connectivity to smaller receptive fields with higher connectivity. This design is efficient, flexible, and cheap computational cost. It could also boost the robustness of the model to scale variation.

C. Global Context Module

Visual attention has been claimed as a promising solution to overcome the interference of complex backgrounds. These models have achieved improved performance, however, with a cost of higher model complexities and heavier computational burden, since they usually use self-attention [57] or non-local modules [58].

Drawing inspiration from [30] and [31], we propose an efficient and lightweight GCM to model the dependencies across the channels. The GCM designed in our work is depicted in Fig. 5.

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. Let $x_c$ be the feature map corresponding to the $c$th channel, that is, $x_c = [x_c^{i,j}]_{H \times W} \in \mathbb{R}^{H \times W}, c \in \{1, 2, \ldots, C\}$. A GCM is embedded to capture global context information of each channel. The module is formulated as

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

where $\alpha_c$ denotes the embedding weight and $\epsilon$ is a small constant to avoid the deviation at zero points. This GCM is somewhat similar to the global average pooling (GAP) but more robust than it [31].

Generally, to effectively learn cross-channel interactions, typical solutions are SENet [27] or CBAM [28], however, they destroy the correspondence between channels. Here, we adopt an alternation strategy, which first adaptively determines the kernel sizes $k$ ($k = 3$ in this article) and then performs a 1-D convolution operation, that is,

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

where $C1D$ means 1-D convolution.

A subsequent channel normalization is applied, which can be formulated as

$$\bar{s}_c = \sqrt{C} \hat{s}_c = \frac{\sqrt{C} \hat{s}_c}{\|s\|_2} \tag{3}$$

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

$$\tilde{x}_c^{\text{att}} = \tanh(w_c \tilde{s}_c + b_c) \tag{4}$$
respectively.

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

D. BCL Function

To optimize models, Euclidean distance ($L_2$ loss) between the prediction and the ground-truth density maps is widely used. However, the loss is not robust to the occlusion, scale variation, and nonuniform density. Recently, Ma et al. [32] proposed a novel supervision manner, named BL to relieve this problem. It constructs a density contribution model from point annotations and then defines the loss as the difference between the count expectation and the ground-truth number at each annotated point

$$L_{Bayesian} \equiv \sum_{n=1}^{N} (1 - E[c_n]) + \mathcal{F}(0 - E[c_n]) \quad (5)$$

where $N$ is the total number of labeled objects and $E[c_n]$ and $E[c_0]$ indicate the expected counts for each instance and the entire background, respectively. The first term denotes that impelling the foreground count at each annotation point equals 1, while the second term means enforcing the background count to be zero. $\mathcal{F}()$ is a distance function, and we adopt $\ell_1$ distance metric as suggested in [32].

Although reliable and effective, there may exist inconsistencies between the training phase and the testing stage. Therefore, apart from BL, we add a counting loss to mitigate this issue. The counting loss is defined as

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

where $F(X_i; \Theta)$ and $Y_i$ represent the count integrated by the estimated density map and ground-truth count of the $i$th image, respectively. $\Theta$ denotes training parameters and $\| \cdot \|_1$ means $\ell_1$-norm.

Therefore, the overall loss function is the combination of BL $L_{Bayesian}$ and counting loss $L_{Count}$

$$L_{Overall} = L_{Bayesian} + \lambda L_{Count} \quad (7)$$

where $\lambda$ is a tunable positive hyperparameter.

IV. EXPERIMENTAL RESULTS

In this section, the datasets, evaluation protocols, and implementation details are first introduced. Then ablation studies and comparisons 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 Protocols

1) Datasets: Extensive experiments are conducted on four RSOC datasets, including RSOC [7], [8], car parking lot dataset (CARPK) [9], pontifical catholic university of parana + (PUCPR+) [9], and drone-crowd [44] to evaluate the effectiveness and superiority of the proposed approach. Moreover, to validate the generalization ability and robustness of the model, we also conduct experiments on four widely used crowd counting datasets, that is, ShanghaiTech Part_A and Part_B [11], UCF-QNRF [37], and UCF_CC_50 [38]. The statistics of the datasets is presented in Table I.

1) RSOC [7], [8] is a RSOC dataset, which is composed of four categories, including buildings, small vehicles, large vehicles, and ships. The dataset consists of 3057 images with 286 539 instances in total. In 2468 building images, 1205 and 1263 are used for training and testing; in 280 small vehicle images, 222 images for training and 58 for testing; in 172 large vehicle images, 108 for training and 64 for testing; in 137 ship images, 97 images for training and 40 images for testing, respectively.

2) CARPK [9] is a large-scale drone-view car counting dataset, which contains 1448 images with nearly 90k cars in total, of which 989 images for training and the remaining 459 images for testing.

3) PUCPR+ [9] is also a car counting dataset, where all the images are captured from the tenth floor of a building. The dataset contains 125 images with approximately 17k cars, of which 100 images are served as a training set, and the rest as a testing set.

