



# GCMNet: Gated Cascade Multi-scale Network for Crowd Counting

Jinfang Zheng<sup>1</sup>, Panpan Zhao<sup>1</sup>, Jinyang Xie<sup>1</sup>, Chen Lyu<sup>1,2</sup>, and Lei Lyu<sup>1,2</sup>(✉)

<sup>1</sup> School of Information Science and Engineering, Shandong Normal University,  
Jinan 250358, China

{lvchen,lvlei}@sdnu.edu.cn

<sup>2</sup> Shandong Provincial Key Laboratory for Distributed Computer Software Novel  
Technology, Jinan 250358, China

**Abstract.** With the rapid development of convolutional neural networks, many CNN-based methods have emerged and made promising progress in the field of crowd counting. However, dealing with extremely scale variation remains a challenging but attractive issue. In this paper, we propose an innovative Gated Cascade Multi-scale Network (GCM-Net) to tackle with the issue by taking full advantage of the representation of multi-scale features in a multi-level network. First of all, we implement such an idea by obtaining rich contextual information with a multi-scale contextual information enhancement module. Then, considering the pixel-level image detail information that is lost during the successive feature extraction process, we propose a hopping cascade module to refine this detail information. However, naively refining all the detail information is sub-optimal. Therefore, a gated information selection delivery module is designed to adaptively control the delivery of information between multi-level features. Combined with our proposed module, our method can effectively generate high-quality crowd density maps. The superiority of our method over current methods is demonstrated through extensive experiments on four challenging datasets.

**Keywords:** Crowd counting · Hopping cascade · Gated information · Multi-scale feature · Deep learning

## 1 Introduction

Crowd counting based on computer vision aims at generating high-quality density maps of crowd scenes, thereby calculating the total number of the crowd. It is widely used in public safety and video surveillance. What's more, the proposed methods for crowd counting can be extended to other fields with similar tasks, including traffic control, agricultural monitoring, and cell counting.

With the rapid growth of deep learning, many CNN-based methods have made amazing improvements in crowd counting. However, crowd counting is still a difficult task due to the complexity of the scenes, especially the large scale variation (Fig. 1).



**Fig. 1.** Scale variation in crowd scenes.

In recent years, numerous methods have been proposed to tackle with the problem of scale variation. MCNN [31] uses filters with different sizes to solve the size variation of the human head. CSRNet [12] adopts dilated convolutions as the back-end part to extract deeper features by expanding the receptive fields. Kang et al. [11] propose an adaptive fusion feature pyramid to handle multiple scales. CAN [14] combines multiple receptive fields with different sizes and learns the correct context for each image location.

Although above methods have achieved better performance, there are still some deficiencies to be improved. On the one hand, the crowd scene has large scale variations in size, shape, and location, and using a simple multi-column structure does not effectively extract multi-scale contextual information. On the other hand, features captured by earlier layers in the deep network contain less semantic information, so naively cascading multi-level features in the network does not effectively solve large scale variation.

To this end, we introduce an innovative deep learning framework named Gated Cascade Multi-scale Network (GCMNet) to take full advantage of the representation of multi-scale features. The architecture of GCMNet is shown in Fig. 2. To perform more comprehensive multi-scale representations and overcome the drawbacks of multi-branch structure, we design a multi-scale contextual information enhancement module to capture the global context. We employ four parallel convolutional layers with different filter sizes and combine the features generated by these convolutions. By doing this, the representation capability of the network is greatly improved. In addition, with the successive feature extraction process, a large amount of detail information is lost, so we have integrated various pixel-level detail through a hopping cascade module, thus ensuring the completion of multi-level feature fusion. Furthermore, the utilization of hopping cascade module to integrate multi-level features does not weight the importance of the information contained therein. While a gated information selection delivery module is adopted, we can determine the turn-on and turn-off of information in multi-level features to perform adaptive and effective delivery of useful information.

In summary, the main contributions of our work are as follows:

- We design a multi-scale contextual information enhancement module with multiple different sizes of convolutional filters to extract multi-scale contextual information.
- We put forward a hopping cascade module that cascades multi-level features to reconstruct pixel-level image detail.
- We propose a gated information selection delivery module to adaptively control information delivery between multi-level features.

