IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received April 2, 2021, accepted May 1, 2021, date of publication May 10, 2021, date of current version May 18, 2021.

Digital Object Identifier 10.1109/ACCESS.2021.3078481

A Two-Path Network for Cell Counting

NI JIANG AND FEIHONG YU
Department of Optical Engineering, Zhejiang University, Hangzhou 310027, China
Corresponding author: Feihong Yu (feihong@zju.edu.cn)

ABSTRACT The density map is an effective manner to infer how many cells a cell image contains, and it carries valuable information. However, a fine-grained density map requires rich spatial information to recover the distribution details. In this paper, we propose a cell counting network with two paths, i.e. detail path and context path, which respectively extract spatial details and semantics. The detail path encodes the spatial information with small convolutional kernels. The context path rapidly enlarges the receptive field and extracts multi-scale features with an atrous spatial pyramid pooling. At the end of the two paths, we design a feature fusion module to merge the high-level feature maps from the two paths. To decrease the parameters and computation, we directly upsample the fused feature maps to the input size and decode them to obtain the density map. The proposed model is evaluated on three cell datasets and a popular crowd dataset Shanghaitech Part-B. The experiments illustrated that the proposed model not only achieves superior performance on cell datasets but also generalizes well on the crowd dataset.

INDEX TERMS Cell counting, density map, bilateral path.

I. INTRODUCTION

Object counting, which aims to estimate the number of objects in a scene, is a popular research, and it has a wide range of applications, e.g., crowd counting [1], [2], cell counting [3], [4], and vehicle counting [5], [6]. In recent years, it is popular to count objects by estimating the density map [7], which can reflect the density at each pixel, and the sum over all pixels is the count result. Besides, the density map also provides more supervision information to be trained. Because crowd counting and vehicle counting have plenty of images to learn, many works [8]–[10] tend to design large networks to boost counting performances. However, the large network is prone to be overfitting on small cell datasets. It is necessary to design the cell counting network with fewer training parameters, which is similar to the idea of light-weight models [11], [12]. To predict a high-quality density map, many previous works [4], [13] adopted the U-shape networks to reuse the low-level features to extract spatial details. The low-level feature maps are connected to the decoder, and the combined feature maps are gradually upsampled to the input size. For cell counting, there are two points that should be highlighted: light-weight and well-preserved spatial detail. Inspired by the success of the bilateral segmentation network (BiSeNet) [14] in segmentation, we propose a cell counting network with two paths. The detail path is in charge of extracting spatial information. We use small convolutional kernels to encode the input image. Although the low-level feature maps are rich in spatial information, it contains noise which is harmful for density map estimation. We design a detail guide module using the high-level feature maps to guide the low-level feature maps, and the details targeted on cell regions are highlighted. The context path rapidly enlarges the receptive field and encodes more semantic information. Meanwhile, we use the depthwise separable convolutions [15] to replace the general convolutions, saving more computation resources. In the end, we append an atrous spatial pyramid pooling (ASPP) [16] to gather multi-scale information. To improve the feature representation, we bridge the detail path and context path like BiSeNet V2 [17]. Because many cells are small in size, their information may be discarded in repetitive pooling operations. To recover the information of small cells, we upsample the feature maps to the input size and refine them with the guided detail features to obtain a fine-grained density map. In experiments, we evaluate the proposed network on three different cell datasets [7], [18], [19] and achieve excellent performance. The critical feature maps are visualized to explain how the proposed network learns detail information and context information. Furthermore, we also prove that the proposed network generalizes well on a crowd dataset, Shanghaitech Part-B [1].

The associate editor coordinating the review of this manuscript and approving it for publication was Yuan Zhang.

The main differences between the proposed model and the other state-of-the-art cell counting models [4], [20], [21] are the extractions of detail information and context information are decoupled, which makes the features more representative, and the depthwise separable convolutions make the model light-weight. In summary, our main contributions are summarized as follows:

1) We successfully extend the BiSeNet [14], [17] from the semantic segmentation task to the counting task. The proposed cell counting network makes an excellent performance on cell datasets and a crowd dataset.
2) On the detail path, a detail guide module is designed to guide the low-level feature maps with high-level semantics, which highlights the cell locations.
3) We design a feature fusion module to make the high-level feature maps from both paths connect with each other, which can boost performance.
4) We propose a refinement module to improve the density map and accuracy.

II. RELATED WORK

Since the architecture of our proposed counting network is inspired by BiSeNet, which is a light-weight model, we review the related works about the counting models and the light-weight models in this section.

A. COUNTING MODELS

Traditional methods often count cells by detection, they aim to extract individuals from the background. Take the red blood cells as an example, Maitra *et al*. [22] used Hough Transform to detect cells, and Sharif *et al*. [23] segmented cells by morphological operators. Both the two counting methods relied on the prior knowledge of cells, which limited the application. To count the clustered cells, the distance transform [24] is applied to divide the cluster into separated cells. Zhang *et al*. [25] claimed that the distance transform may fail to segment cells when they overlapped heavily and proposed to segment clustered cells based on the curvature information. Although these methods have made progress, they still cannot cope with the cell image where the cell density is denser. To overcome this issue, density map estimation [7] is proposed. It learns image contents to predict the number of objects as well as the object distribution. The early approaches often design features manually and learn the mapping between the handcrafted features and density values by linear transformation [7], ridge regression [26], random forest [27], etc. Later, researchers realized the convolutional neural network (CNN) has powerful feature representation capability and started to learn the mapping by CNNs. Xie *et al*. [3] explored a simple network to count and detect small cells, and it outperformed previous machine learning approaches [7], [26], [27] on VGG Cells [7]. SAU-Net [4] incorporated a self-attention module and adopted an online batch normalization to be adaptive to small datasets. These works counted cells by integrating over the density map, and there are some works that count cells by detecting cells on the density map. Rad *et al*. [28] proposed a residual dilated U-Net to count blastomere cells of the human embryo. Based on this work, they designed a content-based loss function and built dense connections in the encoder and decoder [20], which contributed to better performance. Pan *et al*. [13] used multi-scale branches to cover cell clumps and robustly detect cells. Zhu *et al*. [29] proposed a fully convolutional network to count cells and evaluated the proposed network both on density map and detection. Besides cell counting, density map estimation also has a wide application in crowd counting. To overcome the scale variations, the multi-column networks [1], [30] and the hierarchical architecture [31] are often used to obtain multi-scale receptive fields. Furthermore, Zou *et al*. came up with different ways [32], [33] to handle the multi-scale problem. In video processing, the temporal information is encoded to boost the counting performance [34]. Recently, researchers also pay attention to the light-weight counting models [35], [36]. Apart from the density map estimation, there are many other counting approaches. Akram *et al*. [37] first detected cells by predicting bounding boxes and then segmented cells in the proposed bounding boxes. Count-ception [21] predicted the number of cells in an image patch and scanned the whole image with a stride to count overall cells. However, it lost many spatial details. Aich and Stavness [38] also regressed the cell number but as well as used the Gaussian activation map to supervise the class activation map, which can suppress false detections and improve the count performance. Marsden *et al*. [39] proposed a counting network that can be adaptive to various visual domains and achieved impressive performance. Unlike the above methods, Lu *et al*. [40] tackled the counting problem by matching the counting object.

