



HRANet: Hierarchical region-aware network for crowd counting

Jinyang Xie¹ · Lingyu Gu¹ · Zhonghui Li² · Lei Lyu^{1,3}

Accepted: 23 November 2021 / Published online: 2 February 2022

© The Author(s), under exclusive licence to Springer Science+Business Media, LLC, part of Springer Nature 2022

Abstract

Aiming to tackle the most intractable problems of scale variation and complex backgrounds in crowd counting, we present an innovative framework called Hierarchical Region-Aware Network (HRANet) for crowd counting in this paper, which can better focus on crowd regions to accurately predict crowd density. In our implementation, first, we design a Region-Aware Module (RAM) to capture the internal differences within different regions of the feature map, thus adaptively extracting contextual features within different regions. Furthermore, we propose a Region Recalibration Module (RRM) which adopts a novel region-aware attention mechanism (RAAM) to further recalibrate the feature weights of different regions. By the integration of the above two modules, the influence of background regions can be effectively suppressed. Besides, considering the local correlations within different regions of the crowd density map, a Region Awareness Loss (RAL) is designed to reduce false identification while producing the locally consistent density map. Extensive experiments on five challenging datasets demonstrate that the proposed method significantly outperforms existing methods in terms of counting accuracy and quality of the generated density map. In addition, a series of specific experiments in crowd gathering scenes indicate that our method can be practically applied to crowd localization.

Keywords Crowd counting · Multi-scale feature · Density map estimation · Attention mechanism

1 Introduction

Aiming to estimate the number of persons in dense and complex scenes, crowd counting is receiving increasing academic attention due to its great value in real-world applications such as public safety [1, 2], intelligence gathering [3, 4], emergency management [5, 6], and data analysis [7, 8]. What is more, to prevent the spread of COVID-19 virus in public places, it can be used to perform social distance monitoring by providing effective crowd size information [9].

Motivated by the recent successful use of Convolutional Neural Network (CNN) in various computer vision tasks, most crowd counting methods are inclined to adopt CNN-based

methods. Despite the impressive advancements have been achieved, these approaches suffer from two intractable problems. On the one hand, the size of individuals in the image tends to vary with the distance from the camera, which leads to the scale variation problem [10], as illustrated in the top of Fig. 1. On the other hand, the background regions in the crowd image have a similar appearance or color with the foreground regions, which is not conducive to accurate identification of crowd regions [11], as illustrated in the bottom of Fig. 1.

To address the above problems, numerous solutions are put forward from different technical perspective [12, 13], such as multi-column architectures [14, 15], adaptive feature pyramids [16, 17], and multiple CNNs with different sizes of receptive fields [18, 19]. In addition, considering the low quality of the input images, some methods try to pre-process the input images by introducing meta-heuristic-based image processing algorithms [20–25] and image optimization algorithms [26, 27], thus improving the processing efficiency of the crowd images. Though these schemes can mitigate these problems to some extent, it is still challenging to achieve high-precision crowd counting in extremely complex scenarios due to the presence of various background interferences. To be specific, firstly, the influence of background regions

✉ Zhonghui Li
sdmost@126.com

✉ Lei Lyu
lvlei@sdu.edu.cn

¹ School of Information Science and Engineering, Shandong Normal University, 250358 Jinan, China

² Jinan Rail Transit Group Engineering Research Consultation Co., Ltd, 250101 Jinan, China

³ Shandong Provincial Key Laboratory for Distributed Computer Software Novel Technology, 250358 Jinan, China

Fig. 1 Two problems that impact the accuracy of crowd counting. Top: Scale variation. Bottom: Complex background



is not fully considered. These methods are not targeted to suppress the weight of the background region in the feature map, so that features irrelevant with the crowd can easily be extracted from the background regions [28]. In this case, it is necessary to further calibrate the weights of different regions in the feature map, thus suppressing the representation ability of the background regions. Secondly, the local correlation is ignored. Due to the presence of scale variation, different crowd regions in density map contain different scale information, which is local correlation [29]. However, the Euclidean loss commonly used by existing methods does not effectively capture local correlation, so that the generated crowd density maps have poor local continuity. Therefore, we can devise a new loss function that can intentionally learn local correlations on different regions.

According to the above analysis, we propose a Hierarchical Region-Aware Network (HRANet) based on the multi-level architecture, which can adaptively recalibrate the weights of different local regions to better focus on the crowd regions. In our implementation, to capture the internal differences within different regions in crowd images, we design a Region-Aware Module (RAM), which can flexibly extract region adaptive features at each level. Additionally, to further suppress the weight of background regions in these features, we implement a region-aware attention mechanism on the Region Recalibration Module (RRM) to capture the relative influence of region features on different image locations. Based on this influence, RRM can recalibrate the weights of different regions. By combining the above two components, our HRANet can

effectively suppress the influence of background regions on the counting results. In addition, we design a Region Awareness Loss (RAL) to enforce our model to learn global and local correlations within different regions. Specifically, we divide the ground-truth and the generated density map into different size sub-regions and further compute the structural similarity between these sub-regions, so that our method can produce locally consistent density maps and reduce false identifications.

