Learning Multi-Level Features to Improve Crowd Counting

ZHANQIANG HUO, BIN LU, AIZHONG MI, FEN LUO, AND YINGXU QIAO

School of Computer Science and Technology, Henan Polytechnic University, Jiaozuo 454003, China

Corresponding author: Fen Luo (luofenjsj@hpu.edu.cn)

This work was supported in part by the Henan University Scientific and Technological Innovation Team Support Program under Grant 19IRTSTHN012, and in part by the Foundation of Henan Scientific and Technological Project under Grant 192102210281.

ABSTRACT
Crowd counting is a task that aims to estimate the number of people in an image. Recent crowd counting methods make significant progress by employing convolutional neural networks to regress crowd density maps. One of the most challenging problems in this task is the drastic scale variation of the region of interest in images. In this paper, a Feature Fusion Attention Network (FFANet) is proposed for crowd counting. Firstly, the VGG16 network is adapted as the backbone of the FFANet to extract the features of crowd images. Then, the extracted features are fused by the subsequent two stages. Specifically, the information enhancement operations on the multi-levels features are conducted by Feature Fusion Attention Module (FFAM), which are further refined by the Residual Block (RB). Finally, the features are processed by the Compression Module (CM) to generate a density map. To demonstrate the effectiveness, the proposed algorithm is verified on three benchmark datasets. Evaluation of the algorithm performances in comparison with other state-of-the-art methods indicates the proposed FFANet outperforms the existing methods.

INDEX TERMS
Crowd counting, scale variation, feature fusion attention.

I. INTRODUCTION
Crowd counting is one of the promising applications in computer vision. It is a task that aims to estimate people’s numbers in an image. The predicted results can be used in a wide range of fields, such as intelligent transportation [1], public security [2], agriculture monitoring [3], video surveillance [4] and so on. However, crowd counting is also a highly challenging task because of occlusion, low image resolution, perspective distortion, scale variation of objects, etc [5]. To obtain accurate results, researchers have paid lots of attention to study the above issues.

Early methods for crowd counting are based on manual features extraction of the human body and various regression functions [6]. These methods usually don’t perform well in dense crowd scenes where pedestrians are severely occluded or overlapped. With the development of deep learning technology in computer vision, the algorithms based on CNN have made great progress by conquering the scales variation. These methods typically design different sized filters architectures to extract multi-scale features [7]. However, human scales change continuously in the entire image and current models can only concern some discrete scales. It brings a major problem that these methods ignore a larger number of crowds in an image.

Some researchers [8]–[10] discover that CNN’s shallower layers focus on low-level texture and spatial information which can help the model determine the location of the target. The deeper layers focus on high-level contextual and semantic information which can help the model to identify the type of targets. Inspired by the above research, we reasoned that the fusion of these multi-level features can effectively solve the crowd scales variation. There are two ways to realize feature fusion, one is to merge the features on its channel axis and the other is to conduct element-wise summation. The defect of these two methods is that they can’t utilize the information contained in the feature effectively, which results in the waste of calculation. In this paper, we introduce the FFAM to realize the information enhancement of the multi-level features. First, the FFAM utilizes the contextual and semantic information contained in the high-level features to enhance channel-wise information in the low-level features. Second, the spatial information of the processed low-level features would be extracted by the FFAM to enhance the
In recent years, crowd counting methods have made great progress by employing convolutional neural networks to regress crowd density maps. Researchers have designed a variety of efficient convolution neural networks to solve scale variation [5]. The remainder part of this section describes the multi-column models and the single-column models according to the network structure.

1) MULTI-COLUMN MODELS

Zhang et al. [7] proposed a Multi-column Convolutional Neural Network (MCNN) to overcome the scale variation in images. Each column is composed of different filters to get features with various scales. Inspired by MCNN, Onoro-Rubio et al. [12] designed a scale-aware counting model that can predict crowd distribution and the number of the crowd without perspective information, by extracting features from the image with different resolutions to overcome the perspective distortion. Switch-CNN which trains a classifier to choose the optimal branch from the multi-column network for crowd image patches is proposed by Sam et al. [13]. SANet designed by Cao et al. [11] uses scale aggregation modules to solve scale variation and extracts these features to generate high-resolution density maps. Guo et al. [14] explored a scale-aware attention fusion with various dilation rates to capture different visual granularities of crowd regions of interest and utilizes deformable convolutions to generate a high-quality density map. Recently, Gao et al. [15] proposed a Perspective Crowd Counting via Spatial Convolutional Network (PCC Net) to solve high appearance similarity, perspective changes and severe congestion.