4) Drone-crowd [44] is a drone-captured dataset for density map estimation, crowd localization, and tracking, which is composed of 112 video clips with 33,600 frames in total. The video clips are annotated with over 4.8 million head annotations and several video-level attributes. All the images are captured by drone-mounted cameras in 70 different scenarios across four different cities in China (i.e., Tianjin, Guangzhou, Daqing, and Hong Kong). For the counting task in this article, we split the dataset into training and test sets, of which 24,600 images for training and the remaining 9000 for testing.

5) ShanghaiTech [11] includes two parts, that is, Part_A and Part_B, with a total number of 1198 images. The images of Part_A are randomly crawled from the Internet, which are across diverse scenes and largely varied densities. Part_A has 482 images, of which 300 are served as the 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.

---

1 https://github.com/gaoguangshuai/counting-from-sky-a-large-scale-dataset-for-remote-sensing-object-counting-and-a-benchmark-method
2 https://lafi.github.io/lpn/
3 https://github.com/visdrone/visdrone-dataset
4 https://pan.baidu.com/s/1muys1z
### TABLE I

| Dataset          | Sensor    | #Images | Training/Test | Average Resolution | Count Statistics |
|------------------|-----------|---------|---------------|--------------------|------------------|
|                  |           |         |               |                    | Total | Min  | Average | Max  |
| RSOC_building [7]| Satellite | 2468    | 1205/1263     | 512×512            | 76,215 | 15   | 30.88   | 142  |
| RSOC_small-vehicle [7] | Satellite | 280     | 222/58        | 2473×2339          | 148,838 | 17   | 531.56  | 8531 |
| RSOC_large-vehicle [7] | Satellite | 172     | 108/64        | 1552×1573          | 16,594  | 12   | 96.48   | 1336 |
| RSOC_ship [7]    | Satellite | 137     | 97/40         | 2558×2668          | 44,892  | 50   | 327.68  | 1661 |
| CARPK [9]        | Drone     | 1448    | 989/459       | 720×1280           | 89,777  | 1    | 62      | 188  |
| PUCPR+ [9]       | Camera    | 125     | 100/25        | 720×1280           | 16,915  | 0    | 135     | 331  |
| DroneCrowd [44]  | Drone     | 33,600  | 24,600/9,000  | 1920×1080          | 4,864,280 | 25   | 144.8   | 455  |
| SHT_A [11]       | CCTV      | 482     | 300/182       | 589×868            | 241,677 | 33   | 501.4   | 3,139 |
| SHT_B [11]       | CCTV      | 716     | 400/316       | 768×1024           | 88,488  | 9    | 123.6   | 578  |
| UCF-QNRF [37]    | CCTV      | 1,535   | 1201/334      | 2013×2902          | 1,251,642 | 49  | 815     | 12,865 |
| UCF_CC_50 [38]   | CCTV      | 50      | –             | 2101×2888          | 63,974  | 94   | 1,280   | 4,543 |

Fig. 6. (Top row) Input images. (Middle row) Density maps generated by baseline and (bottom row) our method. The ground-truth and estimated counts are put at the bottom of each image. Compared with the baseline, our proposed model can obtain more accurate estimations across diverse scenarios.

6) **UCF-QNRF [37]** is a recently released large and challenging dataset, which has a wide range of image resolutions, counts, scale variations, and diverse density distribution. The images of this dataset are crawled from Flickr, Web Search, and Hajj footage, containing 1535 images with over 125 million point annotations, where 1201 images are used for training and the remaining 334 images for testing.

7) **UCF_CC_50 [38]** is composed of 50 images with various resolutions. The dataset is small-scale, yet challenging since the average count is up to 1280.

Following [38], fivefold cross-validation is performed to obtain the final test result.

2) **Evaluation Protocol**: Two most widely used evaluation metrics, that is, 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:

\[
MAE = \frac{1}{K} \sum_{i=1}^{K} |\hat{C}_i - C_i| \\
RMSE = \sqrt{\frac{1}{K} \sum_{i=1}^{K} (\hat{C}_i - C_i)^2}
\]
Fig. 7. Visualization results on the CARPK dataset. (Top row) The original image and the ground-truth counts. (Bottom row) The density maps generated by our proposed method and the estimated counts.

TABLE II
DIFFERENT SETTINGS ON RSOC_BUILDING DATASET

| Baseline | PSM | GCM | MAE  | RMSE |
|-----------|-----|-----|------|------|
| ✓         | ✓   | ✓   | 11.51| 15.96|
| ✓         | ✓   | ✓   | 9.06 | 13.24|
| ✓         | ✓   | ✓   | 8.64 | 12.08|
| ✓         | ✓   | ✓   | 7.54 | 10.32|

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.

B. Implementation Details

We implement our proposed PSGCNet in PyTorch and train it in an end-to-end manner. All the experiments are conducted on one NVIDIA 2080Ti GPU. A truncated VGG19 [56] pre-trained on ImageNet [59] is taken as the backbone, with the FC layers and the last pooling layer removed. During training, the initial learning rate is 1e-4, and an Adam optimizer is used. For better training and avoiding overfitting, random crop and horizontal flipping are applied for augmentation. Specifically, the crop size is 256 x 256 for RSOC_building datasets, ShanghaiTech Part_A, and UCF_CC_50, and 512 x 512 for RSOC_small-vehicle, RSOC_large-vehicle, RSOC_ship, CARPK, PUCPR+, DroneCrowd, ShanghaiTech Part_B, and UCF_QNRF, since they have large image sizes. In addition, for all the datasets, 10% of the images are randomly sampled for validation from each training set. The batch size is set to 1 for all the datasets.