B. LIGHT-WEIGHT MODELS

Recently, the problem of how to make a network perform well on the devices with limited computation resources attracts high attention. MobileNet V1 [12] is built on depthwise separable convolutions, which have fewer parameters and computation than the general convolutions. Also, two hyper-parameters are introduced to make the model smaller and faster. ShuffleNet V1 [11] reduces computation by group convolutions with channel shuffle operations and surpasses MobileNet V1. MobileNet V2 [41] follows the depthwise separable convolutions in MobileNet V1 and designs an inverted residual structure with a linear bottleneck, which preserves information as much as possible and improves performance. Based on the analysis of factors for overall runtimes, Ma *et al*. presented four practical guidelines to help design networks efficiently and proposed ShuffleNet V2 [42]. These light-weight models can be applied to different tasks, e.g., image classification [43], object detection [44], and semantic segmentation [45]. In semantic segmentation, each pixel is assigned a class label, so the spatial information is crucial for high accuracy. Yu *et al*. proposed a segmentation network with two paths to respectively preserve spatial details

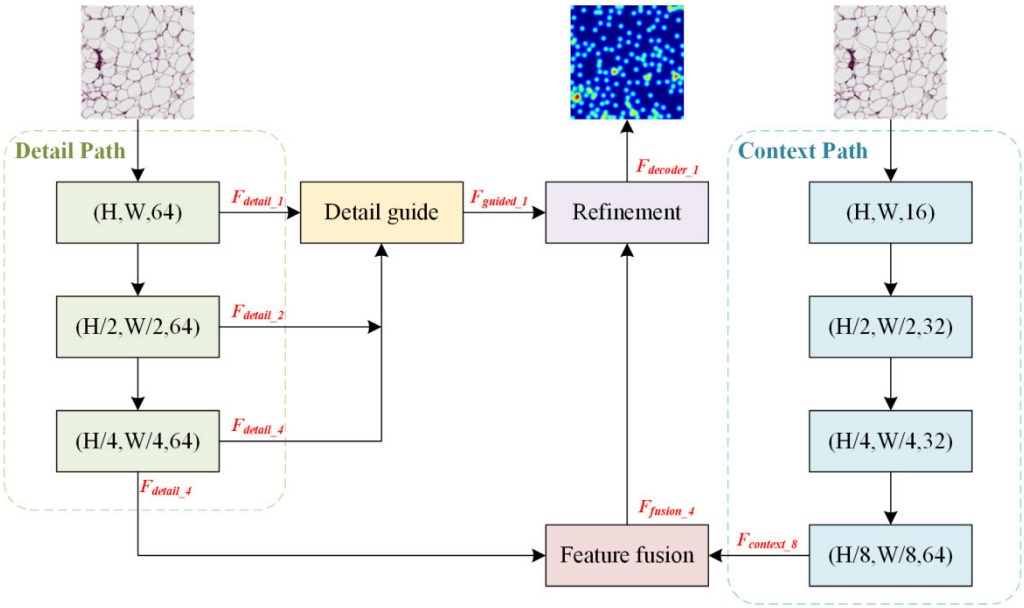

FIGURE 1. An overview of the proposed cell counting network.

and extract context information [14]. The spatial path with wide channels encodes details, and the context path with fast downsampling operations enlarges the receptive field. As an improved version, BiSeNet V2 [17] designs an aggregation layer to efficiently merge the two types of features and improves accuracy and speed.

Similar to semantic segmentation, the density map estimation is a pixel-level task, which needs abundant details to recover the cell locations. Different from semantic segmentation requiring the details such as edges, the density map estimation requires the details about small cell locations which are prone to be lost in pooling operations and cannot be recovered in upsampling operations. In this paper, we propose a counting network based on BiSeNet V1 to cope with the cell datasets.

III. THE PROPOSED MODEL

In this section, we introduce the proposed network. The overall architecture is demonstrated first. Then the submodules are detailed in sequence.

A. THE ARCHITECTURE

The proposed cell counting network consists of two paths, as shown in Fig. 1. The detail path adopts general convolution layers to extract detail features and per stage contains two convolutional layers. Each convolution (Conv) layer is followed by batch normalization (BN) and ReLU. As Fig. 1 indicates, the feature maps are downsampled twice, which is implemented by a convolution layer with stride $= 2$. On the detail path, all convolutional filters keep 3×3 with 64 channels to encode spatial information. Feature maps at different stages are merged to generate a well-guided detail feature map. Compared to the detail path, the context path has deeper layers and narrower channels. It extracts context information with depthwise separable convolutions to reduce computation and enlarge receptive filed by pooling. At the end of the context path, an ASPP is applied to capture multi-scale features. Inspired by the aggregation layer in BiSeNet V2 [17], we design a feature fusion module to bridge the detail path and the context path and merge the two types of feature maps. Different from BiSeNet V1 that the feature maps are directly upsampled to input size to obtain the segmentation result, we use the guided detail feature maps to refine the upsampled feature maps to be adaptive to the counting task.

Depthwise separable convolution consists of a 3×3 depthwise convolution (DW) and a 1×1 pointwise convolution. On the context path, we use a depthwise separable convolution at the first stage and the filter channels in the same stage keep the same. Fig. 2 displays the three types of context blocks (CBs) we design. Fig. 2(a) is the first CB. It is a simple depthwise separable convolution with a residual connection and a 1×1 convolution layer is to merge information. Fig. 2(b) is the CB with a stride, and we name it CB_x2. We place CB_x2 at the first layer per stage, excluding the first stage. Because the input of CB_x2 is from the previous stage, the feature maps from the two branches may be different in channel dimension, we use a concatenation layer to merge features and use a 1×1 filter to decrease dimension. Fig. 2(c) is an ASPP and it is placed at the deepest layer of the context path. The ASPP uses convolution layers with different dilation rates to capture multi-scale receptive fields and adopts a global average pooling (GAP) to obtain global information. The residual connection is also applied to ASPP. The context path can be represented as (Depthwise separable convolution)-(CB_x2-CB)-(CB_x2-CB)-(CB_x2-CB-CB)-(ASPP), where the operations in a bracket belong to the same stage.