2) SINGLE-COLUMN MODELS

CSRNet [16] used dilated convolution layers to expand the receptive field and replace pooling operations. By taking advantage of these designs, CSRNet can easily generate high-quality density maps. Shi et al. [17] proposed a Perspective Aware Convolutional Neural Network (PACNN), which can add perspective information based on crowd estimation and effectively solve the scale variation. AD CrowdNet [18] consists of two CNN networks. An attention aware network firstly detects the crowd regions in the image and calculates the congestion degree of these areas. Based on the detected crowd area, a multi-scale deformable network is used to generate high-quality density maps. More recently, Jiang et al. [19] proposed an effective Multi-Level Convolutional Neural Network (MLCNN) architecture that first adaptively learns multi-level density maps and then fuses them to predict the final output.

B. ATTENTION MODELS

Attention models are first applied in the field of machine translation and then developed into many deep learning fields such as object detection [20], image classification [21], image segmentation [22] and face recognition [23]. Hu et al. [24] proposed a lightweight attention mechanism named SENet which could automatically obtain the importance of each feature channel-wise information to enhance the useful features. Jon. et al. combined the channel attention mechanism and the spatial attention mechanism to propose CBAM [25] and BAM [26]. Li et al. [27] proposed a dynamic selection mechanism.
that allows each neuron to select a different receptive field for the size of the target. Cao et al. [28] combined the advantages of Non-local [29] and SENet and proposed a new global context module, achieving significant results on computer vision tasks.

Multi-column models typically design different sized filters architectures to extract multi-scale features. However, human scales change continuously in the entire image and current models can only concern some discrete scales. It brings a major problem that these methods ignore a larger number of crowds in an image. Single-column models need to rely on other information or auxiliary network to solve the scale variation, for example, PACNN needs to generate perspective maps and train the corresponding network branches. This has led to additional work and increased difficulty in model training. Therefore, combined with the above works, we design a simple and effective network based on multi-level feature fusion to solve the problem of scale variation in crowd counting.

III. METHOD

In this section, we firstly introduce the structure of the proposed FFANet. Then, we introduce the associated modules and the loss function. Fig. 2 shows the details of the network structure. Table 1 describes the configuration of convolutional layers in the proposed FFANet. (3,3)-64-BN-ReLU means that the convolutional block contains 3 × 3 kernel size convolutional operation, 64 output channels, a BN [30] layer and a ReLU activation layer. (FFAM + RB) represents the cascade structure composed of the FFAM and the RB.

A. OVERVIEW

The FFANet consists of a backbone network, feature fusion attention modules, residual blocks and a compression module.

**Backbone** the backbone network is a pre-trained VGG16 [31] network with the fully connected layers removed. A BN layer is added at the back of all convolutional layers in the VGG16 network. The backbone network contains 13 convolutional layers which are divided into 5 convolutional blocks. We take the output of convolution blocks from the third to the fifth as the objects of feature fusion.

**FFAM** the proposed FFAM fuses multi-level features of the backbone and utilizes this information diversity to enhance channel-wise information in the low-level feature and spatial-wise information in the high-level feature.

**RB & CM** the RB is a residual block composed of a 1 × 1 kernel size convolutional layer and two 3 × 3 convolutional layers to refine the features of FFAM output. The compression module (CM) compresses the feature map into a single-channel crowd density map.

**B. FEATURE FUSION ATTENTION MODULE**

Fig. 3 displays the architecture of the FFAM. The input is the features of two adjacent levels in the backbone network. The FFAM upsamples the high-level feature \( F^{i+1} \) and concatenates it with the low-level feature \( F^i \) in the channel axis. The channel attention module uses the concatenated features to output vectors \( w^c \) to enhance channel-wise information in \( F^i \). Fig. 4 (a) describes the structure of the channel attention module. It is formulated as

\[
\hat{w}^c = \phi ([F^i, U(F^{i+1})])
\]

where \( U(\cdot) \) denotes the upsampling layer, \([\cdot]\) denotes the concatenation layer, \( \phi(\cdot) \) denotes the channel attention module. Channel-wise enhanced \( F' \) is obtained by element-wise multiplying vector \( w^c \) and \( F^i \). It is formulated as

\[
F' = w^c \odot F^i
\]

The spatial attention module calculates \( F'' \) to obtain the spatial weight to enhance the high-level feature. Fig. 4 (b) represents the structure of the spatial attention module. We define this operation as

\[
F'' = w^s \odot U(F^{i+1})
\]

where \( \varphi(\cdot) \) denotes the spatial attention module, \( w^s \) denotes the spatial weight. Spatial-wise enhanced \( F'' \) is obtained by element-wise multiplying \( w^s \) and \( U(F^{i+1}) \). It is formulated as

\[
F'' = w^s \odot U(F^{i+1})
\]

