Research Article

Multi-Scale Guided Attention Network for Crowd Counting

Pengfei Li , Min Zhang , Jian Wan, and Ming Jiang

Hangzhou Dianzi University, Baiyang Road #2, Hangzhou, China

Correspondence should be addressed to Min Zhang; hz_andy@163.com

Received 29 January 2021; Revised 1 September 2021; Accepted 16 September 2021; Published 1 October 2021

Academic Editor: Qianchuan Zhao

Copyright © 2021 Pengfei Li et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

The CNN-based crowd counting method uses image pyramid and dense connection to fuse features to solve the problems of multiscale and information loss. However, these operations lead to information redundancy and confusion between crowd and background information. In this paper, we propose a multi-scale guided attention network (MGANet) to solve the above problems. Specifically, the multilayer features of the network are fused by a top-down approach to obtain multiscale information and context information. The attention mechanism is used to guide the acquired features of each layer in space and channel so that the network pays more attention to the crowd in the image, ignores irrelevant information, and further integrates to obtain the final high-quality density map. Besides, we propose a counting loss function combining SSIM Loss, MAE Loss, and MSE Loss to achieve effective network convergence. We experiment on four major datasets and obtain good results. The effectiveness of the network modules is proved by the corresponding ablation experiments. The source code is available at <https://github.com/lpfworld/MGANet>.

1. Introduction

Crowd counting can count the number of people in images or video frames to realize the effective management of different scenes such as meetings and sports events. It has been widely used in public security, intelligent traffic, video surveillance, etc. [1–5]. Crowd counting can also be used to count the number of cells, viruses, and animals, extending the field of research into medical and behavioral science [6]. However, it is still a challenging task in the field of computer vision due to the problems of crowd occlusion, scale variation, uneven data distribution, and so forth (see Figure 1).

With the excellent performance of convolutional neural networks in various fields of computer vision, researchers have made various attempts on crowd counting. Some researchers tried to use multicolumn structures with different convolution kernels to solve the problem of multiscale [7–10]. Other researchers used parallel convolution kernels to obtain feature maps with different scales or fused multiscale information by the dense connection of multilayer features [11, 12]. However, in these similar structures, the features learned from different branches have greater repeatability, which makes little contribution to the extraction

of multiscale information. At the same time, these redundant features cause crowd density maps to be disturbed when generated and make background or other image content easily mistaken for the crowd [12, 13]. Some networks are tried to use spatial attention in the training process to emphasize the crowd in images, to solve the problems such as background interference [13, 14].

The feature maps of different layers in the network contain different scale features and semantic features, so we cascade the features of different layers from top to bottom. The low-level network contains more detailed information, which is conducive to the formation of a density map of high-density scenes, while the high-level network contains more semantic information, which is conducive to the distinction between the human head and background noise. In this way, different scales of information can be obtained without increasing the complexity of the network structure. In comparison, most crowd counting networks use a down-top multilayer fusion and then use deconvolution or dilated convolution to keep the final density map scale unchanged. But in our top-down network structure, we do not need to consider how to recover the final density map scale. We cascade the multilayer features of the network but not

FIGURE 1: Samples from the ShanghaiTech dataset. (a) The first row is the images. (b) The second row is the corresponding ground truth density map.

overlay the multilayer feature maps of the network through channels. Cascading operations at different layers can effectively prevent information loss so that the final density map contains complete context information.

The fusion of features of different layers can obtain rich information, but a further selection is needed before generating the density map. Most networks use spatial attention mechanisms to selectively strengthen the human head area [15–17]. However, the channel information contains corresponding to the special class. In the crowd image, the classes are head and background. Therefore, we considered optimizing the resulting features by combining spatial attention with channel attention.

MSE loss is the most commonly used loss function in crowd counting methods. However, in crowd scenes, the texture features and pixel correlation of high-density areas and low-density areas are different. MSE Loss is based on the assumption of pixel independence and ignores the local correlation of the density map. Besides, MSE Loss does not take into account global counting errors of input images. To optimize the loss function, we combine SSIM Loss, MAE Loss, and MSE Loss as the final loss. The loss function can calculate the local consistency of the predicted density map and the ground truth density map and calculate the difference between the predicted number of people and the real number of people so that the network can better converge to generate the high-quality density map.

In this paper, we propose a novel network for crowd counting. The network integrates the multiscale features of the crowd from top to bottom and uses spatial attention mechanisms and channel attention mechanisms to further guide the features to produce high-quality density maps. This network is called MGANet, and its structure is shown in Figure 2. Specifically, our network consists of two parts: MFE (Multiscale Feature Extraction) and MFG (Multiscale Feature Guide). MFE uses the VGG16 backbone to extract the multiscale features and

fuse the multilayer features step by step in a top-down manner, which can not only get the semantic features contained in different levels but also well express the detailed features of people of different scales. MFG differentiates the head information and background information through channel attention, locates the head area more accurately through spatial attention mechanism, and uses the output of the two kinds of attention to form an effective density map. MFG contains three columns of feature outputs with different resolutions, which are converted into density maps by Conv 1×1 , and the density maps are upsampled to the same resolution. SSIM Loss, MAE Loss, and MSE Loss are combined as the final loss function. Experiments on four datasets (ShanghaiTech [11], UCF_CC_50 [18], WorldExpo’10 [19], and Smart City [12]) demonstrate the effectiveness and robustness of our method.

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

- (1) We propose a novel crowd counting network MGANet, including MFE and MFG. It can effectively deal with the information redundancy and confusion caused in the process of multiscale feature fusion and generate a high-quality density map for accurate crowd counting.
- (2) We propose a top-down feature fusion network based on MFE, which realizes the complementarity of different layers of feature information by a multilayer feature cascade. The network can effectively adapt to the change of crowd scale and prevent the loss of context information. MFG is a network containing channel attention and spatial attention, which can eliminate the influence of redundant features, and the resulting density map pays more attention to the crowd.