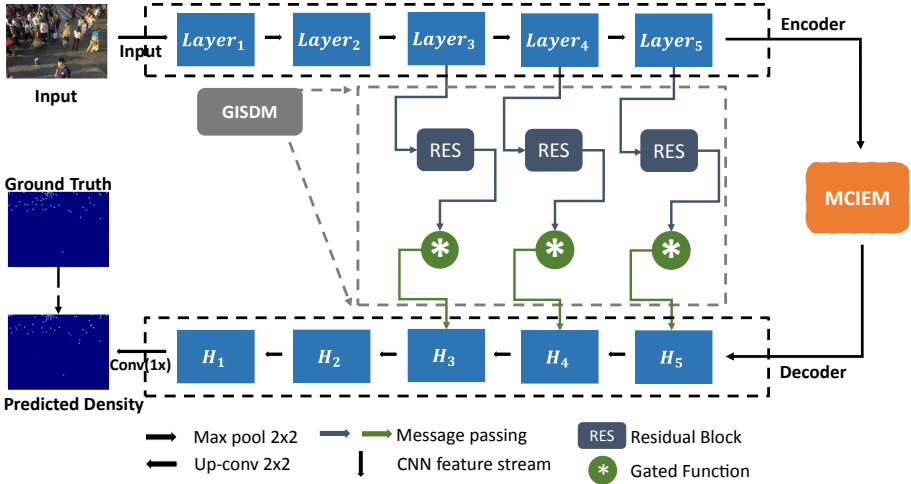


Fig. 2. The overall framework of our GCMNet.

## 2 Related Works

In recent years, significant improvements have been achieved in crowd counting from traditional methods [3, 7] to CNN-based methods [9, 28]. In this paper, we mainly focus on three categories of CNN-based methods: multi-scale feature extraction methods, multi-level feature fusion methods, and feature-wise gated convolution methods.

### 2.1 Multi-scale Feature Extraction Methods

This kind of method aims to address the scale variation in crowd counting with multi-scale contextual information. Zhang et al. [31] propose a multi-column convolutional neural network to extract multi-scale features. Similarly, Sam et al. [20] put forward the Switching-CNN, which uses the density variation to improve the accuracy and localization of crowd counting. Cao et al. propose the SANet [1] for extracting multi-scale features based on the Inception architecture of

encoders. ADCrowdNet [13] combines multi-scale deformable convolution with an attention mechanism to construct a cascade framework. Jiang et al. [10] design a grid coding network that captures multi-scale features by integrating multiple decoding paths. In addition, the spatial pyramid pooling (SPP) [5] uses pooling layers with different sizes to extract multi-scale feature maps and finally aggregates them into a fixed-length vector, thus improving robustness and accuracy. Therefore, it is widely used in SCNet [26], PaDNet [25], and CAN [14] for extracting multi-scale features.

In this paper, we utilize four parallel convolutional layers to extract multi-scale features and fuse features to improve the redundancy arising from the multi-branch structure.

## 2.2 Multi-level Feature Fusion Methods

Several recent works for complex and intensive prediction tasks have demonstrated that features from multiple layers are favorable to produce better results. Deeply encoded features contain semantic information of the object, while shallowly encoded features conserve more spatially detailed information. Several studies on crowd counting [15, 23, 31] have attempted to use features from multi-level convolutional neural networks for more accurate information extraction. Many studies [15, 31] predict the independent results of each stage and finally fuse them to obtain multi-scale information. Sindagi et al. [23] introduce a multi-level bottom-top and top-bottom fusion method to combine shallower information with deeper information.

Different from the above methods, we propose a hopping cascade module to perform multi-level feature fusion with hopping cascade, thereby the pixel-level image details lost during extraction can be regained.

## 2.3 Feature-Wise Gated Convolution Methods

The introduction of gating mechanisms in convolutions has also been extensively studied in language, vision, and speech. Dauphin et al. [2] effectively reduce gradient dispersion by using linear gating units and also retain the ability to be nonlinear. Oord et al. [18] employ a selected-pass mechanism to improve performance and convergence speed. Yu et al. [29] propose an end-to-end gated evolution-based generative image restoration system to improve the restoration of free-form masks and user-guided inputs. WaveNet [17] applies gated activation units to audio sequences to simulate audio signals and obtains better results.

In this study, we propose a gated information selection delivery module to adaptively control the information delivery between multi-level features during the hopping cascade.

## 3 Proposed Algorithm

In this section, we will outline the overall framework of our GCMNet and give a detailed introduction of the theory to realize each module.

### 3.1 Overview of Network Architecture

The overall framework is shown in Fig. 2. Following the practice of most previous work, we adopt VGG-16 [22] as the backbone network and choose the first five stages ( $Layer_1 - Layer_5$ ) of the pre-trained VGG-16 to generate the hopping features at five levels, which are represented as  $F^e = \{f_i^e, i = 1, \dots, 5\}$ . After  $Layer_5$ , we add the Multi-scale Contextual Information Enhancement Module (MCIEM) consisting of multiple convolutional layers with different sizes of filters to capture global context information. Afterwards, to reconstruct the pixel-level image detail information that is lost in the successive feature extraction, we propose the hopping cascade module to cascade the hopping features  $F^e$  with the upsampling features  $F^d = \{f_i^d, i = 1, \dots, 5\}$  generated by upsampling operations. Moreover, we design the Gated Information Selection Delivery Module (GISDM) to control the delivery of the pixel-level image detail information in  $F^e$  with the aim of effectively integrating the multi-level features in the cascade process.