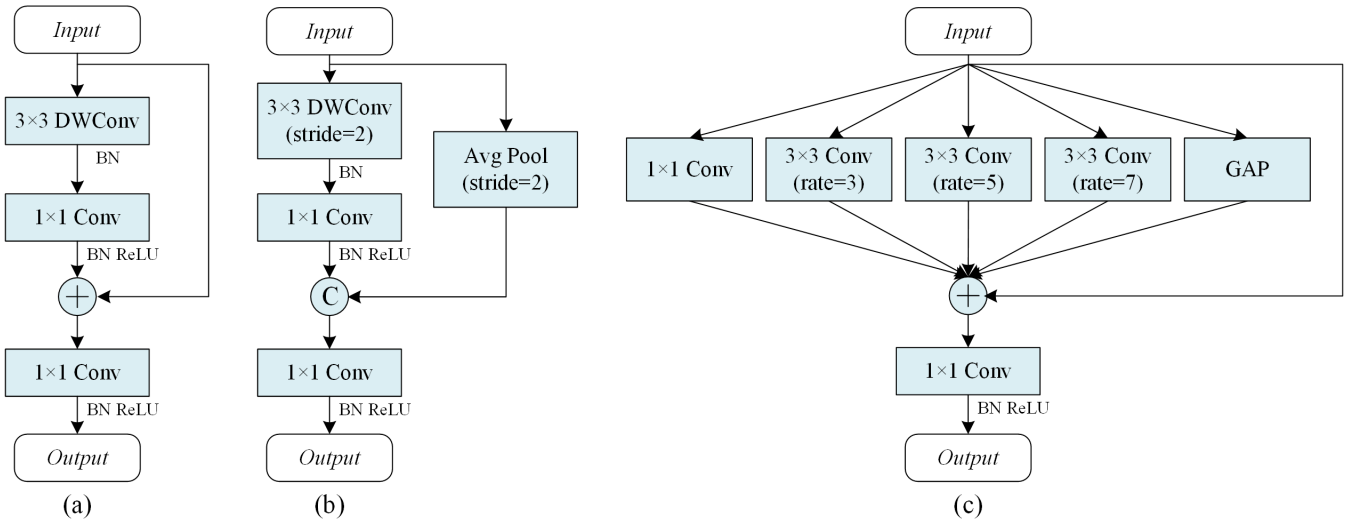

FIGURE 2. Context blocks. (a) Basic context block. (b) context block with downsampling. (c) ASPP.

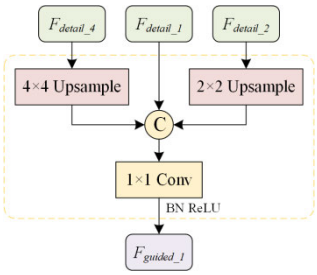

FIGURE 3. Detail guide module.

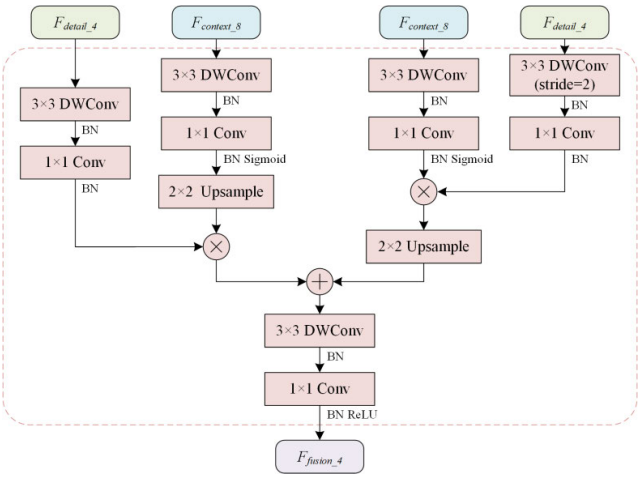

FIGURE 4. Feature fusion module.

B. DETAIL GUIDE MODULE

It is well-known that the low-level features preserve rich spatial information such as edges and lines. However, these details do not help much with the density map estimation. We use a fixed Gaussian kernel to generate ground truth and the label cannot align with the cells of various shapes and sizes. Zhang *et al*. proposed to introduce semantic information to low-level features to guide extracting details [46]. Following this perspective, we design a detail guide module to force the low-level feature maps to pay more attention to the cell locations. The detail guide module is shown in Fig. 3. The low-resolution feature maps with more semantic information are upsampled to the highest resolution and the three feature maps are merged by a 1×1 convolution kernel. Benefit from the fusion, the details evolve from the edges to cell locations under the guidance of high-level semantics, which is proved in experiments.

C. FEATURE FUSION MODULE

BiSeNet V2 [17] proposed a guided aggregation layer to merge the two types of features, and we redesign it to fit our proposed network. Fig. 4 shows the redesigned module. The input detail feature maps are $1/4$ of the original image, and the input context feature maps are $1/8$ of the original image. The context feature maps contain more semantics, which makes it reasonable to use the context feature maps to guide the detail feature maps. The two types of features are fused at both scales, which introduces multi-scale information and also connects the two paths more closely.

D. REFINEMENT MODULE

Some cells are so smaller that their information may be lost after pooling operations. Therefore, it is necessary to reuse the low-level feature maps to recover the lost information. To obtain a fine-grained density map, we need to refine the upsampled feature map carefully. Fig. 5 displays the process of refinement. First, the upsampled feature maps are combined with the guided detail feature maps by addition. Then, a depthwise separable convolution is applied to decode

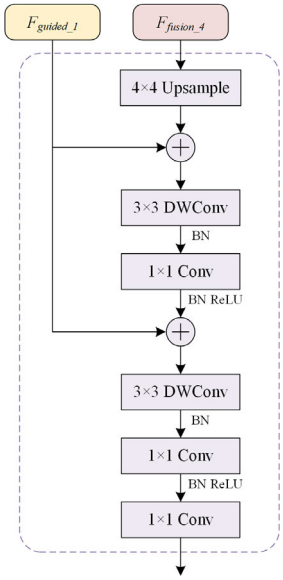

FIGURE 5. Refinement module.

features. Usually, the convolution operation is repeated several times to improve accuracy, we make the spatial details participate in all convolution operations to boost performance.

E. LOSS FUNCTION

In this paper, the density map estimation is a pixel-to-pixel task. We use the Euclidean distance to measure the difference between the estimated density map and the ground truth. The loss function is defined as:

$$L = \frac{1}{N}\sum_{i=1}^{N}\left\|\hat{D}_i - D_i\right\|_2^2 \tag{1}$$

where N is the number of training images, $\hat{D}_i$ is the estimated density map of the i^{th} image, and D_i is the ground truth of the i^{th} image.