FIGURE 2: The architecture of a multi-scale guided attention network. Different feature maps are obtained at different layers of the backbone network, and these feature maps are fused by MFE. MFG guides the feature map obtained by MFE, outputs the density map, and carries out the final count.

- (3) We propose a counting loss function, combining SSIM Loss, MAE Loss, and MSE Loss, which is designed to achieve effective convergence of the network.

2. Related Work

The researchers summarized a series of excellent crowd counting methods [20, 21]. We briefly introduce the research related to the paper, including crowd counting methods and the attention mechanism related to the crowd.

2.1. Crowd Counting Methods. Crowd counting methods can be classified into two categories: traditional methods and CNN-based methods.

2.1.1. Traditional Methods. Traditional methods are mainly classified into two categories: detection-based methods and regression-based methods. In the early days, people used the detection-based method to count the crowd. The main feature of this method was that the sliding windows were used to manually extract the whole of the body features of the human body for detection, for example, Haar wavelet,

HOG, or pedestrian edge information [4, 22]. But in a crowded situation, these methods do not work. To solve these problems, researchers have tried to examine specific body parts rather than the whole body, such as the shoulders or head [23, 24], but these methods are still unable to cope with high-density crowd scenes. The regression-based method counts the crowd by learning a feature-to-crowd mapping [25–27]. This method firstly extracts the low-level features such as texture and edge and then selects the appropriate machine learning regression model to realize the mapping of low-level features to the crowd. Whether the detection method or the regression method, the output information is relatively limited and the processing steps are relatively complicated.

2.1.2. CNN-Based Methods. In recent years, deep learning has developed rapidly, and great progress has been made in crowd counting based on CNN methods. Researchers try to use CNN-based target detectors for counting, including YOLO [28] and Faster-RCNN [29]. Although this method has made great progress compared with the traditional method, it can get better prediction results for the dense crowd in the image by using the fully convolutional network

mapping to the density map. A detailed CNN-based counting survey is presented [30]. MCNN uses a three-column convolutional neural network with a similar structure. The three-column network contains convolution kernels of different sizes. Its purpose is to use convolution kernels of various scales to adapt to the head size of different scales [10]. Switch-CNN is also a multicolumn structure, which uses a density classifier to enable different density patches to be distributed to appropriate networks [8]. MSCNN used multiscale clusters for computation, which can generate scale-dependent features in a single column structure, thus achieving high performance of crowd counting, with high accuracy and cost-effectiveness in practical applications [31]. CSRNet is divided into front-end and back-end networks, and the VGG16 with full connection layer is eliminated as the front-end network, while the dilated convolutional neural network is used as the back-end network. While maintaining the resolution, the perception domain is enlarged to generate a high-quality crowd density map [11]. CP-CNN firstly extracts and classifies the features of the whole input image, converts the classification results into a global context related to the density level, and performs the same operation for the patches cut from the image to obtain the local context. Finally, the whole feature is constrained, so that the network can adaptively learn the features of the corresponding density level for any image [32]. SANet extracts the head information of multiple scales of the image, which uses a module similar to inception architecture. Different sizes of convolution kernels are used in each convolutional layer at the same time, and finally, the final high-resolution density map is obtained through deconvolution [33]. DADNet fuses the scale-aware attention output with different expansion rates to capture different visual granularity of the area of interest to the crowd and uses deformable convolution to generate high-quality density maps [15]. CAN fuses feature acquired from different receptive fields, learns the importance of each feature in the image, and adaptively encodes the scale of context information to accurately predict crowd density. These studies realize the aggregation of multiscale information utilizing multicolumn structure, parallel dilated convolution, or fusion of features of different levels [16]. MS-GAN combines a multiscale convolutional neural network (generator) and an adversarial network (discriminator) to generate a high-quality density map [34].

2.2. Attention Mechanism. To extract effective features, the attention mechanism has also been extensively studied in the field of crowd counting. DecideNet believes that the detection-based method is better in the sparse scene, while the regression method is better in the dense crowd scene. Therefore, the attention mechanism is adopted to adjust the weight of the two methods of detection and regression adaptively according to the change of crowd density [35]. DADNet uses multicolumns dilated Conv to deal with multiscale problems and uses Conv + sigmoid to predict an attention map on different columns to achieve feature selection [15]. SCAR proposed to add a spatialwise attention

model and a channelwise attention model to filter image features and then integrate features [36]. The focus for free integrates the tasks of segmentation, counting, and classification; uses segmentation to provide spatial attention for counting; and uses classification to provide channel attention for counting, to promote the accuracy of counting [37]. ASNet first generates interest masks at different density levels, then generates density maps and scaling factors, multiplying them to output individual attention-based density maps, adding these density maps to produce the final density map [20]. These attention mechanisms improve the counting accuracy only depending on the feature fusion method. This paper also considers the use of channel and spatial attention mechanisms for multiscale fused features. With a series of attention modules, the final attention effort can be refined and the noise gradually eliminated, giving more weight to the really important areas.

3. Method

We propose that MGANet can accurately count people with different crowding degrees. In this section, we elaborate on four aspects of MFE, MFG, feature fusion, and loss function.