C. Ablation Studies

To validate the effectiveness of each module of our approach, we conduct ablation studies on the RSOC_building dataset. The baseline method is BL [32]. The specific settings are shown in Table II.

1) Effect of PSM and GCM: From Table II, we can observe that when PSM is introduced, the performance can achieve a significant improvement. Specifically, there are relative improvements of 21.28% and 17.04% with respect to MAE and RMSE, demonstrating the robustness of the proposed PSM to the problem of large-scale variation. To validate the robustness of the model to the complex background interference, we adopt a GCM. 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 24.93% and 24.31% with respect to MAE and RMSE, which proves that it has made a significant impact on highlighting objects parts while diminishing background noise.

2) Effect of the Hyperparameter \( \lambda \): To verify the effectiveness of the proposed BCL function, we conduct experiments under the condition of different \( \lambda \). As can be observed from Table III, when \( \lambda = 0.1 \), we can obtain the best performance.

3) Different Backbones: Our proposed modules and loss function can be readily applied to any network structure to improve performance. Here, we apply them to VGG-19 and VGG-16 backbones and perform comparisons with the \( L_2 \) loss-based models. The quantitative results from Table IV show that our method achieves better performance compared with the naive \( L_2 \)-based modes by a considerable margin.

D. Comparisons on RSOC Dataset

We compare our approach with state-of-the-art methods and show results in Table V and visualize some representative density maps in Fig. 6. Our model achieves substantial improvements on all four subsets. Specifically, we improve the baseline with improvements of 34.49%, 6.76%, 17.85%, and 11.01% on building, small-vehicle, large-vehicle, and ship subsets in terms of MAE, respectively, indicating that our proposed method has a strong counting performance.

E. Comparisons on CARPK and PUCPR+ Datasets

Table VI reports the MAE and RMSE results on two car counting datasets, that is, CARPK and PUCPR+ [9]. We compare our proposed approach with state-of-the-art car counting methods including detection-based counting methods (you only look once (YOLO) [64], Faster regional convolutional neural network (RCNN) [65], LPN [9], SSD [67], YOLO9000 [68], RetinaNet [69], and IEP Counting [42]), a regression-based counting method (one-look regression [66]), and density map estimation-based methods (MCNN [11], CSRNet [20], and BL [32]). The results reveal that our method consistently performs better than the comparative methods, which demonstrates the superiority of our method both in sparse and congested scenarios. Specifically,
Fig. 8. (Top row) Input images. (Middle row) Density maps generated by baseline and (bottom row) our method. 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 estimations from sparse to highly congested scenes.

**TABLE V**

| Methods  | Datasets | Year & Venue | RSOC_Building | RSOC_Small-vehicle | RSOC_Large-vehicle | RSOC_Ship |
|----------|----------|--------------|----------------|-------------------|-------------------|-----------|
|          |          |              | MAE | RMSE | MAE | RMS | MAE | RMS | MAE | RMS | MAE | RMS |
| MCNN [11] | 2016 CVPR | 13.65 | 15.66 | 488.65 | 1371.44 | 36.56 | 55.55 | 265.91 | 412.30 |
| CMTL [60] | 2017 AVSS | 12.78 | 15.99 | 490.53 | 1321.11 | 61.02 | 78.25 | 251.17 | 403.07 |
| CSRNet [20] | 2018 CVPR | 8.00 | 11.78 | 443.72 | 1252.22 | 34.10 | 46.42 | 240.01 | 394.81 |
| SANet [23] | 2018 ECCV | 29.01 | 32.96 | 497.22 | 1276.66 | 62.78 | 79.65 | 302.37 | 436.91 |
| SFCN [61] | 2019 CVPR | 8.94 | 12.87 | 440.70 | 1248.27 | 33.93 | 49.74 | 240.16 | 394.81 |
| SPN [21] | 2019 WACV | 7.74 | 11.48 | 445.16 | 1252.92 | 36.21 | 50.65 | 241.43 | 392.88 |
| SCAR [52] | 2019 NC | 26.90 | 31.35 | 497.22 | 1276.65 | 62.78 | 79.64 | 302.37 | 436.92 |
| CAN [22] | 2019 CVPR | 9.12 | 13.38 | 457.56 | 1260.39 | 34.56 | 49.63 | 282.69 | 423.44 |
| SPANet [63] | 2019 CVPR | 8.18 | 11.75 | 435.29 | 1284.15 | 29.04 | 47.01 | 201.61 | 332.87 |
| BL [32] | 2019 ICCV | 11.51 | 15.96 | 168.62 | 280.50 | 13.39 | 35.24 | 84.18 | 136.21 |
| ASPDNet [7], [8] | 2020 ICASSP/TGRS | 7.59 | 10.66 | 433.23 | 1238.61 | 18.76 | 31.06 | 193.83 | 318.95 |
| PSGCNet (Ours) | – | **7.54** | **10.52** | **157.55** | **245.31** | **11.00** | **17.65** | **74.91** | **112.11** |