Finally, the two enhanced features would be concatenated in the channel axis. It is formulated as

\[
\hat{F} = [F', F'']
\]
FIGURE 2. Overview of the proposed FFANet. The image is fed to the first five convolutional blocks of VGG16 for extracting features. The output from the conv_5 is upsampled and fused with the output from conv_4 to achieve one-stage feature fusion. Then the fused feature maps and the output of the conv_4 are further upsampled and fused with the feature maps from conv_3 to achieve two-stage feature fusion. Finally, the final fused features are processed by the CM to generate a density map.

C. LOSS FUNCTION

We define a joint loss function which consists of Mean Squared Error (MSE) loss and Structural Similarity Index (SSIM) loss. MSE loss is used to minimize the Euclidean distance between the ground truth and the estimated density map. Ref. [11] reveals the fact that MSE loss employed by many previous methods is dependent on the pixel independence hypothesis and doesn’t consider the local correlation of the density map. Therefore, we utilize the SSIM loss as part of the loss function to improve the result. The joint loss function is defined as follows

$$L = L_{MSE} + \lambda L_{SSIM}$$

$$= \frac{1}{N} \sum_{i=1}^{N} \left| D_i^G - D_i^P \right|^2$$
3)-1-Sigmoid means that the convolutional layer contains a 3×3 kernel size convolutional operation, 1 output channels and the Sigmoid activation function.

\[
\text{Dense-}(N)-\text{Sigmoid} = \left\{ \text{Dense-}(N)-\text{ReLU}, \text{Dense-}(N)-\text{Sigmoid} \right\}
\]

\[
\text{FIGURE 4. Structure of the Channel Attention module and Spatial Attention module. Dense-(N)-Sigmoid means that the dense layer contains N neurons and the Sigmoid activation function. The N of the dense layer is set to \{128, 256\} in stages 1, 2 in sequence. Conv-\text{-(3,3)}\text{-Sigmoid}}
\]

where \(N\) is the number of training batch size, \(D_i^P\) is the predicted density map, \(M\) is the number of the pixels in the density map, \(\lambda\) is the parameters which are used to balance \(L^{\text{MSE}}\) and \(L^{\text{SSIM}}\). Means \((\mu_{P_i}, \mu_{G_i})\) and standard deviation \((\sigma_{P_i}, \sigma_{G_i}, \sigma_{P,G_i})\) in SSIM loss are calculated with a Gaussian kernel that configures a standard deviation of 1.5 within an \(11 \times 11\) region at each position \(j\).

\[
\text{SSIM}_i = \left( \frac{2\mu_{P_i}\mu_{G_i} + C_1}{\mu^2_{P_i} + \mu^2_{G_i} + C_1} \right) \cdot \left( \frac{2\sigma_{P_i}\sigma_{G_i} + C_2}{\sigma^2_{P_i} + \sigma^2_{G_i} + C_2} \right)
\]

IV. EXPERIMENT

In this section, we validate the effectiveness of our method on three public crowd counting datasets: ShanghaiTech [7], UCF_CC_50 [32], UCSD [33]. Then we conduct the ablation studies about the hyperparameter \(\lambda\), the density map generation and the structure of the network.

A. DATASETS AND TRAINING DETAILS

- ShanghaiTech. The ShanghaiTech dataset includes 1198 images with 330,165 annotated heads. It consists of two parts: Part A and Part B. Specifically, Part A consists of 482 images, which are randomly selected from the Internet and Part B is selected from the surveillance on the streets of Shanghai. These two parts are further divided into training and evaluation sets, 300 and 182 images from Part A are selected for training and testing respectively, while 400 and 316 images from Part B are also chosen.

- UCF_CC_50. The UCF_CC_50 dataset contains 50 images with different resolutions and each image has an average of 1280 people. In the whole dataset, the number of individuals in each image ranges from 94 to 4543, which indicates that there is a large count variance between images. Considering the small size of the dataset, we use a cross-validation protocol for training and testing our methods following the approach from ref. [7].

- UCSD. The dataset consists of 2000 frames from a surveillance video camera on the UCSD campus. The resolution of each frame is 158 × 238. The average number of people in each frame is 25. The dataset provides the region of interest to ignore the background. Following ref. [11], frames #601 to #1400 are used for training and the rest for testing. To satisfy the constraints of the backbone on the shape of the input tensor, we resize the resolution of the image to \(400 \times 608\). This operation can not only meet the input restrictions but also ensure that the image content is not distorted.