3.1. MFE (Multiscale Feature Extraction). In the process of crowd image acquisition, the different distances between the crowd and camera in the same scene will cause a perspective effect; that is, there is a multiscale problem. To solve the problem, we propose a feature fusion method from top to bottom. VGG16 has a good characteristic representation ability, and its network is easy to modify. Therefore, our network chooses VGG16 as the front-end feature extractor. Among others, Conv3-3, Conv4-3, and Conv5-3 are used to output features of dimensions 1/4, 1/8, and 1/16 of the original input dimensions. Due to the forward propagation of the network, the receptive field of the feature map gradually becomes larger. In the feature map generated by the front layer of the network, attention should be paid to the small-size heads, while in the feature map generated by the rear layer, attention should be paid to the large-size heads. The calculation formula of the receptive field is shown in

$$RF_i = (RF_{i+1} - 1) * stride_i + Ksize_i, \quad (1)$$

where RF_i is the receptive field of the convolution layer at the i layer, RF_{i+1} is the stride of the receptive field at the $i + 1$ layer, $stride$ is the step length of convolution, and $Ksize$ is the convolution kernel. Besides, the high-level features contain more semantic information, while the low-level features retain the location information of the input image. Therefore, it is an effective method to improve network performance by combining the features of different levels. Our specific fusion method is shown in

$$F_{ij} = F_{i-1,j} + f(U_{i-1,j-1}), \quad (2)$$

where F_{ij} represents the i th row; the j th column fuses features, $i = \{1, 2, 3\}$, $j = \{1, 2, 3\}$; $f(U_{i-1,j-1})$ represents the $(i - 1)$ th row; and the $(j - 1)$ th column is the feature obtained by the upsampling module. This fusion method can

not only solve the multiscale problem of the human head but also make full use of the semantic information and location information of the feature map of different layers. In this work, $f(U)$ was the key to connecting the layers, we named upsampled embedded block (UEB). Its detailed design is shown in Figure 3.

Firstly, the feature maps of the Conv5_3 output with the nearest neighbor interpolation are upsampled by UEB and fused with the feature map of the Conv4_3 output; the same operation is done with the Conv4_3 and Conv3_3. The network obtains the first layer of fusion features and repeats similar operations. With Conv3_3, Conv4_3, and Conv5_3, we had three sets of feature fusions (M_1, M_2, M_3) which were fed into the MFG for further manipulation.

3.2. MFG (Multiscale Feature Guide). The features obtained from MFE need to be advanced to generate a high-quality density map. The specific operation includes spatial attention mechanism and channel attention mechanism.

3.2.1. Spatial Attention. The full convolutional network regresses the pixels of the density map and does not explicitly give more attention to the head region. In other words, crowd background may influence training loss. To solve the problem of crowd and background confusion, we use spatial attention to make the network generate a spatial attention map of crowd head information. Then, the attention map is applied to suppress the selection of nonhead region information to make the network focus more on the head region (see Figure 4).

M_i , the feature map taken by MFE on different columns, is used to predict an attention map with conv + sigmoid of $1 * 1$, respectively, and then multiplied by M_i with attention map to obtain the spatial attention feature map F_i^{s-attr} . The process is shown in

$$F_i^{s-attr} = M_i \bullet W_i \in R^{H*W*ch}, \quad (3)$$

where W_i is the attention map obtained by conv + sigmoid and i is $\{1, 2, 3\}$. The solution process is shown in the following equations:

$$I_i = \text{Sigmoid}(C_{\mathbb{H}}(M_i, \Theta_c)) \in R^{H*W}, \quad (4)$$

$$W_i = \frac{I_i}{\sum_l I_l} \in R^{H*W}, \quad (5)$$

where $i \in [1, S]$, I_i is the feature map of each channel, $C_{\mathbb{H}}$ represents the convolution of $1 * 1$, Θ_c represent the parameters of $1 * 1$ convolution, and Sigmoid is the activation function.

3.2.2. Channel Attention. Each channel of a high-level feature can be viewed as a specific category of responses, with different semantic responses linked to each other. By utilizing the dependency relationship between different channels, the interdependence between features can be

FIGURE 3: The architecture of upsampled embedded block (UEB). The bilinear interpolation method is used for upsampling to restore the high-level features to the same size as the low-level features. Conv $3 * 3$ convolution is responsible for learning weights, extracting the high-level semantics for better-integrated information, and the fusion features could be further fused from top to bottom. The specific fusion features of Conv3_3, Conv4_3, and Conv5_3 from the VGG16 backbone network are used.

enhanced and the expression of feature semantics can be improved (see Figure 5).

The dimension of M_i is $C * H * W$. First, reshape M_i to get $C * N$, where $N = H * W$. Then, the transpose matrix of M_i is multiplied by C_1 . Finally, C_1 is sent to softmax to obtain C_2 . The above operation process is shown as follows:

$$x_{ij} = \frac{\exp(M_i \bullet M_j)}{\sum_{i=1}^C \exp(M_i \bullet M_j)}, \quad (6)$$

where x_{ji} is the effect on the i th channel on the j th channel and C_2 consists of x_{ji} . C_2 and M_i reshape matrix multiplication and will get the final result combined with M_i channel attention to F_i^{C-attr} ; its expression is as follows:

$$F_i^{C-attr} = w \sum_{i=1}^C (x_{ji} M_i) + M_j. \quad (7)$$

w is initialized to 0 and gradually learns to assign more weight to it. In the process, the calculated C_2 matrix also plays a role of attention, each row of which calculates the dependency relationship between a certain channel and other channels. The value is changed to 0-1 by softmax probability; the greater the value is, the stronger the dependency will be. By multiplying the attention map and M_i , the dependent channels were selectively integrated, semantic feature expression was improved, and long-distance semantic dependence between different channels was modeled.