**TABLE VI**

| Methods  | Datasets | CARPK [9] | PUCPR+ [9] |
|----------|----------|-----------|------------|
|          |          | MAE | RMSE | MAE | RMSE | MAE | RMSE |
| YOLO [64] | 102.89 | 110.02 | 156.72 | 200.54 |
| Faster RCNN [65] | 48.89 | 57.55 | 156.00 | 200.42 |
| One-Look Regression [66] | 103.48 | 110.64 | 156.70 | 200.39 |
| SSD [67] | 39.46 | 66.84 | 21.88 | 36.73 |
| YOLO90000 [68] | 37.33 | 42.32 | 119.24 | 132.22 |
| LIP [9] | 38.59 | 43.18 | 97.96 | 133.25 |
| ReinaNet [69] | 23.80 | 36.79 | 22.76 | 34.46 |
| LEP [42] | 16.62 | 22.30 | 24.58 | 33.12 |
| MCNN [11] | 51.83 | 15.17 |
| CSRNet [20] | 39.10 | 43.30 | 21.86 | 29.53 |
| BL [32] | 11.48 | 13.32 | 8.65 | 10.24 |
| PSGCNet (Ours) | **8.15** | **10.46** | **5.24** | **7.36** |

compared with several outstanding object detectors such as Faster RCNN [65] and YOLO [64], our proposed method surpasses them by a large margin. Moreover, compared with one-look regression [66], our approach shows better performance. We conjecture that it may be uncontrollable when regressing the count directly. Furthermore, compared with the density map estimation methods, that is, MCNN [11], CSRNet [20], and BL [32], our proposed method still obtains the highest count scores. We visualize some qualitative results in Fig. 7. It demonstrates that the proposed method not only performs a better counting performance, but also shows strong localization ability.

**F. Comparisons on DroneCrowd Dataset**

We also evaluate our method on a more challenging dataset, called DroneCrowd [44]. Table VII lists the counting results with respect to MAE and RMSE, PSGCNet achieves comparable performance when compared with the state-of-the-art methods. To further analyze the results, we also report the performance on several subsets according to three video-level attributes, that is, two categories of scales, including Large (the diameter of objects > 15 pixels) and Small (the diameter...
TABLE VII
PERFORMANCE COMPARISON ON THE DRONE CROWD DATASET [44]

| Methods | Speed (FPS) | Overall | Large | Small | Crowded | Sunny | Night | Sparse |
|---------|-------------|---------|-------|-------|---------|-------|-------|--------|
| MCNN [11] | 28.98 | 34.7 | 42.5 | 36.3 | 44.1 | 31.7 | 40.1 | 21.0 | 27.5 | 39.0 | 43.9 | 67.2 | 68.7 | 29.5 | 35.3 | 37.7 | 46.2 |
| CMTL [60] | 3.31 | 36.7 | 65.9 | 53.3 | 65.2 | 81.5 | 69.7 | 59.5 | 68.9 | 56.5 | 67.8 | 48.2 | 58.3 | 81.6 | 88.7 | 42.2 | 47.9 |
| MSGNet [76] | 1.79 | 58.0 | 75.5 | 58.4 | 77.9 | 57.8 | 71.1 | 64.5 | 83.8 | 53.8 | 63.3 | 56.1 | 57.3 | 71.4 | 104.6 | 38.7 | 48.8 |
| LCPCNN [73] | 3.08 | 106.9 | 130.6 | 126.3 | 140.3 | 128.2 | 154.8 | 147.1 | 160.3 | 137.1 | 151.7 | 156.6 | 113.8 | 208.5 | 211.1 | 95.4 | 110.0 |
| SwitchesCNN [35] | 0.01 | 66.5 | 77.8 | 61.3 | 74.2 | 74.0 | 83.0 | 58.0 | 63.4 | 69.0 | 80.9 | 92.8 | 105.8 | 57.7 | 79.8 | 65.7 | 78.7 |
| ASSCF [36] | 1.38 | 58.1 | 60.2 | 57.0 | 70.6 | 54.8 | 39.7 | 42.5 | 46.4 | 39.3 | 44.3 | 56.6 | 106.6 | 34.0 | 41.3 | 55.1 | 65.5 |
| AMGCN [75] | 0.16 | 165.6 | 167.7 | 166.7 | 164.9 | 163.8 | 165.9 | 160.5 | 162.3 | 174.8 | 171.1 | 162.3 | 164.3 | 165.5 | 167.7 | 165.6 | 167.8 |
| StartPooling [72] | 0.73 | 68.8 | 77.7 | 68.7 | 77.1 | 68.8 | 77.3 | 66.5 | 75.9 | 74.0 | 83.4 | 85.2 | 67.4 | 95.7 | 101.1 | 53.1 | 59.1 |
| STNNet [44] | 3.31 | 68.5 | 65.3 | 67.6 | 73.8 | 31.7 | 41.3 | 45.7 | 57.3 | 36.5 | 32.3 | 40.5 | 34.0 | 38.5 | 43.4 | 25.9 | 38.9 |
| CSRNet [40] | 3.92 | 19.8 | 21.6 | 17.3 | 25.4 | 22.7 | 25.9 | 12.8 | 16.6 | 19.1 | 22.5 | 29.3 | 43.8 | 30.2 | 24.0 | 19.7 | 25.5 |
| CAN [22] | 7.14 | 22.1 | 35.4 | 18.9 | 26.7 | 26.9 | 41.8 | 11.2 | 14.9 | 14.8 | 17.5 | 69.4 | 73.6 | 14.4 | 17.9 | 25.6 | 39.7 |
| DM-Count [16] | 10.04 | 18.4 | 25.7 | 19.2 | 29.6 | 17.2 | 22.4 | 12.4 | 16.8 | 12.6 | 13.2 | 51.1 | 55.7 | 17.8 | 21.8 | 18.9 | 29.6 |
| STNNet-FC [44] | 3.41 | 15.8 | 18.7 | 16.9 | 18.4 | 15.4 | 32.8 | 18.1 | 21.7 | 19.7 | 23.2 | 24.2 | 14.5 | 18.5 | 21.8 | 14.3 | 14.9 |
| PGSCNet (Ours) | 4.79 | 24.7 | 31.9 | 24.7 | 32.8 | 24.0 | 31.3 | 18.8 | 21.6 | 13.4 | 18.6 | 26.7 | 54.5 | 18.2 | 21.2 | 18.5 | 21.2 |