In general, the contributions of our work are threefold:

- (1) We design a Region-Aware Module (RAM), which captures the internal variation of the crowd image to adaptively extract contextual information of different regions.
- (2) We design a Region Recalibration Module (RRM), which adopt a novel region-aware attention mechanism to capture the relative influence of regional features on different image locations, thus recalibrating the weights of local regions at each level.
- (3) We design a Region Awareness Loss (RAL), which can enforce the proposed model to learn global and local correlations within different sizes of regions to produce locally consistent density maps.

The organization of the rest of the paper is as follows. Section 2 illustrates the related work associated with our method. Section 3 describes our method in detail. Section 4 performs detailed comparative experiments.

Section 5 analyzes the superiority of our method. Section 6 concludes the whole paper.

2 Related work

2.1 Crowd counting

Motivated by the successful use of deep learning, many CNN-based methods [30, 31] have widely developed in areas of crowd counting and have made significant gains. In this study, we focus on two categories of approaches associated with our work: multi-column based method and multi-level based method.

2.1.1 Multi-column based method

Multi-column based methods usually adopt multiple branches with different sized convolution kernels to encode different receptive fields, so that they can capture multi-scale features. As a pioneering work that employs multiple columns, MCNN [32] extracts multi-scale features by utilizing three branches with convolution kernels of different sizes. In the same way, CP-CNN [33] adopts two branches to extract the local and global contextual information in feature maps, respectively, and further combines them to generate the density maps. DeepCount [34] learns density information from multiple branches to train a density-aware regressor. DSSINet [29] uses a message-passing mechanism to refine the feature information between three branches. DADNet [35] introduces dilated CNN with different rates on the multi-column structure to capture more multi-scale contextual information. McML [36] introduces a statistical network, so that it can estimate the mutual information between different scales.

In general, these multi-column based methods can effectively extract multi-scale features from different branches to accommodate crowds on different scale. However, the network architecture of each branch in these methods are almost identical, which inevitably leads to significant information redundancy.

2.1.2 Multi-level based method

Multi-level based methods exploit the hierarchical structure of CNN to capture multi-scale features from within the backbone network. For example, ANF [37] introduces Conditional Random Fields (CRFs) to aggregate multi-level features. It also incorporates non-local attention mechanisms as inter- and intra-layer attention to expand the receptive fields between different levels. MBTTBF [38] introduces a bi-directional feature fusion method in the multi-level network to facilitate the full fusion of multi-level features. TEDNet

[39] proposes a decoder with dense skip connections in the multi-level structure to hierarchically aggregate multi-level features. SaCNN [40] extracts multi-level features from the backbone FCN hierarchically and resizes them to the same size for generating crowd density maps.

This kind of methods effectively extract features at different scales from the network, which significantly reduces the computational effort of the model. However, since the low-level features extracted by these methods contain more edge information, which may interfere with the final counting results.

2.2 Loss function

Loss functions aim to measure the difference between the predicted results and the real data, thus guiding the model to achieve better optimization results. The loss function commonly used in crowd counting is the MSE loss [41], which is mainly used to evaluate the variability of the data. After that, considering the local similarity between different regions due to scale variation, some researches [42, 43] start to design different similarity preserving metrics [44, 45] and similarity losses [28, 39] to reduce the difference between the ground-truth and the estimated density map. DSSINet [29] designs a Dilated Multiscale Structural Similarity (DMS-SSIM) loss to learn local correlations within regions of different sizes. Specifically, the DMS-SSIM is computed by measuring the structural similarity between the multi-scale region centered at the given pixel on the estimated density map and the ground-truth. TEDNet [39] proposes a combinatorial loss to learn local correlations and spatial correlations in density maps. CFANet [28] designs a novel structural loss function (SL), which can improve both the structural similarity and counting accuracy between the estimated density map and the ground-truth.

3 Proposed method

In this section, we first provide the overview of the proposed Hierarchical Region-Aware Network (HRANet) and then describe each module in detail.

3.1 Overview

In our implementation, the HRANet adopts an encoder-decoder architecture base on the UNet [46], whereby we can obtain a high-resolution estimated density map, and meanwhile retains more spatial detail information. As illustrated by Fig. 2, our HRANet consists of four components: encoder, RAM, RRM, and decoder.