3.2.3. Fusion of Features. For the output of spatial attention and channel attention, the elementwise sum is executed to complete feature fusion, and the density map is obtained through Conv $1*1$. Since the density map size is $1/4$, $1/8$, and $1/16$ of the original input size, the upsampling operation is needed to obtain the feature map with the final size of $1/4$ of the original image size.

FIGURE 4: Schematic diagram of the spatial attention network.

3.3. Loss Function. In the crowd scene, the local features of the high-density crowd are different, but MSE Loss is based on the assumption of pixel independence, and the local correlation of the density map is not considered. Besides, the counting error of the image is not taken into account. Therefore, SSIM loss and MAE loss are considered to be added to the loss function. The SSIM Loss calculates the similarity between two images according to the local features, and the similarity between the generated crowd density map and the truth value can be compared. The MAE Loss directly measures the difference between the estimated crowd number and the ground truth.

3.3.1. SSIM Loss. SSIM is an indicator widely used in the field of image quality assessment. It calculates the similarity between two images based on local patterns (including mean, variance, and covariance). The value range of SSIM is $[-1, 1]$. The more similar the two images are, the greater the value is. When the two images are the same, it is equal to 1. First, the local statistics are estimated using an 11×11 normalized Gaussian kernel with a standard deviation of 1.5. Then, the weight set from $W = \{W(r) | r \in R, R = \{(-5, 5), \dots, (-5, 5)\}\}$ is defined, where r is the center and R contains the kernel of all locations. Therefore, for each position t , the local statistics of the density map F_d and the corresponding truth value D are calculated. The local mean μ_{F_d} and variance $\sigma_{F_d}^2$ of F_d are calculated, as follows:

$$\mu_{F_d}(t_{F_d}) = \sum_{r_{F_d} \in R_{F_d}} W(r_{F_d}) \bullet F(t_{F_d} + r_{F_d}), \quad (8)$$

$$\sigma_{F_d}(t_{F_d}) = \sum_{r_{F_d} \in R_{F_d}} W(r_{F_d}) \bullet [F(t_{F_d} + r_{F_d}) - \mu_{F_d}(t_{F_d})]^2. \quad (9)$$

The local mean μ_d and variance σ_D^2 of D are as follows:

$$\mu_d(t_d) = \sum_{r_d \in R_d} W(r_d) \bullet F(t_d + r_d), \quad (10)$$

$$\sigma_D^2(t_d) = \sum_{r_d \in R_d} W(r_d) \bullet [F(t_d + r_d) - \mu_{F_d}(t_d)]^2. \quad (11)$$

We can calculate the local covariance $\sigma_{F_d D}$ between F_d and D as follows:

$$\sigma_{F_d D}(t) = \sum_{r \in R} W(r) [F(t + r) - \mu_{F_d}(t_{F_d})] \bullet [Y(t + r) - \mu_d(t_d)]. \quad (12)$$

According to these indicators, SSIM is calculated point by point:

$$\text{SSIM} = \frac{(2\mu_{F_d} + Q_1)(2\sigma_{F_d D} + Q_2)}{(\mu_{F_d}^2 + \mu_D^2 + Q_1)(\mu_{F_d}^2 + \mu_D^2 + Q_2)}, \quad (13)$$

where Q_1 and Q_2 are random very small constants to avoid being divided by 0.

Finally, the loss function of SSIM can be defined as follows:

$$L_s = 1 - \frac{1}{M} \sum_t \text{SSIM}(t), \quad (14)$$

where M is the sum of pixels in the density map.

3.3.2. MAE Loss. Mean absolute error loss is introduced to assess the counts and the estimated values as follows:

$$L_c = \frac{1}{N} \sum_{i=1}^N |C(I_i) - C'(I_i)|, \quad (15)$$

where I_i means the density map generated by MGANet, $C(I_i)$ is the estimated count, and $C'(I_i)$ is the label value.

3.3.3. MSE Loss. The Euclidean loss is used to assess the difference between the training density map and the model output density map, to facilitate the model to adjust parameters and produce a density map closer to the real situation. The Euclidean loss can measure estimation error at the pixel level. The loss function is given as follows:

$$L_E = \frac{1}{2N} \sum_{i=1}^N \|F(X_i; \theta) - F_i\|_2^2, \quad (16)$$

where $F(X_i; \theta)$ is the output of MGANet, θ is the variable parameters of the model parameters, X_i is the input image, and F_i represents the ground truth result.

FIGURE 5: Channel attention mechanism.

3.3.4. The Final Count Loss. The final loss consists of L_s , L_c , and L_E . α and β are weighting factors of L_s and L_c which are set as 0.001 and 0.001 separately:

$$L(\Theta) = L_E + \alpha L_s + \beta L_c. \quad (17)$$

4. Experiment

In this section, we introduce the evaluation metrics, the three standard benchmark datasets, the experimental setup, and the training method in order. Finally, we perform experimental results on the datasets.

4.1. Training Details. MGANet is an end-to-end structure, which is implemented using the PyTorch framework. So, the training process is very simple. We set the training batch size as 1. MGANet uses standard SGD as the optimization with the momentum at 0.9. Besides, we employ random Gaussian initialization with a standard deviation of 0.01. The initial learning rate is set to 0.001 and decreases as the number of iterations increases.

4.2. Evaluation Metrics. Following the existing literature, the evaluation metrics are the mean absolute error (MAE) and mean squared error (MSE), which can evaluate the performance of our method. MAE indicates the accuracy of the model counting, and MSE represents the robustness of the model. The formulae are as follows:

$$\text{MAE} = \frac{1}{N} |Z_i - Z'_i|, \quad (18)$$

$$\text{MSE} = \sqrt{\frac{1}{N} \sum_{i=1}^N (Z_i - Z'_i)^2}, \quad (19)$$

where N represents the number of test images. Z_i is the actual number of people in the i th test image. Z'_i is the corresponding estimated count, which is the output by our model.