IV. EXPERIMENTS

In this section, we first introduce the datasets, implementation details, and evaluation metrics. Then, we compare the proposed cell counting network with state-of-the-art methods. Finally, we conduct the ablation study to prove the effectiveness of each submodule.

A. DATASETS

We experimented on four datasets: VGG Cells [7], MBM Cells [21], Adipocyte Cells [18], and Shanghaitech Part-B [1].

VGG Cells is synthetic and it realistically simulates the cell overlaps, defocused blur, vignetting, etc. It consists of 200 images containing 174 ± 64 cells on average, where 100 images for training and 100 images for testing. MBM Cells collects the healthy human bone marrow cells that are sampled from eight patients. The cell nuclei are dyed blue and sparsely distributed. There are 44 images with an average of 126 ± 33 cells, where 30 images are the training set and the rest are the testing set. Adipocyte Cells contains 200 images and per image averagely contains 165 ± 44 cells, 100 images are used for training and 100 images for testing. The cells are densely adjoined and dramatically varied in shape and size. Shanghaitech Part-B is a crowd dataset, which records the busy streets in Shanghai. It contains 716 images, 400 images for training and 316 images for testing.

B. IMPLEMENTATION DETAILS

Pre-processing. Due to the three pooling operations on the context path, the size of the input image has to be a multiple of 8. The original resolution of MBM Cells is 300×300, and we split the image into 4 patches with a 152×152 resolution. The pixels that are counted repeatedly are averaged as the final results when inference. For Adipocyte Cells, we padded the images from 150×150 to 152×152. Shanghaitech Part-B has a high resolution and we resized the image from 768×1024 to 192×256 to reduce the computational overhead.

Data augmentation. For cell datasets, we randomly applied horizontal shifting, vertical shifting, horizontal flipping, and vertical flipping. Considering human postures in reality, we applied the same augmentations but vertical flipping on the crowd dataset.

Optimization. In our experiments, we adopted the Adam optimizer with a weight decay of $1e^{-03}$. Especially, we set weight decay to $1e^{-04}$ on MBM Cells. The network is optimized with L_2 regularization for 900 epochs. During the first 400 epochs, the learning rate is set to $1e^{-03}$. In the next 400 epochs, the learning rate is decreased to $5e^{-04}$. In the last 100 epochs, the learning rate changes to $1e^{-04}$

C. EVALUATION METRICS

Mean absolute error (MAE) and root mean squared error (RMSE) are the two common metrics to evaluate the counting performance. Following previous works [1]–[5], [21], we calculated MAE in cell counting and both MAE and RMSE in crowd counting. They are defined as:

$$MAE = \frac{1}{N}\sum_{i=1}^{N}\left|\hat{y}_i - y_i\right| \tag{2}$$

$$RMSE = \sqrt{\frac{1}{N}\sum_{i=1}^{N}\left(\hat{y}_i - y_i\right)^2} \tag{3}$$

where, N is the number of testing images, $\hat{y}_i$ and y_i are the predicted object number and the true object number of the i^{th} image, respectively.

D. COMPARISONS WITH THE STATE-OF-THE-ARTS

To validate our proposed method, we compared it with other state-of-the-art methods. Table 1, Table 2, and Table 3 display the comparison results on VGG Cells, MBM Cells, and Adipocyte Cells, respectively. Because cell datasets are small, we randomly and equally selected training data and

TABLE 1. Comparison results on VGG Cells.

Method	$N=16$	$N=32$	$N=50$
Lempitsky *et al*. [7]	3.8 ± 0.2	3.5 ± 0.2	N/A
Fiaschi *et al*. [27]	N/A	3.2 ± 0.1	N/A
FCRN-A [3]	3.4 ± 0.2	2.9 ± 0.2	2.9 ± 0.2
SAU-Net [4]	N/A	N/A	2.6 ± 0.4
Count-ception [21]	2.9 ± 0.5	2.4 ± 0.4	2.3 ± 0.4
Cell-Net [20]	2.7 ± 0.6	N/A	2.2 ± 0.5
The proposed method	2.6 ± 0.2	2.3 ± 0.2	2.2 ± 0.2

TABLE 2. Comparison results on MBM Cells.

Method	$N=5$	$N=10$	$N=15$
Marsden *et al*. [39]	23.6 ± 4.6	21.5 ± 4.2	20.5 ± 3.5
Count-ception [21]	12.6 ± 3.0	10.7 ± 2.5	8.8 ± 2.3
Cell-Net [20]	11.3 ± 4.8	9.8 ± 3.2	N/A
SAU-Net [4]	N/A	N/A	5.7 ± 1.2
The proposed method	8.2 ± 1.1	6.9 ± 0.9	6.0 ± 0.6

TABLE 3. Comparison results on Adipocyte Cells.

Method	$N=10$	$N=25$	$N=50$
Count-ception [21]	25.1 ± 2.9	21.9 ± 2.8	19.4 ± 2.2
SAU-Net [4]	N/A	N/A	14.2 ± 1.6
The proposed method	13.8 ± 0.7	11.6 ± 0.4	10.6 ± 0.3

validation data from the training set and repeated training 10 times. The mean and standard deviation of MAE are reported. N represents the number of images for training. For VGG Cells, our proposed method performed slightly better than Count-ception [21] and Cell-Net [20] and achieved the state-of-the-art. Count-ception regressed the count in a patch and Cell Net paid more attention to feature encoder and decoder, while both of them did not reuse the low-level detail information, which may result in underestimation of small cells. Marsden *et al*. proposed a network that can count different datasets using a shared model [39]. However, it did not perform well on MBM Cells because the number of training images is too small to be learned sufficiently. The cells of MBM Cells are sparse and small, which makes the count errors for Count-ception and Cell-Net higher. SAU-Net [4] is a U-shape network with an attention module. It encodes features with global information and also reused the low-level details. Our proposed method is competitive to it. For Adipocyte Cells, our proposed method performed better than the two compared methods. The Adipocyte cells varies in size and it is necessary to capture the features of multi-scale cells. In our proposed method, we add an ASPP in the context path to learn the multi-scale information and made an excellent performance on Adipocyte Cells.