### 3.2 Multi-scale Contextual Information Enhancement Module

It is observed that the output features fused by using parallel convolution contain more image details than the features generated by successive convolution operations. Therefore, we come up with the MCIEM to capture global context information. The module consists of four parallel convolutional layers with filters of different sizes  $k \in \{3, 7, 11, 15\}$  and four max-pooling layers. The details of the MCIEM is given in Fig. 3.

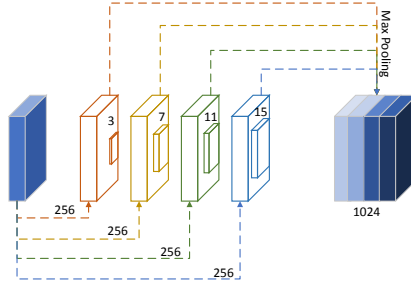


Fig. 3. Details of MCIEM.

Firstly, the multi-level features  $f_5^e$  extracted by the backbone network are taken as the input to the MCIEM. Then the four parallel convolutions with the receptive field of  $3 \times 3$ ,  $7 \times 7$ ,  $11 \times 11$ , and  $15 \times 15$  are used to extract multi-scale features. Finally, these features are fed into a  $2 \times 2$  max-pooling layer and then fused together to extract more comprehensive contextual features. With the MCIEM, multi-scale features can encode richer contextual information.

### 3.3 Hopping Cascade

Though MCIEM can extract effective contextual information through multi-scale features, some pixel-level image detail information is lost in this extraction process. Therefore, we introduce the hopping cascade module to reconstruct the lost pixel-level image detail information.

Specifically, after the MCIEM, we choose the  $H_1 - H_5$  with 32-fold bilinear upsampling operations to generate upsampling features  $F^d = \{f_i^d, i = 1, \dots, 5\}$ . Meanwhile, the lost pixel-level image detail information is reconstructed by cascading  $F^e$  with  $F^d$ . Our cascade module takes the hopping features  $f_3^e, f_4^e, f_5^e$  and upsampling features  $f_3^d, f_4^d, f_5^d$  as input. The cascade process is implemented by the following equation.

$$H_i = ReLU(Conv(f_i^e; \theta)) + ReLU(Conv(f_i^d; \theta)) \tag{1}$$

where  $Conv(*; \theta)$  is a convolutional layer with parameter  $\theta = \{W, b\}$ ,  $ReLU()$  is an activation function.  $f_i^e$  is parallel to the multi-level feature  $f_i^d$  and they have the same size.

### 3.4 Gated Information Selection Delivery Module

The pixel-level image detail information is reconstructed with the hopping cascade module, but not all of the pixel-level detail information contributes to the realization of accurate crowd counting. Therefore, we propose the GISDM to deliver this information from adaptive selection, which consists of a residual block and a gated function, as shown in Fig. 4.

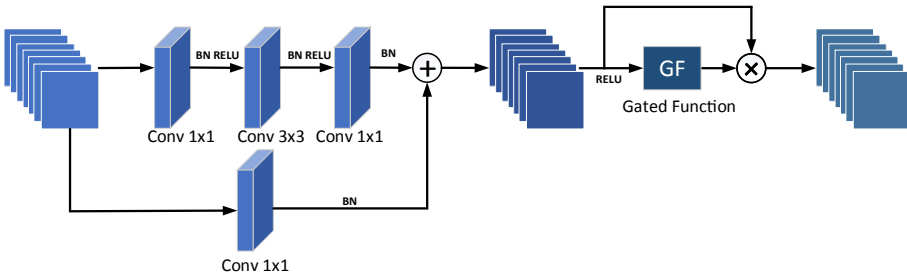


Fig. 4. Details of GISDM.

In our implementation, we feed the hopping features into a residual block to improve the representation ability of hopping features, which is expressed as  $G_i$ :

$$G_i = Res(ReLU(Conv(f_i^e; \theta))) \tag{2}$$

where  $Res(*)$  represents the residual block.

Additionally, we introduce the gated function to further calibrate this information and achieve adaptive delivery of pixel-level detail information instead of indiscriminately delivering all information among multi-level features. The gated function is essentially a convolutional layer with sigmoid activation in the range of  $[0, 1]$ . Let  $GF(x; \theta)$  denotes the gated function:

$$GF(x; \theta) = Sig(Conv(x; \theta)) \quad (3)$$

where  $Sig()$  represents sigmoid function,  $Conv(x; \theta)$  is a  $1 \times 1$  convolutional layer of channels with  $x$ .

With the gated function,  $G_i$  can be rewritten as:

$$G_i = GF(G_i; \theta) \otimes Res(ReLU(Conv(f_i^e; \theta))) \quad (4)$$

where  $\otimes$  represents an element-wise product.

Therefore, the  $H_i$  is summarized as:

$$H_i = Conv(G_i; \theta) + ReLU(Conv(f_i^d; \theta)) \quad (5)$$

where  $G_i$  is the updated features after performing the GISDM.