To flexibly capture deeper features, we adopt the first 13 layers pre-trained VGG-16 as encoder by virtue of its

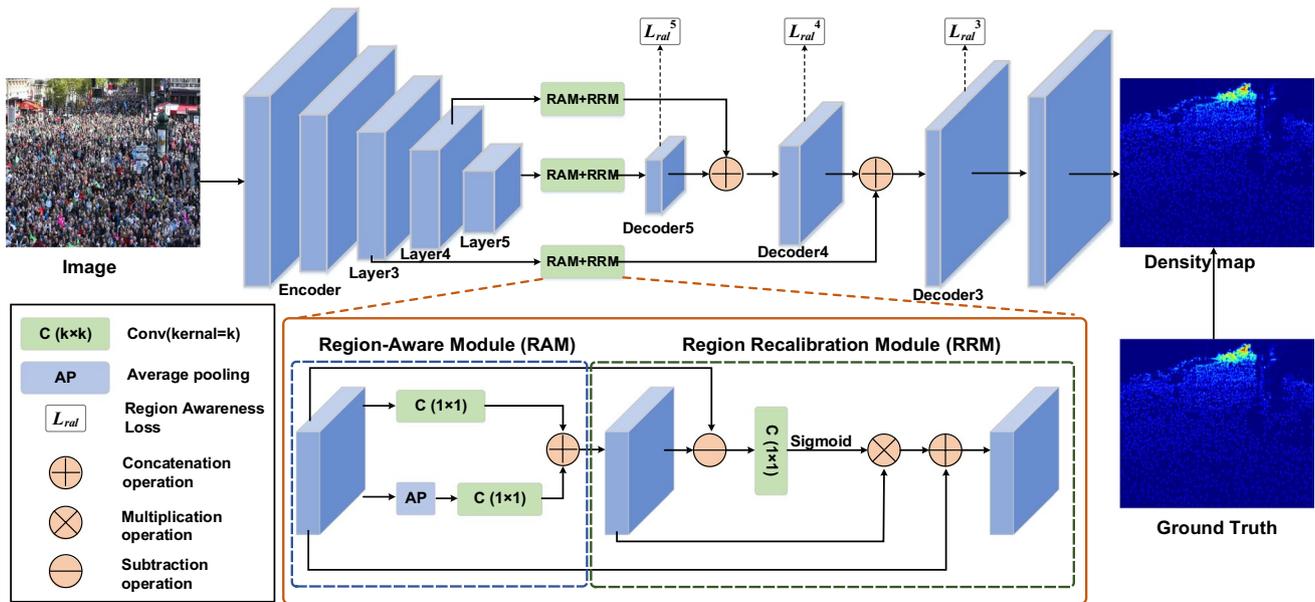


Fig. 2 The architecture of Hierarchical Region-Aware Network (HRANet). The HRANet consists of four main components: Encoder, Region-Aware Module (RAM), Region Recalibration Module (RRM), and Decoder. First, the encoder is used to extract the multi-level feature representation of the given image. Then the RAM

extracts adaptive contextual features of different regions at each level. After that, the RRM is used to further recalibrate the feature weights on different regions. Finally, multiple decoders are used to aggregate the multi-level features hierarchically to generate the final crowd density map

adaptive structure for follow-up feature fusion. Given a crowd image I , the encoder output the multi-level features $\{F_3^1, F_4^1, F_5^1\}$, which contain the contextual information at different scales.

Due to the fact that feature F_i^1 at each level can only encode a single size of receptive field, leading to its insufficient expression capacity. To remedy this, we design the RAM to extract wider contextual information, and the RRM to recalibrate weights for these contextual information, thus improving the expression capability of feature F_i^1 . To be specific, the RAM is responsible for capturing the internal differences within different regions in multi-level features F_i^1 and generating region adaptive features F_i^2 . Subsequently, the RRM is utilized to further capture the relative influence of F_i^2 on each image location to recalibrate the weight of different regions, and generate final output features F_i^3 .

Based on the final output features, we apply multiple decoders to aggregate them hierarchically. Here, each decoder is composed of a bi-linear up-sampling operation and a 1×1 convolutional operation.

$$F_i^3 = Conv(Concat(F_i^3, U_{bi}(F_{i+1}^3))) \quad (1)$$

where U_{bi} represents the up-sampling operation, $Concat()$ represents the concatenation operation, $Conv()$ represents the 1×1 convolutional operation.

Finally, we apply a set of convolutional layers to regress the aggregated features to produce the crowd density map.