4.3. Ground Truth Generation. For the density map to be able to adapt to various conditions of the crowd image, it can be expressed as $F(x)$ with N heads. The calculation method of $F(x)$ is to convolve the delta function $\delta(x - x_i)$ with a Gaussian Kernel $G_{\sigma_i}(x)$ normalized to 1:

$$F(x) = \sum_{i=1}^N \delta(x - x_i) * G_{\sigma_i}(x), \quad \text{with } \sigma_i = \beta \bar{d}^i, \quad (20)$$

where x_i represents per pedestrian head at the pixel, σ_i is the crowd distribution of all the images in the dataset, β is a constant, and $\bar{d}^i$ represents the average distance of k nearest neighbors of the target.

In our experiments, we follow the configuration in CSRNet [11]. Certain parameters are set to fixed values ($\beta = 0.3, k = 3$). The setups are shown in Table 1.

4.4. Comparisons with State of the Art. We evaluate our method on three publicly available crowd counting datasets: ShanghaiTech, UCF_CC_50, WorldExpo'10, and Smart City. These datasets include different crowd situations, such as dense scenes and sparse scenes. Tables 2–5 show the comparison results of MAE and MSE, respectively. Besides, the results of the model tests are visualized in Figures 6–8.

4.4.1. ShanghaiTech Dataset. ShanghaiTech dataset [11] includes two sets: Part A and Part B. It is taken in sparse scenes, which consists of 1198 crowd images with 330,165 people. Part A consists of 300 training images and 182 testing images. Part B contains 400 training images and 316 testing images, counting from 9 to 578.

ShanghaiTech dataset test results and visual results are shown in Table 6 and Figure 6. In the visualization section, we tested the low-density to high-density crowd images in Part A and Part B, respectively. We scored 96.7 on MSE on Part A, and other data also performed well. It can be seen that we perform well in both low-density and high-density scenarios.

4.4.2. UCF_CC_50 Dataset. UCF_CC_50 dataset [18] is a very challenging dataset, due to different perspectives, small dataset size, and resolutions. It contains 50 crowd images with 63974 people. The annotated persons range from 94 to 4543, with an average of 1280. Fivefold cross-validation is the most commonly used method in this dataset. Our results

TABLE 1: The setups for different datasets.

Dataset	Parameter settings
ShanghaiTech Part A [11]	$\sigma = 4$
ShanghaiTech Part B [11]	$\sigma = 15$
WorldExpo'10 [19]	$\sigma = 3$
Smart City [12]	$\sigma = 3$
UCF_CC_50 [18]	Geometry-adaptive kernels

TABLE 2: Experimental comparison data on UCF_CC_50 dataset.

Dataset	UCF-CC-50	
	MAE	MSE
Zhang et al. [19]	467.0	498.5
MCNN [10]	377.6	509.1
Switch-CNN [8]	318.1	439.2
CP-CNN [31]	295.8	320.9
CSRNet [11]	266.1	397.5
SANet [2]	258.4	344.9
PACNN [13]	241.7	320.7
CAN [16]	212.2	243.7
DADNet [15]	285.5	389.7
SDANet [38]	227.6	316.4
Ours	240.8	311.5

TABLE 3: Experimental comparison data on WorldExpo'10 dataset.

Dataset	WorldExpo'10					
	S1	S2	S3	S4	S5	Avg.
Zhang et al. [19]	9.8	14.1	14.3	22.2	3.7	12.9
MCNN [10]	3.4	20.6	12.9	13.0	8.1	11.6
Switch-CNN [8]	4.4	15.7	10.0	11.0	5.9	9.4
CP-CNN [31]	2.9	14.7	10.5	10.4	5.8	8.9
CSRNet [11]	2.9	11.5	8.6	16.6	3.4	8.6
PACNN [13]	2.3	12.5	9.1	11.2	3.8	7.8
CAN [16]	2.4	9.4	8.8	11.2	4.0	7.2
SDANet [38]	2.0	14.3	12.5	9.5	2.5	8.1
Ours	2.1	13.5	9.3	11.4	3.0	7.86

are lower than 212.2 and 243.7 of CAN [16]. In the UCF_CC_50 dataset, images are high-density scenes without complex background information. Our MFGNet is more suitable for complex background situations.

The test results and visual results on the UCF_CC_50 are shown in Table 2 and Figure 7. It can be seen that our MAE and MSE are 240.8 and 311.5, respectively, which are greatly improved compared with other methods, proving that our MAE and MSE also perform well in small datasets and high-density scenarios.

4.4.3. WorldExpo'10 Dataset. WorldExpo'10 [19] includes 108 different surveillance cameras, containing 3,980 training frames in 1,132 video sequences. It can provide the cross scene to evaluate the model. And regions of interest (ROI) are provided for all scenes.

The test results and visual results on the WorldExpo'10 dataset are shown in Table 3 and Figure 8. The data is divided into 5 different scenarios with large background interference. We tested each of the five scenarios, and the final score

is 7.86. In S1 and S5 scenarios, the scores are 2.1 and 3.0, respectively, better than 2.4 and 4.0 in CAN [16]. It can be seen that our model also performs well on the multiscene crowd counting dataset under complex background.

4.4.4. Smart City Dataset. Smart City [12] contains 50 images. When collecting data, the shooting angle is high, and the scene includes ten scene images such as a sidewalk and shopping mall. The images are divided into indoor scenes and outdoor scenes, and there are few pedestrians, ranging from 1 to 14 people from the lowest to the highest, with an average of 7.4 people.