## 4 Experiments

In this section, we first give the description of the four widely used datasets and the implementation settings. Additionally, we compare our method with state-of-the-art methods by evaluating counting performance and density map quality. Finally, we perform an extensive ablation study to demonstrate the effectiveness of each component of our method.

### 4.1 Datasets

**ShanghaiTech Dataset** [31]. The ShanghaiTech dataset is composed of Part A and Part B datasets. Part A dataset includes 482 images, which are randomly crawled from the Internet and represent highly crowded scenes. It is divided into the training sets and test sets. Part B dataset is acquired from the surveillance cameras of commercial streets, representing relatively sparse scenes, with 400 images in the training sets and 316 images in the test sets.

**UCF\_CCF\_50 Dataset** [7]. The UCF\_CCF\_50 dataset is full of challenges. The training sample is limited and it only collects 50 annotated images of complex scenes from the Internet. These images have a large number of different people, ranging from 94 to 4543. There are a total of 63,974 head annotations, with an average of 1,280 per image.

**UCF-QNRF Dataset** [8]. The dataset contains 1535 high-resolution images with 1,251,642 head annotations, which has more head annotations than the previous datasets. The number of people in each image varies from 49 to 12,865. And the training and test sets have 1,201 and 334 images, respectively.

**WorldExpo’10 Dataset** [30]. This dataset includes 1,132 annotated video sequences collected from 103 different scenes captured by 108 surveillance cameras at the 2010 Shanghai World Expo. There are 3,980 annotated frames with a total of 199,923 annotated pedestrians, of which 3,380 annotated frames are used for model training and the other 600 frames are used for model testing.

## 4.2 Settings

**Ground Truth Generation.** We generate ground truth density maps following the same theory as in MCNN [31]. We use a normalized Gaussian kernel to blur each human head annotation thus generating the ground truth density maps  $F(x)$ .

$$F(x) = \sum_{i=1}^N \delta(x - x_i) \times G_{\sigma_i}(x), \text{ with } \sigma_i = \beta \bar{d}^i \quad (6)$$

where  $N$  represents the number of people in the image,  $x$  is the position of the pixel in the image,  $x_i$  represents the labeled position of the  $i^{\text{th}}$  individual,  $\delta(x - x_i)$  denotes a head annotation at pixel  $x_i$ ,  $G_{\sigma_i}$  represents a Gaussian kernel with standard deviation  $\sigma_i$ , and  $\bar{d}^i$  represents the average distance between  $x_i$  and its nearest  $k$  heads. In our implementation, we set  $\beta = 0.3$  and  $\sigma_i = 3$ .

**Evaluation Metrics.** To evaluate the performance of our method, we adopt the Mean Absolute Error (MAE) and Root Mean Square Error (RMSE), which are denoted as Eq. (7) and Eq. (8), respectively.

$$MAE = \frac{1}{N} \sum_{i=1}^N |C_i^{ES} - C_i^{GT}| \quad (7)$$

$$RMSE = \sqrt{\frac{1}{N} \sum_{i=1}^N (C_i^{ES} - C_i^{GT})^2} \quad (8)$$

where  $N$  is the total number of the test images,  $C_i^{ES}$  and  $C_i^{GT}$  are the estimated and ground-truth counts of the  $i^{\text{th}}$  image, respectively.

MAE and RMSE determine the accuracy and the robustness of the crowd counting, respectively. The lower their values, the better performance of the count results.

In addition, the Peak Signal-to-Noise Ratio (PSNR) and Structural Similarity (SSIM) in images are exploited to evaluate the quality of the output density maps.

The PSNR is defined as:

$$PSNR = 10 \times \log_{10} \left( \frac{MAX_I^2}{MSE} \right) \quad (9)$$

where  $MAX_I$  is the maximum possible pixel value of the images.



$$SSIM(x, y) = \frac{(2\mu_x\mu_y + C_1)(2\sigma_{xy} + C_2)}{(\mu_x^2 + \mu_y^2 + C_1)(\sigma_x^2 + \sigma_y^2 + C_2)} \quad (10)$$

where  $\mu_x$  and  $\mu_y$  denote the mean values of images  $x$  and  $y$ , respectively.  $\sigma_x$  and  $\sigma_y$  denote the variance of images  $x$  and  $y$ , respectively.  $\sigma_{xy}$  is the covariance of images  $x$  and  $y$ .  $C_1$  and  $C_2$  are two constants and defined as:

$$\begin{cases} C_1 = (K_1 \times L)^2 \\ C_2 = (K_2 \times L)^2 \end{cases} \quad (11)$$

where  $K_1 = 0.01$ ,  $K_2 = 0.03$ ,  $L = 255$ .

PSNR essentially represents the error between the corresponding pixels. The higher its value, the better the quality of the density map. SSIM measures the similarity between the predicted density map and the ground truth in terms of brightness, contrast and structure. The higher its value, the smaller the image distortion.