If the image in the dataset has the various resolution, the original image will be cropped into \(400 \times 400\) patches. We convert the label to a density map by a fixed-size Gaussian kernel \(G_\sigma(\sigma = 4)\). We use the delta function \(\delta(x - x_i)\) to represent the head position. The ground truth of density map \(Y\) is generated by convolving the Gaussian kernels with each delta function.

\[
Y = \delta(x - x_i) \ast G_\sigma
\]

The parameters of the network are randomly initialized by the Gaussian distribution with mean zero and standard deviation of 0.01 except for the backbone network. We use Adam with a learning rate of 1e-4 and a batch size of 16 to train the network.

B. EVALUATION METRICS

The performance of the model is evaluated by two metrics, the Mean Absolute Error (MAE) and Mean Square Error (MSE). MAE reflects the accuracy of the results predicted by the model and MSE indicates the robustness of the model. They are defined as follows:

\[
\text{MAE} = \frac{1}{N} \sum_{i=1}^{N} \left| C_i - C_{i}^{\text{GT}} \right|
\]

\[
\text{MSE} = \sqrt{\frac{1}{N} \sum_{i=1}^{N} \left| C_i - C_{i}^{\text{GT}} \right|^2}
\]

where \(N\) is the number of the test dataset, \(i\) is the index of the image, \(C_i\) is the estimated number of the image \(i\) and \(C_{i}^{\text{GT}}\) is the ground truth.

C. EXPERIMENTAL RESULTS

In Table 2−4, we compare the performance of our method with several advanced works on ShanghaiTech [7], UCF_CC_50 [32] and UCSD [33]. These tables demonstrate that our method achieves the best MAE and MSE on these three benchmark datasets. Compared with the most advanced
PACNN, the MAE and MSE of our FFANet on Shanghai Tech Part A decreased from 66.3 to 62.4 and RMSE from 106.4 to 102.6, which reveals that the accuracy and robustness of our method are competitive. On the challenging UCF_CC_50, our FFANet outperforms PACNN by 6.2% in MAE, which states that our method is equally effective on a dataset with a small sample size where the number of people changes dramatically. Table 4 shows that our FFANet also achieved the best performance in sparse scenarios. The above results prove that our FFANet has the advantages of accuracy and robustness in both dense and sparse scenes.

To comprehensively evaluate the performance of FFANet and other models, we further verify the results of the proposed FFANet on ShanghaiTech Part A dataset in terms of model parameters, runtime and whether to load the pre-training model. The results on other measures are shown in Table 5. Compared to PACNN with the second-best MAE as shown in Table 5, FFANet achieves higher accuracy with fewer parameters. However, there is still a certain gap between FFANet and the best model [15] in terms of parameters and runtime. In future work, our research direction is to use lightweight technology to reduce the amount of FFANet network parameters and maintain the inference accuracy.

Comparing our FFANet with CSRNet, a better performance on ShanghaiTech Part A and PASCNN, FFANet has achieved higher accuracy. However, there is still a certain gap between FFANet and the best model [15] in terms of parameters and runtime. In future work, our research direction is to use lightweight technology to reduce the amount of FFANet network parameters and maintain the inference accuracy.

Compared with CSRNet, FFANet can capture the spatial distribution of the crowd and generate a clearer density map. Fig. 6 shows some high-quality results of FFANet on three datasets.

## D. ABLATION STUDIES

We present the ablation studies about the hyperparameter \( \lambda \), density map generation, different combinations of CBAM and network and the structure of the network.

1) ABLATION EXPERIMENTS ON LAMBDAN

\( \lambda \) is a hyperparameter that is used to balance the MSE loss and SSIM loss. To analyze the impact of the SSIM loss function on the results, we set different \( \lambda \) values to observe the performance of the FFANet on ShanghaiTech Part A. Fig. 7 shows that when \( \lambda = 100 \) the FFANet achieves the best results.

2) ABLATION EXPERIMENTS ON DENSITY MAP GENERATION

In this experiment, we compare the effects of three values of Gaussian kernel which are commonly used to generate density maps in crowd counting tasks on counting results. Table 6 shows that when \( \sigma = 4 \), the performance of the FFANet is the best.