TABLE VIII
COMPARISONS OF OUR PROPOSED PGSCNet WITH 15 STATE-OF-THE-ART METHODS ON FOUR CROWD COUNTING DATASETS. THE RESULTS OF BASELINE IS THE REIMPLEMENTED VERSION WITH THE PRETRAINED WEIGHTS AT HTTPS://GITHUB.COM/ZHIHENGCV/BAYESIAN-CROWD-COUNTING

of objects ≤ 15 pixels), three categories of illumination conditions, including Cloudy, Sunny, and Night, two density levels, including Crowded (with the number of objects in each frame larger than 150) and Sparse (with the number of objects in each frame less than 150). From the performance of subsets, we can find that our proposed method performs well in the Cloudy, Sunny, and Crowded subsets and degrades in the Night subset, which may be attributed to extremely low illumination and severe class imbalance. In particular, STNNet [44] performs the best across the whole dataset. It is a multi-task learning model to jointly solve density map estimation, localization, and tracking. The method also leverages both spatial and temporal information, in which a neighboring context loss is applied to capture relations among neighboring targets in consecutive frames. Even so, our proposed model achieves comparatively good performance and even surpasses it in the Sunny and Crowded subsets.

G. Comparisons on Crowd Counting Datasets
To further validate the generalization ability and robustness of the proposed model, we extend it on four widely used crowd counting datasets, and the counting results are reported in Table VIII. It demonstrates that our proposed approach can achieve consistent improvements compared with 15 state-of-the-art methods [11], [20], [22], [32], [45], [54], [55], [60], [61], [72], [77]–[82]. Specifically, on the ShanghaiTech dataset, our proposed model increases relative improvements of 12.4%/5.9% on Part_A and 15.4%/28.1% on Part_B with respect to MAE/RMSE. Even on the more crowded UCF_QNRF and UCF_CC_50, we still improve the baseline with relative improvements of 12.5%/11.9% and 20.9%/14.5% with respect to MAE/RMSE. It indicates that our proposed method achieves superior performance not only for sparse, but also highly congested crowd scenes.

In consideration of some methods that may perform well on one dataset and poorly on others, for fairness, we adopt the average ranking evaluation strategy [55] to make a comprehensive evaluation (denoted by avg.R. in Table VIII). The average ranking value is obtained by summing all ranks that one method gains to divide the number of datasets it utilizes. The lower value indicates a higher rank. Therefore, our proposed method obtains the best average ranking, which reveals its powerful ability to deal with diverse crowd scenes.

We visualize some estimated density maps of the proposed method and the baseline in Fig. 8, 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 scale variation of the pedestrians. Compared with the baseline method, our proposed model obtains more accurate estimations across diverse scenes from sparse to highly congested. Moreover, compared with baseline, our method obtains clearer density maps and shows stronger localization ability to certain extent.