**Implementation Details.** We utilize the pre-trained VGG-16 to initialize the parameters of the first five stages of our model, and parameters of the other convolutional layers are initialized randomly using a Gaussian distribution with  $\delta = 0.01$ . Both upsampling and downsampling operations are simulated using bilinear interpolation. We use Adam optimizer to train our network for 200 epochs, and the learning rate is initially set to  $1e-5$ . And the network is trained by minimizing the Euclidean distance between the estimated density map and the ground truth. The loss function is defined as:

$$L(\Theta) = \frac{1}{2N} \sum_{i=1}^N \|F(X_i; \Theta) - D_i\|_2^2 \quad (12)$$

where  $N$  is the number of training images,  $X_i$  is the  $i^{th}$  input image,  $F(X_i; \Theta)$  denotes the estimated density map,  $D_i$  represents the ground truth density map.

### 4.3 Comparisons with the State-of-the-Art

**ShanghaiTech.** We compare our method with several state-of-the-art methods and the comparison results are listed in Table 1. On Part A, our method obtains the MAE improvement by 4.28% and RMSE improvement by 4.46% compared to the second-best result. On Part B, our method achieves the MAE and RMSE improvements by 4.31% and 4.61%, respectively, compared to the second-best result.

**UCF\_CC\_50.** The UCF\_CC\_50 dataset has a huge challenge and we evaluate our method according to 5-fold cross-validation [12]. As shown in Table 1, we compare our method with the current state-of-the-art methods. Our method has a very significant improvement, with MAE and RMSE improved by 18.57% and 18.69%, respectively, compared to the latest CFANet method. Despite the limited training samples, our method converges well in this dataset.

**Table 1.** Comparisons of GCMNet and state-of-the-art methods on three datasets.

Dataset	Part A		Part B		UCF_CC_50		UCF-QNRF	
Method	MAE	RMSE	MAE	RMSE	MAE	RMSE	MAE	RMSE
MCNN [30]	110.2	173.2	26.4	41.3	377.6	509.1	277.0	426.0
CSRNet [12]	68.2	115.0	10.6	16.0	266.1	397.5	120.3	208.5
TEDNet [10]	64.2	109.1	8.2	12.8	249.4	354.5	113	188
BL [16]	62.8	101.8	7.7	12.7	229.3	308.2	88.7	154.8
ASNet [9]	57.78	90.13	–	–	174.84	251.63	91.59	159.71
AMSNet [6]	56.7	93.4	6.7	10.2	208.4	297.3	101.8	163.2
AMRNet [15]	61.59	98.36	7.02	11.00	184.0	265.8	86.6	152.2
CFANet [19]	56.1	89.6	6.5	10.2	203.6	287.3	89.0	152.3
<b>GCMNet(ours)</b>	<b>53.7</b>	<b>85.6</b>	<b>6.22</b>	<b>9.73</b>	<b>165.8</b>	<b>233.6</b>	<b>84.7</b>	<b>148.1</b>

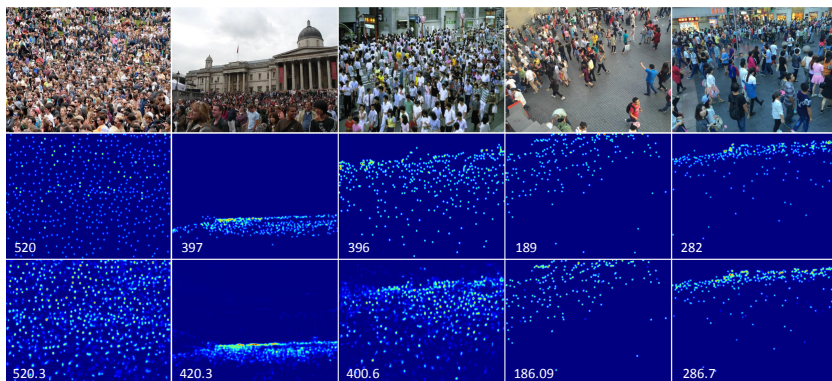
**UCF-QNRF.** Table 1 shows the MAE and RMSE of our method as well as the state-of-the-art methods on UCF-QNRF dataset. The proposed method is compared with eight methods. It can be observed that the proposed method is able to yield the best performance on this dataset. The MAE exceeds the second-best method by 2.19% and RMSE improves over the second-best method by 2.69%.

**WorldExpo’10.** Our method is compared with six state-of-the-art methods. In Table 2, we give the comparison results of MAE for each scene. Our proposed method obtains the best performance in scene 1 (sparse crowd S1), scene 4 (dense crowd S4). Moreover, the best average MAE performance is also achieved.