3) ABLATION EXPERIMENTS ON CBAM ATTENTION MODULE

This part of the study is to discuss the impact of CBAM [25] on the networks. Table 7 shows the performance of different combinations of CBAM and network on the ShanghaiTech Part A. FFANet (Vanilla) indicates FFANet without CBAM, FFANet (Full) means FFANet with CBAM inserted in the backbone network and FFAM. FFANet indicates that the network structure proposed in this article only inserts CBAM in FFAM. Compared with FFANet (Vanilla), the MAE and MSE of FFANet decreased from 64.5 to 62.4 and MSE from 107.9 to 102.6. Furthermore, the network parameters and runtime of FFANet increased by 0.7M and 6ms compared with FFANet (Vanilla). Compared with FFANet (Full), FFANet has achieved a comprehensive lead in all measures. After considering the performance of crowd estimation accuracy

### TABLE 2. Results on the ShanghaiTech dataset.

| Methods          | ShanghaiTech A | ShanghaiTech B |
|------------------|----------------|----------------|
|                  | MAE     | MSE     | MAE     | MSE     |
| Switch-CNN [13]  | 90.4    | 135.0   | 21.6    | 33.4    |
| CP-CNN [34]      | 73.6    | 106.4   | 20.1    | 30.1    |
| IG-CNN [35]      | 72.5    | 118.2   | 13.6    | 21.1    |
| ic-CNN [36]      | 69.8    | 117.3   | 10.7    | 16.0    |
| CSRNet [16]      | 68.2    | 115.0   | 10.6    | 16.0    |
| PCC Net [15]     | 73.5    | 124.0   | 11.0    | 19.0    |
| PACNN [17]       | 66.3    | 106.4   | 8.9     | 13.5    |
| FFANet           | **62.4**| **102.6**| **8.3**| **11.1**|

### TABLE 3. Results on the UCF_CC_50 dataset.

| Methods          | MAE     | MSE     |
|------------------|---------|---------|
| Switch-CNN [13]  | 318.1   | 439.2   |
| CP-CNN [34]      | 295.8   | 320.9   |
| IG-CNN [35]      | 291.4   | 349.4   |
| ic-CNN [36]      | 260.9   | 365.5   |
| CSRNet [16]      | 266.1   | 397.5   |
| PCC Net [15]     | 240.0   | 315.5   |
| PACNN [17]       | 241.7   | 320.7   |
| FFANet           | **226.8**| **316.4**|

### TABLE 4. Results on the UCSD dataset.

| Methods          | MAE     | MSE     |
|------------------|---------|---------|
| MCNN [7]         | 1.60    | 3.31    |
| Onoro et al. [12]| 1.51    | -       |
| Switch-CNN [13]  | 1.62    | 2.10    |
| Huang et al. [37]| 1.00    | 1.40    |
| CSRNet [16]      | 1.16    | 1.47    |
| SPN [38]         | 1.03    | 1.32    |
| FFANet           | **0.97**| **1.30**|
FIGURE 5. Qualitative results on the ShanghaiTech Part A dataset. Column #1: the images from ShanghaiTech Part A; Column #2: the ground truth density maps; Column #3: estimated density maps by CSRNet; Column #4: estimated density maps by our method.

TABLE 6. Results of density map generation on the ShanghaiTech.

| Values          | Part_A | Part_B |
|-----------------|--------|--------|
|                 | MAE    | MSE    | MAE    | MSE    |
| Adaptive-kernel [7] | 66.1   | 110.5  | 8.6    | 11.7   |
| Fixed-kernel ($\sigma = 16$) | 68.8   | 106.3  | 9.0    | 11.9   |
| Fixed-kernel ($\sigma = 4$)   | 62.4   | 102.6  | 8.3    | 11.1   |

TABLE 7. Results of different combinations of CBAM and network on the ShanghaiTech Part A.

| Method           | MAE    | MSE    | Parameters(M) | Runtimes(ms) |
|------------------|--------|--------|---------------|--------------|
| FFANet (Vanilla) | 64.5   | 107.9  | 16.9          | 86           |
| FFANet (Full)    | 63.8   | 109.5  | 17.9          | 101          |
| FFANet           | 62.4   | 102.6  | 17.6          | 92           |

and computational cost, this paper chooses FFANet as the optimal network.

4) ABLATION EXPERIMENTS ON NETWORK STRUCTURE

We study the effects of fusing different levels of features on the accuracy of crowd counting. Table 8 represents the performance of networks with different structures on the ShanghaiTech Part A. Method I means to connect the last layer of VGG16 to the compression module to generate a crowd density map. Method II means that the features extracted from VGG16 are fused in one stage without FFAM and the fused features are connected with the compression module to generate density maps. Method III is similar to Method II, but the extracted features need two-stage feature fusion. Compared with Method I, the MAE and MSE of Method II decreased from 73.6 to 69.1 and MSE from 120.9 to 113.8,
which indicates that feature fusion can effectively improve the counting accuracy. Moreover, the network parameters and runtime of method II are increased by 8.7% and 5.6% compared with Method I. Compared with Method II, the MAE and MSE of Method III reduced from 69.1 to 65.9 and MSE from 113.8 to 113.2. The results show that the performance gain is due to the increase of parameters caused by the stacking of the feature fusion stage, which strengthens the expression ability of the network. However, simply increasing the network parameters will also increase the training difficulty and weaken the robustness of the network. Compared with Method II, the network parameters and runtime of method III are greatly increased, but the MSE of Method III is only decreased by 0.6. The proposed FFAM can be a nice tradeoff between crowd estimation performance and computational cost.