V. CONCLUSION

In this article, we have presented a novel supervised learning framework for dense object counting in remote-sensing images, named PSCGNet. Our PSCGNet is characterized by three components: 1) capturing multi-scale features with an effective PSM; 2) alleviating the interferences of complex background with a lightweight GCM; and 3) a reliable supervision manner combined with BL and counting loss, which is utilized to train the network and learn the count expectation at each annotation point. Extensive experiments on four RSOC datasets demonstrate the effectiveness and superiority of the proposed approach. Moreover, extension experiments on four widely used crowd counting benchmark datasets 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 Trans. Circuits Syst. Video Technol., vol. 29, no. 5, pp. 1408–1422, May 2018.
[2] T. Li, H. Chang, M. Wang, B. Ni, R. Hong, and S. Yan, “Crowded scene analysis: A survey,” IEEE Trans. Circuits Syst. Video Technol., vol. 25, no. 3, pp. 367–386, Mar. 2015.
[3] S. Zhang, G. Wu, J. P. Costeira, and J. M. Moura, “Understanding traffic density from large-scale web camera data,” in Proc. IEEE/CVF Conf. Vis. Pattern Recognit., Oct. 2017, pp. 5898–5907.
[4] C. Zhang, K. Kang, H. Li, X. Wang, R. Xie, and X. Yang, “Data-driven crowd understanding: A baseline for a large-scale crowd dataset,” IEEE Trans. Multimedia, vol. 18, no. 6, pp. 1048–1061, Jun. 2016.
[5] V. Lemptisky and A. Zisserman, “Learning to count objects in images,” in Proc. Adv. Neural Inf. Process. Syst., 2010, pp. 1–9, 2016.
[6] C. Arteta, V. Lemptisky, A. Zisserman, “Counting in the wild,” in Computer Vision—ECCV 2016 (Lecture Notes in Computer Science), vol. 9911, B. Leibe, J. Matas, N. Sebe, and M. Welling, Eds. Cham, Switzerland: Springer, 2016, doi: 10.1007/978-3-319-46478-7_30.
[7] G. Gao, Q. Liu, and Y. Wang, “Counting dense objects in remote sensing images,” in Proc. IEEE Int. Conf. Acoust., Speech Signal Process. (ICASSP), May 2020, pp. 4137–4141.
[8] G. Gao, Q. Liu, and Y. Wang, “Counting from sky: A large-scale data set for remote sensing object counting and a benchmark method,” IEEE Trans. Geosci. Remote Sens., vol. 59, no. 5, pp. 3642–3655, May 2021.
[9] M.-R. Hsieh, Y.-L. Lin, and W. H. Hsu, “Drone-based object counting by spatially regularized regional proposal network,” in Proc. IEEE Int. Conf. Comput. Vis. (ICCV), Oct. 2017, pp. 4145–4153.
[10] D. Du et al., “The unmanned aerial vehicle benchmark: Object detection and tracking,” in Proc. Eur. Conf. Comput. Vis., 2018, pp. 370–386.
[11] Y. Zhang, D. Zhou, S. Chen, S. Gao, and Y. Ma, “Single-image crowd counting via multi-column convolutional neural network,” in Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR), Jun. 2016, pp. 589–597.
[12] V. A. Sindagi and V. M. Patel, “Generating high-quality crowd density maps using contextual pyramid CNNs,” in Proc. IEEE Int. Conf. Comput. Vis. (ICCV), Oct. 2017, pp. 1861–1870.
[13] 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 Proc. 27th ACM Int. Conf. Multimedia, 2019, pp. 1897–1906.
[14] F. Yu and Y. Koltun, “Multi-scale context aggregation by dilated convolutions,” in Proc. Int. Conf. Learn. Represent., 2016, pp. 1–13.
[15] M. Lan, Y. Zhang, L. Zhang, and B. Du, “Global context based automatic road segmentation via dilated convolutional neural network,” Inf. Sci., vol. 535, pp. 156–171, Oct. 2020.
[16] 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, Jan. 2014.
[17] 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 Trans. Pattern Anal. Mach. Intell., vol. 40, no. 4, pp. 834–848, Apr. 2017.
[18] L. Li, Q. Dai, X. Wu, Y. Zhang, and D. Tao, “Stage-wise unsupervised domain adaptation with adversarial self-training for road segmentation of remote-sensing images,” IEEE Trans. Geosci. Remote Sens., vol. 60, 2022, Art. no. 5609413.
[19] C. Szegedy et al., “Going deeper with convolutions,” in Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR), Jun. 2015, pp. 1–9.
[20] Y. Li, X. Zhang, and D. Chen, “CSKNet: Dilated convolutional neural networks for understanding the highly congested scenes,” in Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit., Jun. 2018, pp. 1091–1100.
[21] X. Chen, Y. Bin, N. Sang, and C. Gao, “Scale pyramid network for crowd counting,” in Proc. IEEE Winter Conf. Appl. Comput. Vis. (WACV), Jan. 2019, pp. 1941–1950.
[22] W. Liu, M. Salzmann, and P. Fua, “Context-aware crowd counting,” in Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR), Jun. 2019, pp. 5099–5108.
[23] X. Cao, Z. Wang, Y. Zhao, and F. Su, “Scale aggregation network for accurate and efficient crowd counting,” in Proc. Eur. Conf. Comput. Vis., 2018, pp. 734–750.
[24] J. He, Z. Deng, and Y. Qiao, “Dynamic multi-scale filters for semantic segmentation,” in Proc. IEEE/CVF Int. Conf. Comput. Vis., 2019, pp. 3562–3572.
[25] I. Cosmin Duta, L. Liu, F. Zhu, and L. Shao, “Pyramidal convolution: Rethinking convolutional neural networks for visual recognition,” 2020, arXiv:2006.11538.
[41] R. Bahmanyar, E. Vig, and P. Reinitz, “MRCNet: Crowd counting and density map estimation in aerial and ground imagery,” in Proc. BMVC Workshop Object Detection Regress. Screening, 2019, pp. 1–5.