To further verify the superiority of our proposed model, we compared the model complexity and inference time of different models, as shown in Table 4. It can be observed that our proposed model has the fewest parameters and GFLOPs. Though Count-ception has fewer parameters than SAU-Net, the GFLOPs is higher. It is because there is no pooling operation, which increases the computational overhead. We ran these models on a Tesla T4 GPU and an i5-7300HQ CPU @2.5 GHz, respectively. We recorded the time the model predicting a whole testing set and the inference time is averaged process time per image. The proposed model achieved promised inference speed, especially on CPU. The input data size of VGG Cells testing set is the largest and our proposed time only spent 14.6ms to count an image on average, which outperformed the other two state-of-the-art methods.

To evaluate the generalization of our proposed cell counting model, we also trained the proposed model on pedestrian data and reported the count accuracy. Table 5 shows the comparison results on Shanghaitech Part-B. MCNN [1] is a three-column network and each column has a different receptive field. DecideNet [2] ensembles a detection network and a regression network to respectively handle sparse scenes and dense scenes, and incorporates an attention module to be adaptive to the scene with varying densities. MMCNN [8] has a similar multi-column network to MCNN but with multi-scale input to extract effective features, and it is assigned a density level classification task and a segmentation task to refine the estimated density map. Both MRA-CNN [47] and CAT-CNN [9] take a density level classification task as an auxiliary task to help boost the density map estimation and use an attention map to guide the network focus on head regions. HA-CNN [10] segments foreground and background with supervision to enhance the foreground regions and fuses multi-scale features from different layers. These multi-task networks achieved state-of-the-art accuracy. However, our proposed method is easier to be optimized and performed competitively.

Fig. 6 displays the estimated density maps for four datasets. There are the testing samples from VGG Cells, MBM Cells, Adipocyte Cells, and Shanghaitech Part-B from top to bottom, respectively. The first column lists the testing image. The second column and the third column show the ground truth of the density maps and the estimated density maps, respectively.

E. ABLATION STUDY

In this section, we demonstrate the ablation experiments on Adipocyte Cells to validate the effectiveness of the detail guide module, the feature fusion module, and the refinement module.

1) DETAIL GUIDE VS. NONE DETAIL GUIDE

To extract more helpful detail information, we use the high-level feature maps from detail path to guide the low-level feature maps. The high-level feature maps contain more semantics that is closer to the counting task. Therefore, it can guide the generation of details. Table 6 shows the comparison results. Without the detail guide module, the count error on Adipocyte Cells is increased from 10.6 to 10.9. We also

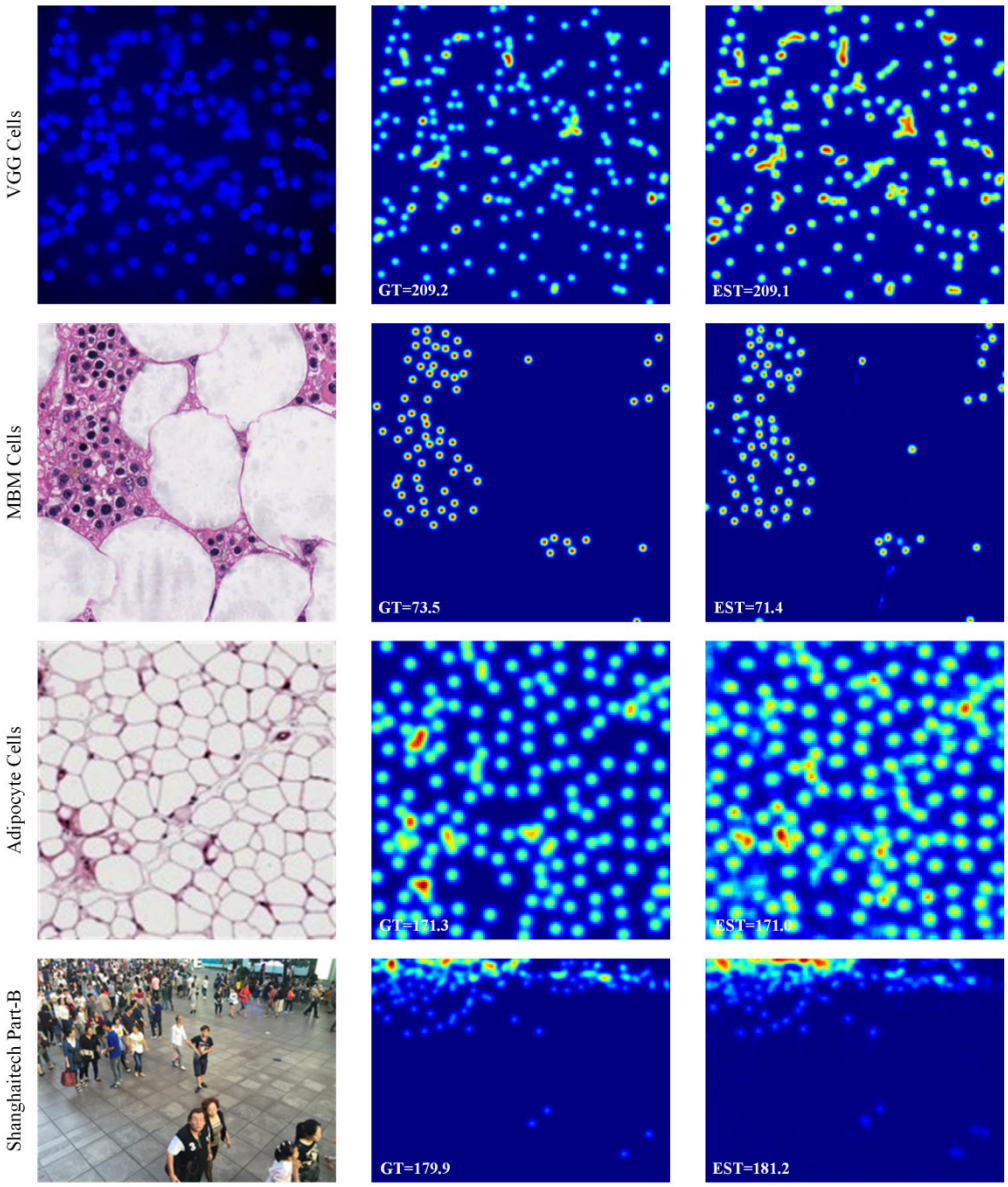

FIGURE 6. Visualization of estimated density maps for four datasets samples.

TABLE 4. The model complexity and inference time of different models.