To evaluate the performance of the FFAM, we added FFAM to Method II and III in the feature fusion stage, resulting in Method IV and V. Compared with Method II, the MAE and MSE of Method IV decreased from 69.1 to 66.5 and MSE from 113.8 to 105.6. Moreover, in terms of network parameters and runtime, Method IV only increases 0.6M and 1ms compared to Method I. In comparison with Method III, the MAE and MSE of Method V decreased from 65.9 to 62.4 and MSE from 113.2 to 102.6. The decrease of MAE indicates that FFAM can solve the scale variation from 113.8 to 113.2.

| Method                                      | MAE  | MSE   | Parameters(M) | Runtimes(ms) |
|---------------------------------------------|------|-------|---------------|--------------|
| I. VGG16 + CM                               | 73.6 | 120.9 | 14.9          | 71           |
| II. VGG16 + stage 1 + CM (w/o FFAM)         | 69.1 | 113.8 | 16.2          | 75           |
| III. VGG16 + stage 1, 2 + CM (w/o FFAM)     | 65.9 | 113.2 | 17.0          | 87           |
| IV. VGG16 + stage 1 + CM (w FFAM)           | 66.5 | 105.6 | 16.6          | 76           |
| V. FFANet (VGG16 + stage 1, 2 + CM (w FFAM))| **62.4** | **102.6** | 17.6          | **92**       |
in an image. Meanwhile, the significant reduction in RMSE indicates that FFAM can well solve the scale changes in the dataset. The above experimental results show that FFAM is effective in dealing with scale changes in crowd counting. For the computational costs, Method V brings an increase of parameters with 0.6M and 5ms for runtime. In conclusion, FFAM can greatly improve the counting accuracy and enhance the robustness of the network under the premise of adding limited parameters.

V. CONCLUSION

In this paper, we proposed a Feature Fusion Attention Network (FFANet) to accurately estimate the number of people in the images. On one hand, the Feature Fusion Attention Module (FFAM) is proposed to realize the information enhancement of the multi-level features which are extracted by the VGG16 network. On the other hand, the enhanced features are processed by the Compression Module (CM) to generate a density map. Evaluation of the algorithm performances in comparison with other state-of-the-art methods indicates that the proposed FFANet is effective for crowd counting.

In near future, we plan to verify the adaptability of our method on other feature extractors. In addition, the performance of FFANet on UCF_CC_50 is still not perfect. This will be another research content we improve. Finally, we also plan to use model lightweight technology to reduce the time complexity of FFANet.