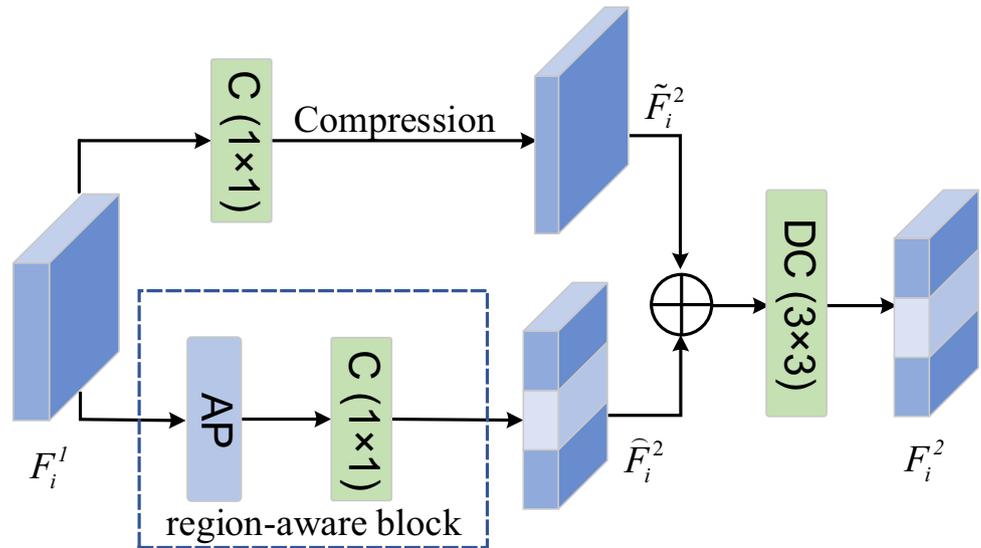
The detailed structure of these convolutional layers is: $C(128, 3)$ - U - $C(64, 3)$ - $C(1, 3)$, where $C(m, n)$ represents the convolutional layer, m represents the number of channels, and n represents the size of the convolutional kernel. U represents the bilinear interpolation up-sampling operation.

3.2 Region-aware module

RAM aims to adaptively extract contextual features on different regions. As illustrated in Fig. 3, the RAM consists of two branches, which are responsible for extracting global contextual features and region adaptive features, respectively.

In the first branch, we first squeeze the number of channels of F_i^1 to preserve the representation power of global contextual features by utilizing a 1×1 convolutional layer. Hereby, we can obtain the global contextual features F_i^2 . In the second branch, we design a region-aware block, consisting of a 3×3 average pooling layer followed by a 1×1 convolutional layer, to capture the internal differences within different regions of the crowd image. Specifically, we first divide the multi-level features F_i^1 into different regions using an average pooling operation. Then, the output features are convolved by a 1×1 convolution to generate the region-aware features \hat{F}_i^2 , which contain adaptive contextual information of different regions. Afterwards, we combine the global contextual features \tilde{F}_i^2 and the region-aware features \hat{F}_i^2 by using the channel

Fig. 3 The details of Region-Aware Module (RAM). \oplus represents the channel concatenation operation



concatenation operation. Final, the final features are passed through a 3×3 depthwise convolution to obtain the region features F_i^2 :

$$F_i^2 = Dconv(Concat(\tilde{F}_i^2, \hat{F}_i^2)) \tag{2}$$

where $Concat()$ is the concatenation operation, $Dconv()$ is the 3×3 depthwise convolutional operation.

3.3 Region recalibration module

To suppress the influence of background features on counting results and further improve the representation ability of multi-level features, RRM is designed to recalibrate the weights of different regions in the feature map. The RRM is implemented with a Region-aware Attention Mechanism (RAAM), as shown in Fig. 4.

Taking region features F_i^2 and multi-level features F_i^1 as input, RAAM first captures the relative influence of region

features on each image location by computing the contrast features F_i^3 :

$$\tilde{F}_i^3 = F_i^2 - F_i^1 \tag{3}$$

Then, we send these contrast features \tilde{F}_i^3 into a 1×1 convolutional network followed by a sigmoid activation layer to calculate weights W_i of different regions:

$$W_i = Sig(Conv(\tilde{F}_i^3)) \tag{4}$$

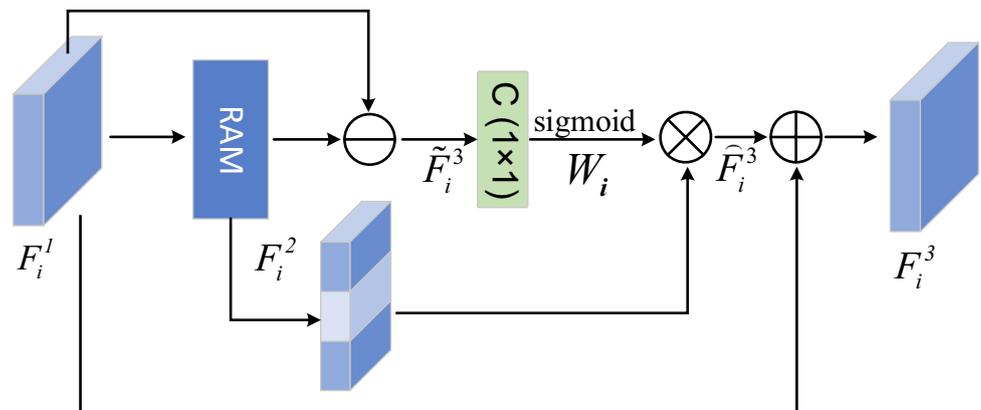
where $Sig()$ is the sigmoid activation function, $Conv()$ is the 1×1 convolutional operation.