[42] T. Stahl, S. L. Pintea, and J. C. van Gemert, “Divide and count: Generic object counting by image divisions,” IEEE Trans. Image Process., vol. 28, no. 2, pp. 1035–1044, Feb. 2019.

[43] W. Li, S. Li, Q. Wu, X. Chen, and K. N. Ngan, “Simultaneously detecting and counting dense vehicles from drone images,” IEEE Trans. Ind. Electron., vol. 66, no. 12, pp. 9651–9662, Dec. 2019.

[44] L. Wen et al., “Detection, tracking, and counting meets drones in crowds: A benchmark,” in Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR), Jun. 2021, pp. 7812–7821.

[45] D. Li, S. Surya, and R. V. Babu, “Switching convolutional neural network for crowd counting,” in Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR), Jul. 2017, pp. 4031–4039.

[46] F. Dai, H. Liu, Y. Ma, X. Zhang, and Q. Zhao, “Dense scale network for crowd counting,” in Proc. Int. Conf. Multimedia Retr., Aug. 2021, pp. 6472–6475.

[47] S. Bai, Z. He, Y. Qiao, H. Hu, W. Wu, and J. Yan, “Adaptive dilated network with self-correction supervision for counting,” in Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit., 2020, pp. 4594–4603.

[48] H. Bai, S. Wen, and S.-H.-G. Chan, “Crowd counting on images with scale variation and isolated clusters,” in Proc. IEEE/CVF Int. Conf. Comput. Vis. Workshop (ICCVW), Oct. 2019, pp. 1–10.

[49] R. Rama Varior, B. Shuai, J. Tighe, and D. Modolo, “Multi-scale attention network for crowd counting,” 2019, arXiv:1901.06026.

[50] J. Dai et al., “Deformable convolutional networks,” in Proc. IEEE/CVF Int. Conf. Comput. Vis. (CVPR), 2017, pp. 764–773.

[51] V. Sindagi and V. Patel, "HA-CCN: Hierarchical attention-based crowd counting network," IEEE Trans. Image Process., vol. 29, pp. 323–335, 2019.

[52] A. Zhang et al., “Relational attention network for crowd counting,” in Proc. IEEE/CVF Int. Conf. Comput. Vis., Jun. 2019, pp. 6788–6797.

[53] A. Zhang et al., “Attentional neural fields for crowd counting,” in Proc. IEEE/CVF Int. Conf. Comput. Vis., 2019, pp. 5714–5723.

[54] Y. Miao, Z. Lin, G. Ding, and J. Han, “Shallow feature based dense attention network for crowd counting,” in Proc. AAAI Conf. Artif. Intell., Apr. 2020, vol. 34, no. 7, pp. 11765–11772.

[55] X. Jiang et al., “Attention scaling for crowd counting,” in Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR), Jun. 2020, vol. 34, pp. 4706–4716.

[56] K. Simonyan and A. Zisserman, “Very deep convolutional networks for large-scale image recognition,” in Proc. Int. Conf. Learn. Represent., 2015, pp. 1–5.

[57] A. Vazwani et al., “Attention is all you need,” in Proc. Adv. Neural Inf. Process. Syst., 2017, pp. 5998–6008.

[58] X. Wang, R. Girshick, A. Gupta, and K. He, “Non-local neural networks,” in Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit., Jun. 2018, pp. 7794–7803.

[59] A. Krizhevsky, I. Sutskever, and G. E. Hinton, “Imagenet classification with deep convolutional neural networks,” in Proc. Adv. Neural Inf. Process. Syst., 2012, pp. 1097–1105.

[60] V. A. Sindagi and V. M. Patel, “CNN-based cascaded multi-task learning of high-level prior and density estimation for crowd counting,” in Proc. 14th IEEE Int. Conf. Adv. Video Signal Based Surrveillance. (AVSS), Aug. 2017, pp. 1–6.

[61] Q. Wang, J. Gao, W. Lin, and Y. Yuan, “Learning from synthetic data for crowd counting in the wild,” in Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit., 2019, pp. 8198–8207.

[62] J. Gao, Q. Wang, and Y. Yuan, “SCAR: Spatial-channel-wise attention regression networks for crowd counting,” Neurocomputing, vol. 363, pp. 1–8, Oct. 2019.

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

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

[65] S. Ren, K. He, R. Girshick, and J. Sun, “Faster R-CNN: Towards real-time object detection with region proposal networks,” IEEE Trans. Pattern Anal. Mach. Intell., vol. 39, no. 6, pp. 1137–1149, Jun. 2017.

[66] T. N. Mundhenk, G. Konjevod, W. A. Sakla, and K. Boakye, “A large contextual dataset for classification, detection and counting of cars with deep learning,” in Proc. Eur. Conf. Comput. Vis., 2016, pp. 785–800.

[67] W. Liu et al., “SSD: Single shot multibox detector,” in Proc. Eur. Conf. Comput. Vis., 2016, pp. 21–37.