REFERENCES

[1] 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), Oct. 2017, pp. 3667–3676.
[2] Y. Hu, H. Chang, F. Nian, Y. Wang, and T. Li, “Dense crowd counting from still images with convolutional neural networks,” J. Vis. Commun. Image Represent., vol. 38, pp. 530–539, Jul. 2016.
[3] H. Lu, Z. Cao, Y. Xiao, B. Zhuang, and C. Shen, “TasselNet: Counting maize tassels in the wild via local counts regression network,” Plant Methods, vol. 13, no. 1, p. 79, Nov. 2017.
[4] M.-R. Hsieh, Y.-L. Lin, and W. H. Hsu, “Drobo-based object counting by spatially regularized regional proposal network,” in Proc. IEEE Int. Conf. Comput. Vis. (ICCV), Oct. 2017, pp. 4145–4153.
[5] Q. Wang, J. Gao, W. Lin, and X. Li, “NWPU-crowd: A large-scale benchmark for crowd counting and localization,” 2020, arXiv:2001.03360. [Online]. Available: http://arxiv.org/abs/2001.03360
[6] P. Dollar, C. Wojek, B. Schiele, and P. Perona, “Pedestrian detection: An evaluation of the state of the art,” IEEE Trans. Pattern Anal. Mach. Intell., vol. 34, no. 4, pp. 743–761, Apr. 2012.
[7] Y. Zhang, D. Zhou, S. Chen, S. Gao, and Y. Ma, “Single-image crowd counting via multi-column convolutional neural network,” in Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR), Jun. 2016, pp. 589–597.
[8] L. Yan, B. Fan, H. Liu, C. Huo, S. Xiang, and C. Pan, “Triplet adversarial domain adaptation for pixel-level classification of VHR remote sensing images,” IEEE Trans. Geosci. Remote Sens., vol. 58, no. 5, pp. 3558–3573, May 2020.
[9] H. Liu, X. Tang, and S. Shen, “Depth-map completion for large indoor scene reconstruction,” Pattern Recognit., vol. 99, pp. 1–11, Mar. 2019, doi: 10.1016/j.patcog.2019.107112.
[10] H. Liu, Z. Xiao, B. Fan, H. Zeng, Y. Zhang, and G. Jiang, “PyGCN: Probability prediction with graph convolutional network for person re-identification,” Neurocomputing, vol. 423, pp. 57–70, Jan. 2021.
[11] X. Cao, Z. Wang, Y. Zhao, and F. Su, “Scale aggregation network for accurate and efficient crowd counting,” in Proc. Eur. Conf. Comput. Vis. (ECCV), Munich, Germany, Sep. 2018, pp. 734–750.
[12] D. Oforo-Rubio and R. J. López-Sastre, “Towards perspective-free object counting with deep learning,” in Proc. Eur. Conf. Comput. Vis. (ECCV), Springer, Amsterdam, Netherlands, 2016, pp. 615–629.
[13] D. B. Sam, S. Surya, and R. V. Babu, “Switching convolutional neural network for crowd counting,” in Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR), Jul. 2017, pp. 4031–4039.
[14] D. Guo, K. Li, Z.-J. Zha, and M. Wang, “DANet: Dilated-Attention-Deformable ConvNet for crowd counting,” in Proc. 27th ACM Int. Conf. Multimedia, Oct. 2019, pp. 1823–1832.
[15] J. Gao, Q. Wang, and X. Li, “PCC Net: Perspective crowd counting via spatial convolutional network,” IEEE Trans. Circuits Syst. Video Technol., vol. 30, no. 10, pp. 3486–3498, Oct. 2020.
[16] 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., Jun. 2018, pp. 1091–1100.
[17] M. Shi, Z. Yang, C. Xu, and Q. Chen, “Revisiting perspective information for efficient crowd counting,” in Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR), Jun. 2019, pp. 3225–3234.
[18] X. Jiang, L. Zhang, P. Lv, Y. Guo, R. Zhu, Y. Li, Y. Pang, X. Li, B. Zhou, and M. Xu, “Learning multi-level density maps for crowd counting,” IEEE Trans. Neural Netw. Learn. Syst., vol. 31, no. 8, pp. 7205–7215, Aug. 2020.
[19] J. U. Kim and Y. Man Ro, “Attentive layer separation for object classification and object localization in object detection,” in Proc. IEEE Int. Conf. Image Process. (ICIP), Sep. 2019, pp. 3995–3999.
[20] M. Shaikh, V. A. Kollerathu, and G. Krishnamurthi, “Recent attention mechanism networks for enhanced classification of biomedical images,” in Proc. IEEE 16th Int. Symp. Biomed. Imag. (ISBI), Apr. 2019, pp. 1260–1264.
[21] Z. Lan, Q. Huang, F. Chen, and Y. Meng, “Aerial image semantic segmentation using spatial and channel attention,” in Proc. IEEE 4th Int. Conf. Image, Vis. Comput. (ICIVC), Jul. 2019, pp. 316–320.
[22] H. Li, J. Wu, L. Wu, J. Huang, J. Chen, and P. Li, “Self residual attention network for deep face recognition,” IEEE Access, vol. 7, pp. 55159–55168, 2019.
[23] J. Hu, L. Shen, and G. Sun, “Squeeze-and-excitation networks,” in Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit., Jun. 2018, pp. 7132–7141.
[24] S. Woo, J. Park, J.-Y. Lee, and I. S. Kweon, “CBAM: Convolutional block attention module,” in Proc. Eur. Conf. Comput. Vis. (ECCV), Munich, Germany, Sep. 2018, pp. 3–19.
[25] J. Park, S. Woo, J.-Y. Lee, and I. So Kweon, “BAM: Bottleneck attention module,” 2018, arXiv:1807.06514. [Online]. Available: http://arxiv.org/abs/1807.06514
[26] X. Li, W. Wang, X. Hu, and J. Yang, “Selective kernel networks,” in Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR), Jun. 2019, pp. 510–519.
[27] Y. Cao, J. Xu, S. Lin, F. Wei, and H. Hu, “GCNet: Non-local networks meet squeeze-excitation networks and beyond,” in Proc. IEEE/CVF Int. Conf. Comput. Vis. Workshop (ICCVW), Oct. 2019, pp. 1971–1980.
[28] X. Wang, R. Girshick, A. Gupta, and K. He, “Non-local neural networks,” in Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit., Jun. 2018, pp. 7794–7803.
[29] S. Ioffe and C. Szegedy, “Batch normalization: Accelerating deep learning training by reducing internal covariate shift,” 2015, arXiv:1502.03167. [Online]. Available: http://arxiv.org/abs/1502.03167
[30] K. Simonyan and A. Zisserman, “Very deep convolutional networks for large-scale image recognition,” 2014, arXiv:1409.1556. [Online]. Available: https://arxiv.org/abs/1409.1556
[31] H. Idrees, I. Saleemi, C. Seibert, and M. Shah, “Multi-source multi-scale counting in extremely dense crowd images,” in Proc. IEEE Conf. Comput. Vis. Pattern Recognit., Jun. 2013, pp. 2547–2554.
[32] A. B. Chan, Z.-S. John Liang, and N. Vasconcelos, “Privacy preserving crowd monitoring: Counting people without people models or tracking,” in Proc. IEEE Conf. Comput. Vis. Pattern Recognit., Jun. 2008, pp. 1–7.
[33] V. A. Sindagi and V. M. Patel, “Generating high-quality crowd density maps using contextual pyramid CNNs,” in Proc. IEEE Int. Conf. Comput. Vis. (ICCV), Oct. 2017, pp. 1861–1870.
[35] D. B. Sam, N. N. Sajjan, R. V. Babu, and M. Srinivasan, “Divide and
grow: Capturing huge diversity in crowd images with incrementally grow-
ing CNN,” in Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.,
Jun. 2018, pp. 3618–3626.