Method	Params (M)	VGG Cells			MBM Cells			Adipocyte Cells		
		GFLOPs	GPU(ms)	CPU(s)	GFLOPs	GPU(ms)	CPU(s)	GFLOPs	GPU(ms)	CPU(s)
Count-ception [21]	0.99	170.0	29.9	1.47	225.0	45.1	1.83	69.6	12.6	0.60
SAU-Net [4]	2.19	20.0	15.3	0.23	27.3	39.6	0.32	6.8	7.1	0.07
The proposed method	0.39	11.4	14.6	0.17	15.5	58.6	0.29	4.1	9.5	0.07

display visual explanations of each stage detail feature map and the guided detail feature map in Fig. 7, where the name of each feature map has been indicated in Fig. 1. Due to the complexity of texture, the MBM Cells, the Adipocyte Cells, and the Shanghaitech Part-B are discussed. As the layer gets deeper, the features become gradually semantic from F_{detail_1} to F_{detail_4}. After the combination of F_{detail_1}, F_{detail_2}, and F_{detail_4}, the low-level feature maps include more semantics and the individuals can be located well, as F_{guide_1}. explains.

2) FEATURE FUSION VS. SIMPLE FEATURE FUSION

We design a feature fusion module to connect the detail path and the context path effectively. As the comparisons, we combined the detail feature maps and the context feature maps

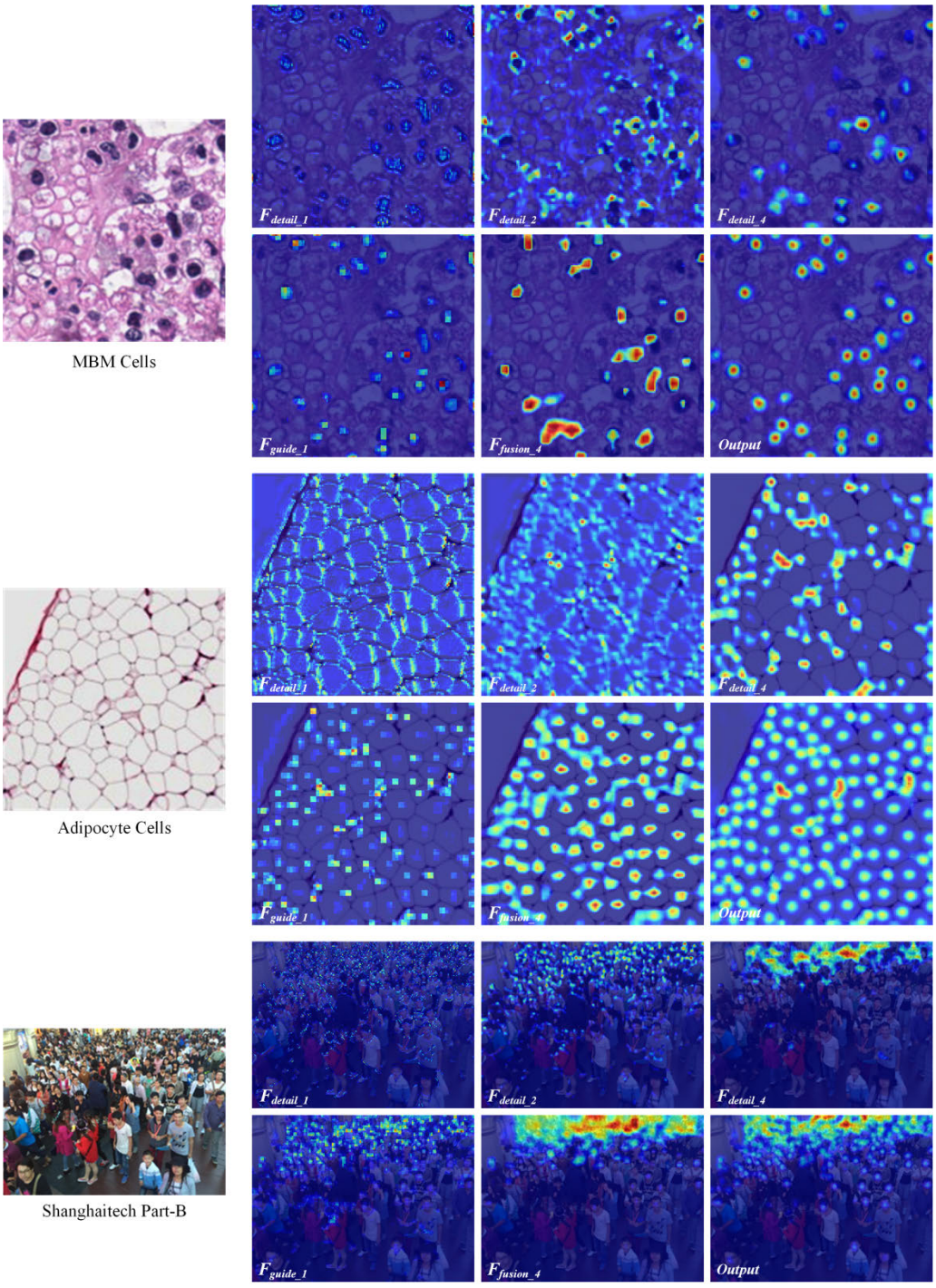

FIGURE 7. The visual explanation of different feature maps.

by a simple operation, e.g., addition or concatenation, and merged them with a 3×3 depthwise separable convolution. Table 7 displays the comparison results and demonstrates the superiority of the feature fusion module.

3) REFINEMENT VS. NONE REFINEMENT

Different from traditional low-level feature maps reusing, we reuse the guided feature maps at each convolution layer to refine the feature maps. To validate its effectiveness, we have tried different reusing strategies, as Table 8 shows. First, we only reused the guided feature maps once and it performed not well. Then we tried to add more convolution layers to boost feature representation. However, it did not improve much. Next, we reused the guided feature maps for different times and the comparison results illustrate that the twice performed best. In Fig. 7, it can be observed that the locations

TABLE 5. Comparison results on Shanghaitech Part-B.

Method	MAE	RMSE
MCNN [1]	26.4	41.3
DecideNet [2]	20.8	29.4
MMCNN [8]	18.5	29.3
MRA-CNN [47]	11.9	21.3
CAT-CNN [9]	11.2	20.0
HA-CNN [10]	8.1	13.4
The proposed method	12.6	21.3

TABLE 6. Detail guide vs. none detail guide.

Method	MAE $\pm$ std
Method w/o detail guide module	10.9 ± 0.3
Method with detail guide module	10.6 ± 0.3

TABLE 7. Feature fusion vs. simple feature fusion.

Method	MAE $\pm$ std
Method with concatenation operation	11.6 ± 0.8
Method with addition operation	11.3 ± 0.6
Method with feature fusion module	10.6 ± 0.3

TABLE 8. Refinement vs. none refinement.