**Table 2.** Comparisons of GCMNet and state-of-the-art methods on WorldExpo’10.

Dataset	WorldExpo’10					
Method	S1	S2	S3	S4	S5	Ave
TEDNet [10]	2.3	10.1	11.3	13.8	2.6	8.0
ADCrowdNet [13]	1.6	13.2	8.7	10.6	2.6	7.3
CAN [14]	2.9	12.0	10.0	7.9	4.3	7.4
PGCNet [27]	2.5	12.7	8.4	13.7	3.2	8.1
RPNNet [28]	2.4	10.2	9.7	11.5	3.8	8.2
ASNet [9]	2.22	10.11	8.89	7.14	4.84	6.64
<b>GCMNet (ours)</b>	<b>1.43</b>	<b>10.22</b>	<b>8.47</b>	<b>7.04</b>	<b>2.84</b>	<b>6.00</b>

In this section, we first conduct experiments on four datasets and then compare our model quantitatively with several state-of-the-art methods. It is clearly seen from the results that our method achieves the best performance on ShanghaiTech, UCF\_CC\_50 and UCF-QNRF datasets, and outperforms some of the



**Fig. 5.** Sample results of the GCMNet on ShanghaiTech dataset. The first row shows the samples of the input image. The second row shows the ground truth for each sample while the third row presents the density map generated by GCMNet. The number in each density map denotes the count number.

current state-of-the-art methods on WorldExpo'10 dataset. And the predicted density maps on ShanghaiTech dataset is also given and compared with the ground truth, as shown in Fig. 5. It can be obviously seen from the figures that our method is advanced for crowd counting in different scenes. Regardless of highly crowded or sparse crowd counting scenes, we effectively address the scale variation in crowd counting. Our method effectively uses multi-scale features for accurate crowd counting.

#### 4.4 Comparison of Density Map Quality

In this section, we compare our method with other representative methods: MCNN, CP-CNN, CSRNet, CFF and SCAR in PSNR and SSIM.

**Table 3.** Comparisons of PSNR and SSIM of GCMNet and representative methods on ShanghaiTech Part A.

Method	PSNR	SSIM
MCNN [31]	21.4	0.52
CP-CNN [24]	21.72	0.72
CSRNet [12]	23.79	0.76
CFF [21]	25.4	0.78
SCAR [4]	23.93	0.81
<b>GCMNet(ours)</b>	<b>28.66</b>	<b>0.84</b>

As shown in Table 3, our GCMNet achieves the highest SSIM and PSNR. In particular, we get PSNR of 28.66 and SSIM of 0.84 on ShanghaiTech Part A

dataset. Compared with SCAR, the PSNR and SSIM are improved by 19.77% and 3.70%, respectively. The results show that our method has a significant advantage in generating high-quality density maps.

#### 4.5 Ablation Study

In this section, we conduct ablation study on ShanghaiTech dataset to verify the effectiveness of each module in our network and analyze the impact of different network combinations on the counting performance.

**Table 4.** Results of ablation study on ShanghaiTech Part A and Part B datasets.

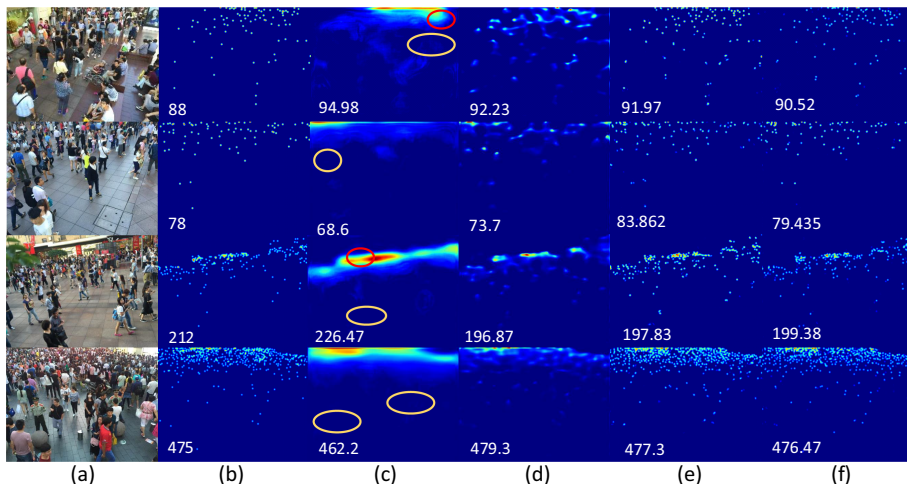
Dataset	Part A		Part B	
Configuration	MAE	RMSE	MAE	RMSE
VGG-16	78.3	120.1	18.3	22.9
VGG-16+MCIEM	66.8	102.3	14.7	17.9
VGG-16+MCIEM+Hopping Cascade	57.1	90.7	8.5	11.6
<b>VGG-16+MCIEM+Hopping Cascade+GISDM</b>	<b>53.7</b>	<b>88.6</b>	<b>6.22</b>	<b>9.73</b>

We use four different combinations to test our model:

- (1) VGG-16: VGG-16 first 13-layer network with 32-fold upsampling operations at the end.
- (2) VGG-16+MCIEM: VGG-16 first 13-layer network with MCIEM for extracting multi-scale contextual information and 32-fold upsampling operations at the end.
- (3) VGG-16+MCIEM+Hopping Cascade: VGG-16 first 13-layer network with MCIEM for extracting multi-scale contextual information and hopping cascade module for cascading the hopping features  $f_3^e$ ,  $f_4^e$ ,  $f_5^e$  with the upsampling features  $f_3^d$ ,  $f_4^d$ ,  $f_5^d$ .
- (4) VGG-16+MCIEM+Hopping Cascade+GISDM: our proposed method.