[36] V. Ranjan, H. Le, and M. Hoai, “Iterative crowd counting,” in Proc. Eur.
Conf. Comput. Vis. (ECCV), Munich, Germany, Sep. 2018, pp. 270–285.

[37] S. Huang, X. Li, Z. Zhang, F. Wu, S. Gao, R. Ji, and J. Han, “Body structure
aware deep crowd counting.” IEEE Trans. Image Process., vol. 27,
no. 3, pp. 1049–1059, Mar. 2018.

[38] X. Chen, Y. Bin, N. Sang, and C. Gao, “Scale pyramid network for
crowd counting,” in Proc. IEEE Winter Conf. Appl. Comput. Vis. (WACV),
Jan. 2019, pp. 1941–1950.

ZHANQIANG HUO received the B.Sc. degree in
mathematics and applied mathematics from the
Hebei Normal University of Science and Technol-
yogy, China, in 2003, and the M.Sc. degree in com-
puter software and theory and the Ph.D. degree
in circuit and system from Yanshan University,
China, in 2006 and 2009, respectively. He is cur-
rently an Associate Professor with the College of
Computer Science and Technology, Henan Poly-
technic University, China. His research interests
include computer vision and machine learning.

AIZHONG MI received the B.Sc. degree in com-
puter and application from the Hefei University
of Technology, China, in 1999, the M.Sc. degree
in computer application technology from Guangxi
University, China, in 2005, and the Ph.D. degree
in computer application technology from the Uni-
versity of Science and Technology Beijing, China,
in 2009. He is currently an Associate Professor
with the College of Computer Science and Tech-
nology, Henan Polytechnic University, China. His
research interests include computer vision and pattern recognition.

BIN LU received the B.S. degree from Hei-
longjiang University, Harbin, China, in 2015. He
is currently pursuing the master’s degree with
Henan Polytechnic University. His research inter-
est includes deep learning applications in com-
puter vision, pattern recognition with specific
interests in crowd analysis, and semantic segmen-
tation.

FEN LUO received the bachelor’s degree in com-
puter application from the Jiaozuo Institute of
Technology, China, in 2002, and the master’s
degree in resource development and planning from
the China University of Mining and Technology,
Beijing, in 2010. She is currently an Internal Asso-
ciate Professor with the College of Computer Sci-
ence and Technology, Henan Polytechnic Univer-
sity, China. Her research interests include imaging
processing and virtual reality technology.

YINGXU QIAO received the B.Sc. degree in com-
puter science and technology and the M.Sc. degree
in controlling theory and engineering from Henan
Polytechnic University, China, in 2003 and 2007,
respectively, where he is currently pursuing the
Ph.D. degree in mining information engineering.
His research interests include computer vision and
machine learning.

* * *