Method	MAE $\pm$ std
Reuse once	11.2 ± 0.6
Reuse once with 1 Conv layer	11.2 ± 0.5
Reuse once with 2 Conv layers	11.1 ± 0.2
Reuse three times	11.0 ± 0.3
Reuse twice	10.6 ± 0.3

of individuals provided by the guided feature maps (F_{guide_1}) are preserved well and helped provide a fine-grained density map.

V. CONCLUSION

In this paper, we proposed a cell counting network with two paths. The detail path aims to provide the location information and the context path encodes the semantic information. First, we leveraged the high-level feature maps to guide the low-level feature maps to make the cell regions focused. Then, we designed a feature fusion module to connect the two paths efficiently. Finally, we suggested reusing the guided detail feature maps at each convolution layer, which can boost the counting performance. Experiments on various datasets and the ablation study demonstrated the effectiveness of our proposed cell counting network.

REFERENCES

[1] 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)*, Las Vegas, NV, USA, Jun. 2016, pp. 589–597.

[2] J. Liu, C. Gao, D. Meng, and A. G. Hauptmann, "DecideNet: Counting varying density crowds through attention guided detection and density estimation," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Salt Lake City, UT, USA, Jun. 2018, pp. 5197–5206.

[3] W. Xie, J. A. Noble, and A. Zisserman, "Microscopy cell counting and detection with fully convolutional regression networks," *Comput. Methods Biomech. Biomed. Eng., Imag. Visualizat.*, vol. 6, no. 3, pp. 283–292, May 2018.

[4] Y. Guo, J. Stein, G. Wu, and A. Krishnamurthy, "SAU-Net: A universal deep network for cell counting," in *Proc. 10th ACM Int. Conf. Bioinf., Comput. Biol. Health Informat.*, Niagara Falls, NY, USA, Sep. 2019, pp. 299–306.

[5] Y. Li, X. Zhang, and D. Chen, "CSRNet: Dilated convolutional neural networks for understanding the highly congested scenes," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Salt Lake City, UT, USA, Jun. 2018, pp. 1091–1100.

[6] S. Zhang, G. Wu, J. P. Costeira, and J. M. F. Moura, "FCN-rLSTM: Deep spatio-temporal neural networks for vehicle counting in city cameras," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Venice, Italy, Oct. 2017, pp. 3687–3696.

[7] V. Lempitsky and A. Zisserman, "Learning to count objects in images," in *Proc. Annu. Conf. Neural Inf. Process. Syst. (NIPS)*, Vancouver, BC, Canada, 2010, pp. 1324–1332.

[8] B. Yang, J. Cao, N. Wang, Y. Zhang, and L. Zou, "Counting challenging crowds robustly using a multi-column multi-task convolutional neural network," *Signal Process., Image Commun.*, vol. 64, pp. 118–129, May 2018.

[9] J. Chen, W. Su, and Z. Wang, "Crowd counting with crowd attention convolutional neural network," *Neurocomputing*, vol. 382, pp. 210–220, Mar. 2020.

[10] V. A. Sindagi and V. M. Patel, "HA-CCN: Hierarchical attention-based crowd counting network," *IEEE Trans. Image Process.*, vol. 29, pp. 323–335, 2020.

[11] X. Zhang, X. Zhou, M. Lin, and J. Sun, "ShuffleNet: An extremely efficient convolutional neural network for mobile devices," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Salt Lake City, UT, USA, Jun. 2018, pp. 6848–6856.

[12] A. G. Howard, M. Zhu, B. Chen, D. Kalenichenko, W. Wang, T. Weyand, M. Andreetto, and H. Adam, "MobileNets: Efficient convolutional neural networks for mobile vision applications," 2017, *arXiv:1704.04861*. [Online]. Available: http://arxiv.org/abs/1704.04861

[13] X. Pan, D. Yang, L. Li, Z. Liu, H. Yang, Z. Cao, Y. He, Z. Ma, and Y. Chen, "Cell detection in pathology and microscopy images with multi-scale fully convolutional neural networks," *World Wide Web*, vol. 21, no. 6, pp. 1721–1743, Nov. 2018.

[14] C. Yu, J. Wang, C. Peng, C. Gao, G. Yu, and N. Sang, "BiseNet: Bilateral segmentation network for real-time semantic segmentation," in *Proc. Eur. Conf. Comput. Vis. (ECCV)*, Munich, Germany, 2018, pp. 325–341.

[15] L. SIfre and S. Mallat, "Rigid-motion scattering for texture classification," 2014, *arXiv:1403.1687*. [Online]. Available: http://arxiv.org/abs/1403.1687

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

[17] C. Yu, C. Gao, J. Wang, G. Yu, C. Shen, and N. Sang, "BiSeNet v2: Bilateral network with guided aggregation for real-time semantic segmentation," 2020, *arXiv:2004.02147*. [Online]. Available: http://arxiv.org/abs/2004.02147

[18] J. Lonsdale, J. Thomas, M. Salvatore, R. Phillips, E. Lo, S. Shad, R. Hasz, G. Walters, F. Garcia, and N. Young, "The genotype-tissue expression (GTEx) project," *Biopreservation Biobanking*, vol. 13, no. 5, pp. 307–308, Oct. 2015.

[19] P. Kainz, M. Urschler, S. Schulter, P. Wohlhart, and V. Lepetit, "You should use regression to detect cells," in *Proc. Int. Conf. Med. Image Comput. Comput.-Assist. Intervent.*, 2015, pp. 276–283.

[20] R. Moradi Rad, P. Saeedi, J. Au, and J. Havelock, "Cell-net: Embryonic cell counting and centroid localization via residual incremental Atrous pyramid and progressive upsampling convolution," *IEEE Access*, vol. 7, pp. 81945–81955, Jun. 2019.