[68] J. Redmon and A. Farhadi, “Yolo9000: Better, faster, stronger,” in Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit., 2017, pp. 7263–7271.

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

[70] L. Zeng, X. Xu, B. Cai, S. Qiu, and T. Zhang, “Multi-scale convolutional neural networks for crowd counting,” in Proc. IEEE Int. Conf. Image Process. (ICIP), Sep. 2017, pp. 465–469.

[71] I. H. Laradji, N. Restamzadeh, P. V. Pinheiro, D. Vazquez, and M. Schmidt, “Where are the blobs: Counting by localization with point supervision,” in Proc. Eur. Conf. Comput. Vis., 2018, pp. 547–562.

[72] Z. Shen, Y. Xu, B. Ni, M. Wang, J. Hu, and X. Yang, “Crowd counting via adversarial cross-scale consistency pursuit,” in Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit., Oct. 2018, pp. 5245–5254.

[73] S. Li, H. Li, Q. Wu, X. Chen, and P. Zhou, “Dual path dilated convnet for crowd counting,” in Proc. Conf. Comput. Vis. Pattern Recognit. Workshops, 2018, pp. 195–204.

[74] S. Huang, X. Li, Z.-Q. Cheng, Z. Zhang, and A. Hauptmann, “Stacked pooling: Improving crowd counting by boosting scale invariance,” 2018, arXiv:1808.07456.

[75] X. Jiang et al., “Crowd counting and density estimation by trellis encoder-decoder networks,” in Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR), Jun. 2019, pp. 6133–6142.

[76] 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 Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit., 2019, pp. 8382–8390.

[77] S. Che, J. Guo, K. Li, Z.-J. Zha, and M. Wang, “DADNet: Dilated-attention deformable convnet for crowd counting,” in Proc. 27th ACM Int. Conf. Multimedia, 2019, pp. 1823–1832.

[78] M.-H. Oh, P. Olsen, and K. N. Ramamurthy, “Crowd counting with decomposed uncertainty,” in Proc. AAAI Conf. Artif. Intell., vol. 34, no. 7, 2020, pp. 11799–11806.

[79] A. Luo et al., “Hybrid graph neural networks for crowd counting,” in Proc. AAAI Conf. Artif. Intell., 2020, vol. 34, no. 7, pp. 11693–11700.

[80] L. Liu, Z. Qiu, G. Li, S. Liu, W. Ouyang, and L. Lin, “Crowd counting with deep structured scale integration network,” in Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV), Oct. 2019, pp. 1774–1783.
Zhenghui Hu received the B.S. degree in computer science from the Zhejiang University of Technology, Hangzhou, China, in 2011, and the Ph.D. degree in computer science from Beihang University, Beijing, China, in 2020.

She is currently a Senior Research Associate with the Hangzhou Innovation Institute, Beihang University, Hangzhou. Her research interests include computer vision, hybrid intelligence, and crowdsourcing-based software engineering.

Lu Li received the B.Sc. degree in software engineering and the M.Sc. degree in computer technology from Henan Polytechnic University, Jiaozuo, China, in 2014 and 2017, respectively, and the Ph.D. degree in cartography and geography information system from the Aerospace Information Research Institute, Chinese Academy of Sciences, Beijing, China, in 2020.

He is currently a Post-Doctoral Researcher with the Research Center of Artificial Intelligence, Hangzhou Innovation Institute, Beihang University, Hangzhou, China. His research interests include remote-sensing image analysis, synthetic aperture radar (SAR) image change detection, and deep learning.

Qi Wen received the B.S. degree from the Beijing Institute of Technology, Beijing, China, in 2004, and the Ph.D. degree from the Institute of Remote Sensing Applications, Chinese Academy of Sciences, Beijing, in 2009.

He is currently an Associate Researcher with the National Disaster Reduction Center of China, Beijing. His research interests include remote-sensing analysis and high-resolution remote sensing for disaster reduction.

Yunhong Wang (Fellow, IEEE) received the B.S. degree from Northwestern Polytechnical University, Xi’an, China, in 1989, and the M.S. and Ph.D. degrees from the Nanjing University of Science and Technology, Nanjing, China, in 1995 and 1998, respectively, all in electronics engineering.

She was with the National Laboratory of Pattern Recognition, Institute of Automation, Chinese Academy of Sciences, Beijing, China, from 1998 to 2004. Since 2004, she has been a Professor with the School of Computer Science and Engineering, Beihang University, Beijing, where she is currently the Director of the Laboratory of Intelligent Recognition and Image Processing and the Beijing Key Laboratory of Digital Media. Her research results have published at prestigious journals and prominent conferences, such as the IEEE TRANSACTIONS ON PATTERN ANALYSIS AND MACHINE INTELLIGENCE (TPAMI), IEEE TRANSACTIONS ON IMAGE PROCESSING (TIP), IEEE TRANSACTIONS ON INFORMATION FORENSICS AND SECURITY (TIFS), the Conference on Computer Vision and Pattern Recognition (CVPR), the International Conference on Computer Vision (ICCV), and the European Conference on Computer Vision (ECCV). Her research interests include biometrics, pattern recognition, computer vision, data fusion, and image processing.