To obtain the recalibration features \hat{F}_i^3 , an elemental multiplication operation between the region features F_i^2 and the weights W_i .

$$\hat{F}_i^3 = W_i \otimes F_i^2 \tag{5}$$

where \otimes is elemental multiplication between the weights W_i and the contextual features F_i^2 .

Fig. 4 The details of Region-Aware Attention Mechanism (RAAM). \oplus represents the channel concatenation operation, \ominus represents the subtraction operation, and \otimes represents the elemental multiplication operation



Finally, we combine the recalibration feature \hat{F}_i^3 and the multi-level feature F_i^1 using the channel concatenation operation to obtain the final feature F_i^3 , which contains rich global information and local region adaptive information:

$$F_i^3 = \hat{F}_i^3 + F_i^1 \tag{6}$$

3.4 Loss function

During training, we adopt MSE loss [41, 47] to measure the difference between the estimated density map and ground-truth, which is expressed as

$$L_{den} = \frac{1}{N} \sum_{i=1}^N \|D_i - D_i^{gt}\|_2^2 \tag{7}$$

where N is the number of images, D_i is the density map of the given image I , and D_i^{gt} is the ground-truth.

In addition, considering the local correlation within different size regions of the density map, we also introduce a Region Awareness Loss (RAL) at each level to guide the proposed model for learning the scale diversity at the local regions, as shown in Fig. 5.

Specifically, we first use a 1×1 convolution on the output features of each level to obtain the prediction X_i of of this level, and perform up-sampling operation on X_i to enforce it has the same size as the ground-truth Y . Then the ground-truth Y is divided into three layers of $1 \times 1, 2 \times 2$, and 4×4 sub-regions sequentially by applying different average pooling layers. Additionally, the same operation is performed on X_i . Finally, the structural similarity loss between X_i and Y on each sub-region is computed separately. Thereby, the L_{ral} is obtained by aggregating the losses on all layers:

$$L_{ral} = \frac{1}{N} \sum_{i=1}^N \sum_{j=1}^S \frac{1}{k_j^2} (1 - SSIM(P_{ave}(Y;k_j), P_{ave}(X_i;k_j))) \tag{8}$$

where P_{ave} denotes the average pooling operation, k_j^2 denotes the number of sub-regions, and S is the number of average pooling layers. N is the number of images. In addition, SSIM is the Structural Similarity Loss [48].

Therefore, the final loss is the summation of all the above losses.

$$L_{final} = L_{den} + \lambda \sum_{i=3}^5 L_{ral}^i \tag{9}$$

where $\lambda=0.01$ is the scale-specific weight, which is used to balance the influence of L_{ral}^i .

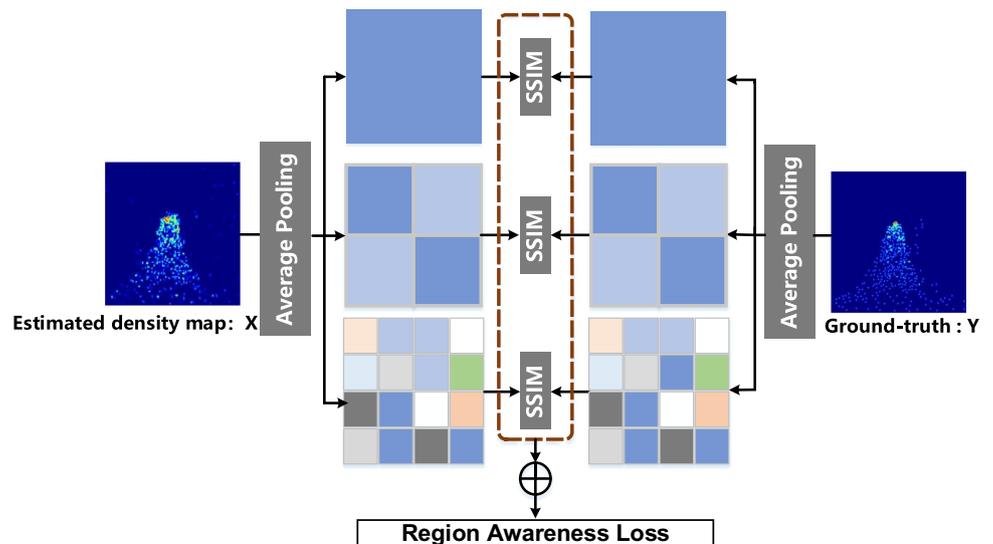
4 Experiments

In this section, we validate the performance of the proposed method on five publicly available crowd counting datasets in several aspects, and compared with existing methods to validate the novelty of the proposed method.