The test results and visual results on the Smart City dataset are shown in Table 4 and Figure 9. Relatively speaking, the background of people in this dataset is more complex. On this dataset, we got the best results. They are 8.2 and 10.2, respectively, better than 9.4 and 11.4 in CAN [16]. It can be seen that our model also performs well on the multiscene crowd counting dataset under complex background.

4.5. Ablation Experiment

4.5.1. Top-Down Feature Fusion. In the experiment to verify the feature fusion function, we conducted three different cascade fusion comparisons according to the network architecture: (1) no cascading fusion: only the original features output by VGG16 are used; (2) one-level cascading fusion: add a top-down cascade operation to the original VGG16 feature; (3) two-level cascading fusion: add a top-down two-tier cascade operation to the original VGG16 feature. Experiments are performed on ShanghaiTech A and B, and the results are shown (see Table 5). With the increase of cascade operation, the indicators gradually become better. However, the complexity of the model is gradually increasing, and the performance will not be significantly improved.

4.5.2. The Function of the Attention Mechanism

(1) Comparison of Different Attention Modules. In the experiments to verify the effect of attention, we set 4 conditions according to the network architecture: (1) no attention; (2) only channel attention is used; (3) only spatial attention is used; and (4) channel attention and spatial attention are used. Experiments are performed on ShanghaiTech A and B, and the results are shown in Table 7.

(2) Comparison of the Same Form of Attention Mechanism. Channel attention and spatial attention are also included in the CBAM [39]. Although there are similar names, they make certain differences. We believe that the tasks of CBAM and MFG are different. CBAM pays more attention to the recognition of target objects, which also makes CBAM have better explanatory. MFG pays more attention to pixel-level attention design and is more suitable for crowd counting. To be fair, we add CBAM and MFG to the MCNN [10] network, respectively, and the

TABLE 4: Experimental comparison data on Smart City dataset.

Dataset	Smart City	
Evaluation	MAE	MSE
Zhang et al. [19]	40.0	46.2
Switch-CNN [8]	23.4	25.2
SaCNN [12]	8.6	11.6
CAN [16]	9.4	11.4
Ours	8.2	10.2

TABLE 5: Experimental results of top-down feature fusion ablation.

Dataset	ShanghaiTech A		ShanghaiTech B	
	MAE	MSE	MAE	MSE
No cascading fusion	77.5	116.7	10.8	16.8
One-level cascading fusion	68.7	109.6	10.2	14.5
Two-level cascading fusion	66.5	96.7	9.2	13.8

FIGURE 6: ShanghaiTech dataset visualization results show that the first line is the original picture, the second line is the ground truth, the third line is the prediction result of the model, and the fourth line is the true value/predicted value.

test results are shown in Table 8. (1) MCNN: the baseline of this paper. It is a fully convolutional neural network with three columns of convolution kernels of different sizes. (2) MCNN + CBAM: CBAM is added to the top of MCNN. (3) MCNN + MFG: MFG is added to the top of MCNN. (4) MGANet: the full model is proposed by ours, which consists of MFE and MFG.

We find that MFG achieves a better performance than CBAM: 84.4/136.7 versus 120.6/184.61 of MAE/MSE on ShanghaiTech A, 14.3/23.3 versus 29.77/49.66 of MAE/MSE on ShanghaiTech B. When MFG is embedded into the proposed network, the proposed model attains the best performance regardless of the counting results or density maps' quality.

4.5.3. Function of Loss

(1) *Different Combinations of Loss Functions.* In the experiments of different loss function combinations, we set four different settings based on MSE Loss, namely, MSE Loss, SSIM + MSE Loss, MAE + MSE Loss, and SSIM + MAE + MSE Loss. The experiment was carried out in ShanghaiTech A and B, and the results are shown in Table 9. It can be seen that MSE Loss, the most commonly used loss, still plays a major role in network convergence, but the combination of SSIM Loss and MAE Loss in this paper can also promote network convergence to a certain extent.

(2) *Setting of Super Parameters α and β .* We briefly compare the different parameter settings of the presented

FIGURE 7: UCF_CC_50 visualization results show that the first column is the original image, the second column is the ground truth, the third column is the prediction results, and the fourth column is the true/predicted values.

FIGURE 8: Visualization results of WorldExpo'10. The first column is the original image, the second column is the ground truth, the third column is the prediction results, and the fourth column is the true/predicted values.

TABLE 6: Experimental comparison data on ShanghaiTech dataset.

Dataset Evaluation	ShanghaiTech A		ShanghaiTech B	
	MAE	MSE	MAE	MSE
Zhang et al. [19]	181.8	277.7	32.0	49.8
MCNN [10]	110.2	173.2	26.4	41.3
Switching-CNN [8]	95.6	138.6	21.6	33.4
CP-CNN [10]	73.6	106.4	20.1	30.1
CSRNet [11]	68.2	115.0	10.6	16.0
SANet [2]	67.0	104.5	8.4	13.6
PACNN [13]	66.3	106.4	8.9	13.5
CAN [16]	62.3	100.0	7.8	12.2
DADNet [15]	64.2	99.9	8.8	13.5
SDANet [38]	63.6	101.8	8.2	12.8
Ours	66.5	96.7	9.2	13.8

FIGURE 9: Visualization results of smart city dataset. The first column is the original image, the second column is the ground truth, the third column is the prediction results, and the fourth column is the true/predicted values.

TABLE 7: Experimental results of attention mechanism ablation.

Dataset Evaluation	ShanghaiTech A		ShanghaiTech B	
	MAE	MSE	MAE	MSE
No attention	112.4	165.3	25.4	44.6
Channel attention	75.6	109.5	15.6	22.7
Spatial attention	86.5	116.3	14.5	21.6
Spatial + channel attention	66.5	96.7	9.2	13.8

TABLE 8: Comparison of the same form of attention mechanism.