We give the experimental results of ablation study in Table 4. It can be seen that directly using VGG-16 for feature extraction does not necessarily yield the best performance. After injecting MCIEM into the network for multi-scale feature extraction, the counting error is greatly reduced compared to the previous stage. Further improvements are made by adding the hopping cascade module, and the results show that, as with MCIEM, the performance of the model is substantially improved and the counting error is substantially reduced. Finally, the embedded GISDM adaptively performs information delivery, which further optimizes the effect of crowd counting. In conclusion, our proposed final model achieves the best performance and further accuracy in estimating the crowd. Each of the structures added to our model is effective and complementary to each other. The counting results are significantly better in the case of both high-density and low-density scenes. Figure 6 gives the stage density maps of the ShanghaiTech Part B

dataset during the ablation study, and it is observed that our final model improves on the previous missing (yellow circles) and redundant (red circles) counts, effectively addressing the problem of scale variation. Our model achieves accurate density estimation and produces high-quality density maps.



**Fig. 6.** Stage results of ablation study on ShanghaiTech Part B dataset. (a) Input image, (b) Ground Truth, (c) Baseline (VGG-16), (d) VGG-16+MCIEM, (e) VGG-16+MCIEM+Hopping Cascade, (f) Ours. The number in each density map denotes the count number. The yellow and red circles label the missing and redundant counts of the Baseline method, respectively.

## 5 Conclusion

This paper proposes a novel end-to-end Gated Cascade Multi-scale Network (GCMNet), which effectively solves the problem of rapid scale variation in crowd counting. With the MCIEM, our GCMNet can capture global context at multiple scales. Then we introduce a hopping cascade module to make full use of the pixel-level image detail information. Subsequently, we design a GISDM to selectively integrate multi-level features by adaptively delivering valid information. Finally, the multi-level features are used to generate the final density maps. Extensive experimental results on four datasets show that our GCMNet is superior under different evaluation metrics. In the future, we will explore better methods to perform multi-scale feature extraction and effective integration of multi-level features.

**Acknowledgement.** This work is supported by the National Natural Science Foundation of China (61976127).

## References

1. Cao, X., Wang, Z., Zhao, Y., Su, F.: Scale aggregation network for accurate and efficient crowd counting. In: Ferrari, V., Hebert, M., Sminchisescu, C., Weiss, Y. (eds.) ECCV 2018. LNCS, vol. 11209, pp. 757–773. Springer, Cham (2018). [https://doi.org/10.1007/978-3-030-01228-1\\_45](https://doi.org/10.1007/978-3-030-01228-1_45)
2. Dauphin, Y.N., Fan, A., Auli, M., Grangier, D.: Language modeling with gated convolutional networks. In: International Conference on Machine Learning, pp. 933–941. PMLR (2017)
3. Dollar, P., Wojek, C., Schiele, B., Perona, P.: Pedestrian detection: an evaluation of the state of the art. *IEEE Trans. Pattern Anal. Mach. Intell.* **34**(4), 743–761 (2011)
4. Gao, J., Wang, Q., Yuan, Y.: SCAR: spatial-/channel-wise attention regression networks for crowd counting. *Neurocomputing* **363**, 1–8 (2019)
5. He, K., Zhang, X., Ren, S., Sun, J.: Spatial pyramid pooling in deep convolutional networks for visual recognition. *IEEE Trans. Pattern Anal. Mach. Intell.* **37**(9), 1904–1916 (2015)
6. Hu, Y., et al.: NAS-count: counting-by-density with neural architecture search. In: Vedaldi, A., Bischof, H., Brox, T., Frahm, J.-M. (eds.) ECCV 2020. LNCS, vol. 12367, pp. 747–766. Springer, Cham (2020). [https://doi.org/10.1007/978-3-030-58542-6\\_45](https://doi.org/10.1007/978-3-030-58542-6_45)
7. Idrees, H., Saleemi, I., Seibert, C., Shah, M.: Multi-source multi-scale counting in extremely dense crowd images. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 2547–2554 (2013)
8. Idrees, H., et al.: Composition loss for counting, density map estimation and localization in dense crowds. In: Ferrari, V., Hebert, M., Sminchisescu, C., Weiss, Y. (eds.) ECCV 2018. LNCS, vol. 11206, pp. 544–559. Springer, Cham (2018). [https://doi.org/10.1007/978-3-030-01216-8\\_33](https://doi.org/10.1007/978-3-030-01216-8_33)
9. Jiang, X., et al.: Attention scaling for crowd counting. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 4706–4715 (2020)
10. Jiang, X., et al.: Crowd counting and density estimation by trellis encoder-decoder networks. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 6133–6142 (2019)
11. Kang, D., Chan, A.B.: Crowd counting by adaptively fusing predictions from an image pyramid. In: British Machine Vision Conference 2018, BMVC 2018, Newcastle, UK, 3–6 September 2018, p. 89 (2018)
12. Li, Y., Zhang, X., Chen, D.: CSRNet: dilated convolutional neural networks for understanding the highly congested scenes. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 1091–1100 (2018)
13. Liu, N., Long, Y., Zou, C., Niu, Q., Pan, L., Wu, H.: ADCrowdNet: an attention-injective deformable convolutional network for crowd understanding. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 3225–3234 (2019)
14. Liu, W., Salzmann, M., Fua, P.: Context-aware crowd counting. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 5099–5108 (2019)
15. Liu, X., Yang, J., Ding, W., Wang, T., Wang, Z., Xiong, J.: Adaptive mixture regression network with local counting map for crowd counting. In: Vedaldi, A., Bischof, H., Brox, T., Frahm, J.-M. (eds.) ECCV 2020. LNCS, vol. 12369, pp. 241–257. Springer, Cham (2020). [https://doi.org/10.1007/978-3-030-58586-0\\_15](https://doi.org/10.1007/978-3-030-58586-0_15)