4.1 Implementation details

Ground-truth generation We use a Gaussian kernel (normalized to 1) to generate ground-truth by blurring each head annotation in the crowd image. Specifically, for sparse crowd scenes, we follow the method of CSRNet [49] which uses a fixed Gaussian kernel to produce the ground-truth. In addition, for dense crowd scenes, we use an adaptive Gaussian kernel to produce ground-truth $F(x)$:

Fig. 5 The details of Region Awareness Loss(RAL). \oplus represents the summation operations



$$F(x) = \sum_{i=1}^N \delta(x - x_i) \times G_{\sigma_i}(x), \text{ with } \sigma_i = \beta \bar{d}_i \quad (10)$$

where x is the location of the pixel in crowd image, N represents the number of head annotations. x_i is the target object in ground-truth δ , and \bar{d}_i is the average distance of its k nearest neighbors. We use the Gaussian kernel with parameter σ_i to convolve $\delta(x - x_i)$, where β is a constant. In our implementation, we set $\beta=0.3$ and $\sigma_i = 3$. Evaluation metrics The mean absolute error (MAE) and the mean square error (MSE) is used to evaluate the counting performance and robustness of our method respectively. They are defined as:

$$MAE = \frac{1}{N} \sum_{i=1}^N |z_i - \hat{z}_i| \quad (11)$$

$$MSE = \sqrt{\frac{1}{N} \sum_{i=1}^N |z_i - \hat{z}_i|^2} \quad (12)$$

where N is the total number of crowd images, z_i is ground-truth of the i -th image, \hat{z}_i is prediction of our model in the i -th image. MAE determines the accuracy of the crowd estimation and MSE determines the robustness of the crowd estimation.

To more visually evaluate the quality of the crowd density maps generated by our method, we adopt the Peak Signal-to-Noise Ratio (PSNR) and the Structural Similarity in Image (SSIM) [48] as evaluation criteria. PSNR is used to calculate the error between the corresponding pixels, SSIM is used to measure the structural similarity between the ground-truth and the estimated density map.

Implementation In our implementation, the encoder of our model consists of the first 13 convolutional layers and 3 pooling layers of VGG-16, which have been pre-trained on ImageNet. For the other convolutional layers, we use a Gaussian distribution with $\delta = 0.01$ to initialize them randomly. In addition, our up-sampling operation is based on bilinear interpolation. We also use Adam Optimizer to train our network for 200 epochs and learning rate is initially set to $1e-5$.

4.2 Datasets

Shanghai Tech [32] This dataset is one of the largest publicly available benchmark. It has 1198 images and 330,165 head annotations, which is composed of Part A and Part B. The Part A has total 482 images randomly collected from the Internet, the crowd scene is relatively dense. The part B includes total 716 images taken from a commercial street, the crowd scene is relatively sparse.

UCF_CC_50 [47] This dataset is the first publicly available and extremely challenging benchmark dataset. It has only 50 highly dense crowd images and the number of annotations for each image varies from 94 to 4543. Since this dataset has less data, in order to effectively conduct the comparison experiments, we perform 5-fold cross-validation following the method of CSRNet [49].

UCF-QNRF [50] This dataset is one of the publicly available large scale crowd image datasets, which consists of 1535 crowd images with 1251642 annotations. Since it has a wide range of people from 49 to 12,865 in each image, leading to its extremely challenging.

WorldExpo'10 [51] This dataset is a publicly large-scale crowd counting benchmark, it includes 1,132 annotated video sequences, each of which has 50 frame rates and its resolution is 576×720 pixels. In the test set, these frames have 5 scenes, each scene includes 120 images.

4.3 Counting performance in sparse crowd scenes

We validate the performance of our method on two datasets with relatively sparse crowd scenes: ShanghaiTech Part B [32] and WorldExpo'10 [51], and compare our method with some existing representative methods to analyze the advantages of our method in sparse crowd scenes.

ShanghaiTech Part B [32] The comparison results are shown in Table 1. As can be seen from the table, in terms of MAE, our method improves by 10% compared with DSSINet which has the best performance. Similarly, in terms of MSE, our method improves by 6% compared with DSSINet.

WorldExpo'10 [51] As can be seen from the table, we achieve the best average MAE performance, which is 10% higher than the second-best method. Furthermore, in scene 2, our method still outperforms the other methods, leading the second place by 5% in terms of MAE. Similarly, in scene 4, our method is 11% ahead of the second best.

In summary, our method significantly outperforms existing methods in crowd scenes with relatively low crowd density, which is attributed to our Region-Aware Module (RAM). Since our RAM can adaptively capture the internal differences within different regions, which results in our method capturing local contextual information to the maximum extent, and thus performing crowd counting more accurately.