Dataset Evaluation	ShanghaiTech A		ShanghaiTech B	
	MAE	MSE	MAE	MSE
MCNN	110.2	173.2	26.4	41.3
MCNN + CBAM	120.6	184.61	29.77	49.66
MCNN + MFG	84.4	136.7	14.3	23.3
MGANet (ours)	66.5	96.7	9.2	13.8

TABLE 9: Experimental results of different combinations of loss functions' ablation.

Dataset Evaluation	ShanghaiTech A		ShanghaiTech B	
	MAE	MSE	MAE	MSE
MSE Loss	85.0	150.4	11.6	15.8
SSIM + MSE Loss	70.3	147.8	10.5	14.4
MAE + MSE Loss	70.6	140.6	10.3	14.5
SSIM + MAE + MSE Loss	66.5	96.7	9.2	13.8

TABLE 10: Comparison of the estimation errors trained with different parameters in loss function on the ShanghaiTech A dataset.

Loss function	α	β	MAE	MSE
$L_E + \alpha L_s + \beta L_c$	10^{-4}	10^{-4}	66.5	96.7
	10^{-3}	10^{-3}	69.9	103.4
	10^{-4}	10^{-3}	67.8	102.5
	10^{-3}	10^{-4}	68.3	98.4

loss function on ShanghaiTech A demonstrated in Table 10. It indicates that when α and β are 10^{-4} and 10^{-4} , the MAE and MSE reach the lowest.

5. Conclusion

In this paper, we propose MGANet for accurate and effective crowd counting. To obtain multiscale information and prevent the loss of context information, we propose a top-down method to concatenate deep and shallow features. To make the network pay more attention to the spatial information and channel information of the crowd in the image and ignore the irrelevant information, we design a combination of spatial attention and channel attention that pay more attention to the pixel level and further guide the features. To obtain a high-quality density map, based on the commonly used MSE Loss, the loss function that can effectively promote network convergence is obtained by combining SSIM Loss and MAE Loss. A large number of experiments have proved the effectiveness and robustness of our method, and the related ablation experiments have also confirmed the effectiveness of each module.

Data Availability

All data can be found at <https://github.com/lpfworld/MGANet>. Other raw data supporting the conclusions of this article can be obtained from the corresponding author upon request.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

Acknowledgments

This work was supported by the Zhejiang Provincial Technical Plan Project (nos. 2020C03105 and 2021C01129).