16. Ma, Z., Wei, X., Hong, X., Gong, Y.: Bayesian loss for crowd count estimation with point supervision. In: Proceedings of the IEEE/CVF International Conference on Computer Vision, pp. 6142–6151 (2019)
17. van den Oord, A., et al.: WaveNet: a generative model for raw audio. In: The 9th ISCA Speech Synthesis Workshop, Sunnyvale, CA, USA, 13–15 September 2016 (2016)
18. Oord, A.v.d., Kalchbrenner, N., Vinyals, O., Espeholt, L., Graves, A., Kavukcuoglu, K.: Conditional image generation with pixelCNN decoders, pp. 4790–4798 (2016)
19. Rong, L., Li, C.: Coarse-and fine-grained attention network with background-aware loss for crowd density map estimation. In: Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision, pp. 3675–3684 (2021)
20. Sam, D.B., Surya, S., Babu, R.V.: Switching convolutional neural network for crowd counting. In: 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 4031–4039. IEEE (2017)
21. Shi, Z., Mettes, P., Snoek, C.G.: Counting with focus for free. In: Proceedings of the IEEE/CVF International Conference on Computer Vision, pp. 4200–4209 (2019)
22. Simonyan, K., Zisserman, A.: Very deep convolutional networks for large-scale image recognition. In: 3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, 7–9 May 2015, Conference Track Proceedings (2015)
23. Sindagi, V., Patel, V.M.: Multi-level bottom-top and top-bottom feature fusion for crowd counting. In: 2019 IEEE/CVF International Conference on Computer Vision (ICCV), pp. 1002–1012 (2019)
24. Sindagi, V.A., Patel, V.M.: Generating high-quality crowd density maps using contextual pyramid CNNs. In: Proceedings of the IEEE International Conference on Computer Vision, pp. 1861–1870 (2017)
25. Tian, Y., Lei, Y., Zhang, J., Wang, J.Z.: PaDNet: pan-density crowd counting. *IEEE Trans. Image Process.* **29**, 2714–2727 (2019)
26. Wang, Z., Xiao, Z., Xie, K., Qiu, Q., Zhen, X., Cao, X.: In defense of single-column networks for crowd counting. In: British Machine Vision Conference 2018, BMVC 2018, Newcastle, UK, 3–6 September 2018, p. 78 (2018)
27. Yan, Z., et al.: Perspective-guided convolution networks for crowd counting. In: Proceedings of the IEEE/CVF International Conference on Computer Vision, pp. 952–961 (2019)
28. Yang, Y., Li, G., Wu, Z., Su, L., Huang, Q., Sebe, N.: Reverse perspective network for perspective-aware object counting. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 4374–4383 (2020)
29. Yu, J., Lin, Z., Yang, J., Shen, X., Lu, X., Huang, T.S.: Free-form image inpainting with gated convolution. In: Proceedings of the IEEE/CVF International Conference on Computer Vision, pp. 4471–4480 (2019)
30. Zhang, C., Li, H., Wang, X., Yang, X.: Cross-scene crowd counting via deep convolutional neural networks. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 833–841 (2015)
31. Zhang, Y., Zhou, D., Chen, S., Gao, S., Ma, Y.: Single-image crowd counting via multi-column convolutional neural network. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 589–597 (2016)