[21] J. P. Cohen, G. Boucher, C. A. Glastonbury, H. Z. Lo, and Y. Bengio, "Count-ception: Counting by fully convolutional redundant counting," in *Proc. IEEE Int. Conf. Comput. Vis. Workshops (ICCVW)*, Venice, Italy, Oct. 2017, pp. 18–26.
[22] M. Maitra, R. K. Gupta, and M. Mukherjee, "Detection and counting of red blood cells in blood cell images using Hough transform," *Int. J. Comput. Appl.*, vol. 53, no. 16, pp. 13–17, Sep. 2012.
[23] J. M. Sharif, M. F. Miswan, M. A. Ngadi, M. S. H. Salam, and M. M. B. A. Jamil, "Red blood cell segmentation using masking and watershed algorithm: A preliminary study," in *Proc. Int. Conf. Biomed. Eng. (ICoBE)*, Feb. 2012, pp. 258–262.
[24] C. Jung and C. Kim, "Segmenting clustered nuclei using H-minima transform-based marker extraction and contour parameterization," *IEEE Trans. Biomed. Eng.*, vol. 57, no. 10, pp. 2600–2604, Oct. 2010.
[25] C. Zhang, C. Sun, R. Su, and T. D. Pham, "Segmentation of clustered nuclei based on curvature weighting," in *Proc. 27th Conf. Image Vis. Comput. New Zealand*, 2012, pp. 49–54.
[26] C. Arteta, V. Lempitsky, J. A. Noble, and A. Zisserman, "Interactive object counting," in *Proc. Eur. Conf. Comput. Vis. (ECCV)*, Zurich, Switzerland, 2014, pp. 504–518.
[27] L. Fiaschi, U. Köthe, R. Nair, and F. A. Hamprecht, "Learning to count with regression forest and structured labels," in *Proc. Int. Conf. Pattern Recognit. (ICPR)*, Tsukuba, Japan, 2012, pp. 2685–2688.
[28] R. M. Rad, P. Saeedi, J. Au, and J. Havelock, "Blastomere cell counting and centroid localization in microscopic images of human embryo," in *Proc. IEEE 20th Int. Workshop Multimedia Signal Process. (MMSP)*, Vancouver, BC, Canada, Aug. 2018, pp. 1–6.
[29] R. Zhu, D. Sui, H. Qin, and A. Hao, "An extended type cell detection and counting method based on FCN," in *Proc. IEEE 17th Int. Conf. Bioinf. Bioeng. (BIBE)*, Herndon, VA, USA, Oct. 2017, pp. 51–56.
[30] D. Onoro-Rubio and R. J. López-Sastre, "Towards perspective-free object counting with deep learning," in *Proc. Eur. Conf. Comput. Vis.*, 2016, pp. 615–629.
[31] Z. Zou, Y. Liu, S. Xu, W. Wei, S. Wen, and P. Zhou, "Crowd counting via hierarchical scale recalibration network," 2020, *arXiv:2003.03545*. [Online]. Available: http://arxiv.org/abs/2003.03545
[32] Z. Zou, Y. Cheng, X. Qu, S. Ji, X. Guo, and P. Zhou, "Attend to count: Crowd counting with adaptive capacity multi-scale CNNs," *Neurocomputing*, vol. 367, pp. 75–83, Nov. 2019.
[33] Z. Zou, X. Su, X. Qu, and P. Zhou, "DA-Net: Learning the fine-grained density distribution with deformation aggregation network," *IEEE Access*, vol. 6, pp. 60745–60756, 2018.
[34] Z. Zou, H. Shao, X. Qu, W. Wei, and P. Zhou, "Enhanced 3D convolutional networks for crowd counting," 2019, *arXiv:1908.04121*. [Online]. Available: http://arxiv.org/abs/1908.04121
[35] P. Wang, C. Gao, Y. Wang, H. Li, and Y. Gao, "MobileCount: An efficient encoder-decoder framework for real-time crowd counting," *Neurocomputing*, vol. 407, pp. 292–299, Sep. 2020.
[36] L. Liu, J. Chen, H. Wu, T. Chen, G. Li, and L. Lin, "Efficient crowd counting via structured knowledge transfer," 2020, *arXiv:2003.10120*. [Online]. Available: http://arxiv.org/abs/2003.10120
[37] S. U. Akram, J. Kannala, L. Eklund, and J. Heikkilä, "Cell segmentation proposal network for microscopy image analysis," in *Proc. Int. Workshop Deep Learn. Med. Image Anal. (DLMIA)*, Athens, Greece, 2016, pp. 21–29.
[38] S. Aich and I. Stavness, "Improving object counting with heatmap regulation," 2018, *arXiv:1803.05494*. [Online]. Available: http://arxiv.org/abs/1803.05494
[39] M. Marsden, K. McGuinness, S. Little, C. E. Keogh, and N. E. O'Connor, "People, penguins and Petri dishes: Adapting object counting models to new visual domains and object types without forgetting," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Salt Lake City, UT, USA, Jun. 2018, pp. 8070–8079.
[40] E. Lu, W. Xie, and A. Zisserman, "Class-agnostic counting," in *Proc. Asian Conf. Comput. Vis. (ACCV)*, Perth, WA, Australia, 2018, pp. 669–684.
[41] M. Sandler, A. Howard, M. Zhu, A. Zhmoginov, and L.-C. Chen, "MobileNetV2: Inverted residuals and linear bottlenecks," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Salt Lake City, UT, USA, Jun. 2018, pp. 4510–4520.
[42] N. Ma, X. Zhang, H.-T. Zheng, and J. Sun, "Shufflenet v2: Practical guidelines for efficient CNN architecture design," in *Proc. Eur. Conf. Comput. Vis. (ECCV)*, Munich, Germany, 2018, pp. 116–131.
[43] F. Saxen, P. Werner, S. Handrich, E. Othman, L. Dinges, and A. Al-Hamadi, "Face attribute detection with MobileNetV2 and NasNet-mobile," in *Proc. 11th Int. Symp. Image Signal Process. Anal. (ISPA)*, Dubrovnik, Croatia, Sep. 2019, pp. 176–180.
[44] N. S. Sanjay and A. Ahmadinia, "MobileNet-tiny: A deep neural network-based real-time object detection for rasberry pi," in *Proc. 18th IEEE Int. Conf. Mach. Learn. Appl. (ICMLA)*, Boca Raton, FL, USA, Dec. 2019, pp. 647–652.
[45] S. Türkmen and J. Heikkilä, "An efficient solution for semantic segmentation: ShuffleNet V2 with Atrous separable convolutions," in *Proc. Scand. Conf. Image Anal. (SCIA)*, 2019, Norrköping, Sweden, pp. 41–53.
[46] Z. Zhang, X. Zhang, C. Peng, X. Xue, and J. Sun, "ExFuse: Enhancing feature fusion for semantic segmentation," in *Proc. Eur. Conf. Comput. Vis. (ECCV)*, Munich, Germany, 2018, pp. 269–284.
[47] Y. Zhang, C. Zhou, F. Chang, and A. C. Kot, "Multi-resolution attention convolutional neural network for crowd counting," *Neurocomputing*, vol. 329, pp. 144–152, Feb. 2019.

NI JIANG received the bachelor's degree in optical information science and technology from the Hefei University of Technology, in 2016. She is currently pursuing the Ph.D. degree in optical engineering with Zhejiang University, Hangzhou, China. Her research interests include image processing and object counting.

FEIHONG YU received the Ph.D. degree from the Engineering Department, Zhejiang University, Hangzhou, China, in 1993. He is currently a Professor with Zhejiang University. His main research interests include optical and digital image processing, microscopic image processing, and optical instrumentation.