References

- [1] L. Boominathan, S. S. Kruthiventi, and R. V. Babu, "Crowdnet: a deep convolutional network for dense crowd counting," in *Proceedings of the 24th ACM International Conference on Multimedia*, pp. 640–644, Amsterdam, Netherlands, October 2016.
- [2] X. Cao, Z. Wang, Y. Zhao, and F. Su, "Scale aggregation network for accurate and efficient crowd counting," in *Proceedings of the European Conference on Computer Vision (ECCV)*, pp. 734–750, Munich, Germany, September 2018.
- [3] A. B. Chan, Z. S. Liang, J. Sheng, and N. Vasconcelos, "Privacy preserving crowd monitoring: counting people without people models or tracking," in *Proceedings of the 2008 IEEE Conference on Computer Vision and Pattern Recognition*, pp. 1–7, IEEE, Anchorage, Alaska, June 2008.
- [4] H. Xiong, H. Lu, C. Liu, L. Liu, Z. Cao, and C. Shen, "From open set to closed set: counting objects by spatial divide-and-conquer," in *Proceedings of the IEEE ICCV*, pp. 8362–8371, Seoul, Korea, November 2019.
- [5] Y. Hu, X. Jiang, X. Liu et al., "NAS-count: counting-by-density with neural architecture search," in *Proceedings of the Computer Vision-ECCV 2020-16th European Conference*, pp. 747–766, Glasgow, UK, August 2020.
- [6] W. Xie, J. A. Noble, and A. Zisserman, "Microscopy cell counting and detection with fully convolutional regression networks," *Computer Methods in Biomechanics and Biomedical Engineering: Imaging & Visualization*, vol. 6, no. 3, pp. 283–292, 2018.
- [7] L. Liu, H. Lu, H. Zou, H. Xiong, Z. Cao, and C. Shen, "Weighing counts: sequential crowd counting by reinforcement learning," *Computer Vision-ECCV 2020*, vol. 8, pp. 164–181, 2020.
- [8] D. B. Sam, S. Surya, and R. V. Babu, "Switching convolutional neural network for crowd counting," in *Proceedings of the 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 4031–4039, IEEE, Honolulu, Hawaii, July 2017.
- [9] Z. H. Ma, X. Wei, and X. P. Hong, "Learning scales from points: a scale-aware probabilistic model for crowd counting," *ACM Multimedia*, pp. 220–228, 2020.
- [10] Y. Zhang, D. Zhou, S. Chen, S. Gao, and Y. Ma, "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, Las Vegas, NV, USA, June 2016.
- [11] Y. Li, X. Zhang, and D. Chen, "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, Salt Lake City, UT, USA, June 2018.
- [12] L. Zhang, M. Shi, and Q. Chen, “Crowd counting via scale-adaptive convolutional neural network,” in *Proceedings of the 2018 IEEE Winter Conference on Applications of Computer Vision (WACV)*, pp. 1113–1121, IEEE, Lake Tahoe, NV, USA, March 2018.
 - [13] M. Shi, Z. Yang, C. Xu, and Q. Chen, “Revisiting perspective information for efficient crowd counting,” in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 7279–7288, Long Beach, CA, USA, June 2019.
 - [14] X. Liu, J. Van De Weijer, and A. D. Bagdanov, “Leveraging unlabeled data for crowd counting by learning to rank,” in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 7661–7669, Salt Lake City, UT, USA, June 2018.
 - [15] D. Guo, K. Li, Z. J. Zha, and M. Wang, “DADNet: dilated-attention-deformable ConvNet for crowd counting,” in *Proceedings of the 27th ACM International Conference on Multimedia*, pp. 1823–1832, Nice, France, October 2019.
 - [16] W. Liu, M. Salzmann, and P. Fua, “Context-aware crowd counting,” in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 5099–5108, Long Beach, CA, USA, June 2019.
 - [17] L. Rong and C. Li, “Coarse-and fine-grained attention network with background-aware loss for crowd density map estimation,” in *Proceedings of the 2021 IEEE Winter Conference on Applications of Computer Vision (WACV)*, pp. 3674–3683, IEEE, June 2023.
 - [18] I. Haroon, I. Saleemi, C. Seibert, and M. Shah, “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, Portland, OR, USA, June 2013.
 - [19] C. Zhang, H. Li, X. Wang, and X. Yang, “Cross-scene crowd counting via deep convolutional neural networks,” in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 833–841, Boston, MA, USA, June 2015.
 - [20] D. Kang, Z. Ma, and A. B. Chan, “Beyond counting: comparisons of density maps for crowd analysis tasks—counting, detection, and tracking,” *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 29, no. 5, pp. 1408–1422, 2018.
 - [21] V. A. Sindagi and V. M. Patel, “A survey of recent advances in cnn-based single image crowd counting and density estimation,” *Pattern Recognition Letters*, vol. 107, pp. 3–16, 2018.
 - [22] M. Enzweiler and D. M. Gavrilu, “Monocular pedestrian detection: survey and experiments,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 31, no. 12, pp. 2179–2195, 2018.
 - [23] P. F. Felzenszwalb, R. B. Girshick, D. McAllester, and D. Ramanan, “Object detection with discriminatively trained part-based models,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 32, no. 9, pp. 1627–1645, 2009.
 - [24] B. Wu and R. Nevatia, “Detection and tracking of multiple, partially occluded humans by bayesian combination of edgelet based part detectors,” *International Journal of Computer Vision*, vol. 75, no. 2, pp. 247–266, 2007.
 - [25] A. B. Chan and N. Vasconcelos, “Bayesian Poisson regression for crowd counting,” in *Proceedings of the 2009 IEEE 12th International Conference on Computer Vision*, pp. 545–551, IEEE, Kyoto, Japan, October 2009.
 - [26] D. Ryan, S. Denman, C. Fookes, and S. Sridharan, “Crowd counting using multiple local features,” in *Proceedings of the 2009 Digital Image Computing: Techniques and Applications*, pp. 81–88, IEEE, Melbourne, Australia, December 2009.
 - [27] K. Chen, C. C. Loy, S. Gong, and T. Xiang, “Feature mining for localised crowd counting,” *BMVC*, vol. 1, no. 2, 2012.
 - [28] J. Redmon and A. Farhadi, “YOLOv3: an incremental improvement,” 2018, <https://arxiv.org/abs/1804.02767>.
 - [29] R. Girshick, “Fast R-CNN,” in *Proceedings of the IEEE International Conference on Computer Vision*, pp. 1440–1448, Santiago, Chile, December 2015.
 - [30] H. Bai and S. Chan, “CNN-based single image crowd counting: network design, loss function and supervisory signal,” 2012, <https://arxiv.org/abs/2012.15685>.
 - [31] L. Zeng, X. Xu, B. Cai, S. Qiu, and T. Zhang, “Multi-scale convolutional neural networks for crowd counting,” in *Proceedings of the 2017 IEEE International Conference on Image Processing*, pp. 465–469, Beijing, China, September 2017.
 - [32] V. A. Sindagi and V. M. Patel, “Generating high-quality crowd density maps using contextual pyramid CNNs,” in *Proceedings of the IEEE International Conference on Computer Vision*, pp. 1861–1870, Venice, Italy, October 2017.
 - [33] X. Jiang, L. Zhang, M. Xu et al., “Attention scaling for crowd counting,” in *Proceedings of the 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 4705–4714, Seattle, WA, USA, June 2020.
 - [34] Y. Zhou, J. Yang, and H. Li, “Adversarial learning for multiscale crowd counting under complex scenes,” *IEEE Transactions on Cybernetics*, pp. 1–10, 2020.
 - [35] J. Liu, C. Gao, D. Meng, and A. G. Hauptmann, “Decidenet: counting varying density crowds through attention guided detection and density estimation,” in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 5197–5206, Salt Lake City, Utah, USA, June 2018.
 - [36] J. Gao, Q. Wang, and Y. Yuan, “SCAR: spatial-/channel-wise attention regression networks for crowd counting,” *Neurocomputing*, vol. 363, pp. 1–8, 2019.
 - [37] Z. Shi, P. Mettes, and C. G. Snoek, “Counting with focus for free,” in *Proceedings of the IEEE International Conference on Computer Vision*, pp. 4200–4209, Seoul, Korea, November 2019.
 - [38] Y. Miao, Z. Lin, G. Ding, and J. Han, “Shallow feature based dense attention network for crowd counting,” *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 34, no. 7, pp. 11765–11772, 2020.
 - [39] S. Woo, J. Park, J.-Y. Lee, and I. S. Kweon, “In CBAM: convolutional block Attention module,” in *Proceedings of the Computer Vision-ECCV 2018-15th European Conference*, pp. 3–19, Munich, Germany, September 2018.