4.4 Counting performance in dense crowd scenes

We validate the performance of our method on three datasets with dense crowd scenes: ShanghaiTech Part A [30], UCF_CC_50 [41], and UCF-QNRF [44], thus analyzing the advantages of our method in dense crowd scenes.

ShanghaiTech Part A [32] The comparison results are shown in Table 2. As can be seen from the table, compares

Table 1 Performance comparison on the dataset with sparse crowd scenes

Method	SHB		WorldExpo'10 (MAE)					
	MAE	MSE	S1	S2	S3	S4	S5	Ave
MCNN [32]	26.4	41.3	3.4	14.1	14.3	22.2	3.7	12.9
CSRNet [49]	10.6	16.0	2.9	11.5	8.6	16.6	3.4	8.6
TEDNet [39]	8.2	12.8	2.3	10.1	11.3	13.8	2.6	8.0
DSSINet [29]	6.9	10.3	1.6	9.5	9.5	10.4	2.5	6.7
HSRNet [52]	7.2	11.8	2.3	9.6	12.7	9.4	3.2	7.4
CAN [53]	7.8	12.2	2.9	12.0	10.0	7.9	4.3	7.4
SDANet [54]	8.2	12.8	2.0	14.3	12.5	9.5	2.5	8.1
DENet [55]	9.6	15.4	2.8	10.7	8.6	15.2	3.5	8.2
HRArNet	6.2	9.7	2.3	9.0	9.4	7.0	2.5	6.0

Table 2 Performance comparison on the dataset with dense crowd scenes

Method	SHA		UCF_CC		UCF-QNRF	
	MAE	MSE	MAE	MSE	MAE	MSE
MCNN [32]	110.2	173.2	377.6	509.1	277.0	426.0
CSRNet [49]	68.2	115.0	266.1	397.5	120.3	208.5
TEDNet [39]	64.2	109.1	249.4	354.5	113.0	188.0
DSSINet [29]	60.6	96.0	216.9	302.4	99.1	159.2
ASANet [56]	67.0	111.4	185.5	268.3	102.7	177.8
MTL-DB [57]	57.7	95.4	279.1	381.9	94.7	169.6
ASNet [16]	57.8	90.1	174.8	251.6	91.6	159.7
CDADNet [58]	57.3	89.8	170.5	228.7	–	–
EDENet [59]	53.7	90.1	–	–	86.6	158.5
HRArNet	52.8	87.2	160.9	235.8	84.6	146.2

with EDENet that has the best performance, our method improves the MAE by 2%, which indicates that our method has higher counting accuracy. Similarly, our method improves the MSE by 3% compared with CDADNet, which reflects the higher robustness of our method.

UCF_CC_50 [47] The comparison results are shown in Table 2. As can be seen from the table, in terms of robustness, our method achieves the second-best performance, CDADNet is 3% ahead of our method. However, in terms of counting accuracy, our method achieves optimal performance, which is 3% ahead of CDADNet.

UCF-QNRF [50] The comparison is summarized in Table 2. As can be seen from the table, our method improves MAE by 2% compared with EDENet that has the best performance. Our method also produces the best MSE, which is 8% higher than the second-best method. Moreover, compared with the recently proposed ASANet and MTL-DP, our method is 18% and 14% ahead of them in terms of MSE, respectively.

The comparison results indicate that our method still has significant advantages in dense crowded regions, which reflects the effectiveness of our Region-Aware Attention Mechanism (RAAM). Since RAAM can further recalibrate the weights of different regions in the feature map, so that

our method can focus on the crowd regions in the image and avoid the influence of complex background on the counting results.

4.5 Density map quality

In this section, we evaluate the quality of the density maps produced by using PSNR and SSIM as evaluation criteria. Specifically, we perform a comparison experiment between our method and some representative methods to analyze the superiority of the proposed method in terms of the quality of the generated density map.

As can be seen from Table 3, our method obtains the PSNR of 28.31 and the SSIM of 0.89, which improves by 9% and 7% over the second-best method. It can be seen from these results that our method is significantly better than other methods, which indicates that our method can generate higher quality crowd density maps.

4.6 Ablation study

To validate the effectiveness of each component in our network, in this section, we perform several sets of ablation experiments on the ShanghaiTech part A&B dataset [32] to

Table 3 Comparison results of density map quality on ShanghaiTech Part A dataset

Method	ShanghaiTech Part A	
	PSNR	SSIM
MCNN [32]	21.40	0.52
CSRNet [49]	23.79	0.76
SCAR [60]	23.93	0.81
SANet [61]	25.40	0.78
TEDNet [39]	25.88	0.83
CAN [53]	23.93	0.81
HRANet	28.31	0.89

analyze the influence of the different settings on the counting performance.

We apply our proposed encoder and the decoder as the backbone, and apply MSE loss as the basic loss function. After that, we gradually add the corresponding modules to verify the effectiveness of these modules on the ShanghaiTech dataset. Furthermore, we visualize some density maps generated by the corresponding components to visually compare their performance, as shown in Fig. 8.

Firstly, we verify the effectiveness of RAM and the results are shown in Table 4. Compared with Backbone individually, Backbone+RAM improves MAE by 3% and MSE by 4% on the ShanghaiTech part A dataset. In addition, Backbone+RAM improves MAE by 17% and MSE by 8% on the ShanghaiTech part B dataset. This significant improvement shows that our RAM can effectively capture the internal variation of the crowd region to extract wider contextual information. As shown in Fig. 8, compared with Backbone, Backbone+RAM can avoid the influence of the background regions to some extent, which further demonstrates the excellent performance of our RAM in extracting local region features.

Secondly, we verify the effectiveness of RRM and the results are shown in Table 4. Compared to the above two

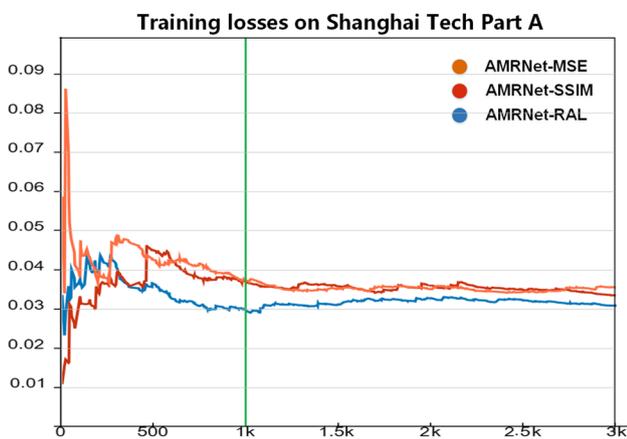
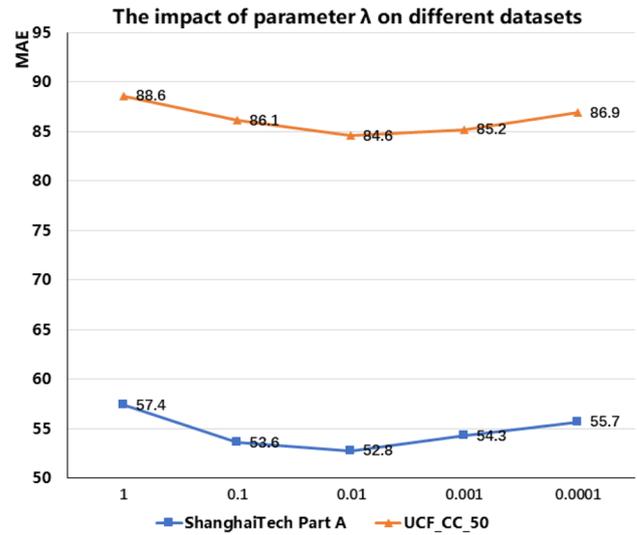
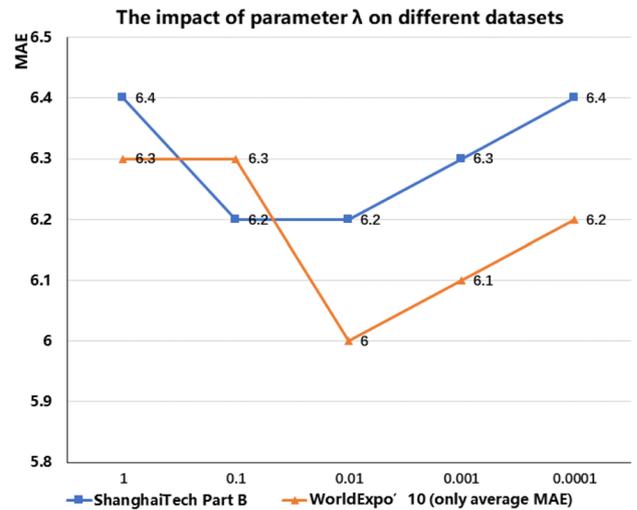


Fig. 6 The training process on the ShanghaiTech part A dataset



(a)



(b)

Fig. 7 The impact of parameter λ on different datasets. (a). The impact of λ on dense crowd scenes. (b). The impact of λ on sparse crowd scenes

versions, Backbone+RAM+RRM improves MAE by 10% and 21% respectively on the ShanghaiTech part A & B dataset. This substantial performance improvement fully validates the effectiveness of the RAAM. As seen in Fig. 8, we use RAAM to further recalibrate the weights of local regions, which results in our method better focusing on the crowd regions and avoiding the influence of background regions on the final counting results.

Thirdly, we verify the effectiveness of the RAL. Compared with using MSE as the loss function individually, using RAL to learn local correlations within different regions improves our model MAE by 10% and MSE by 8%

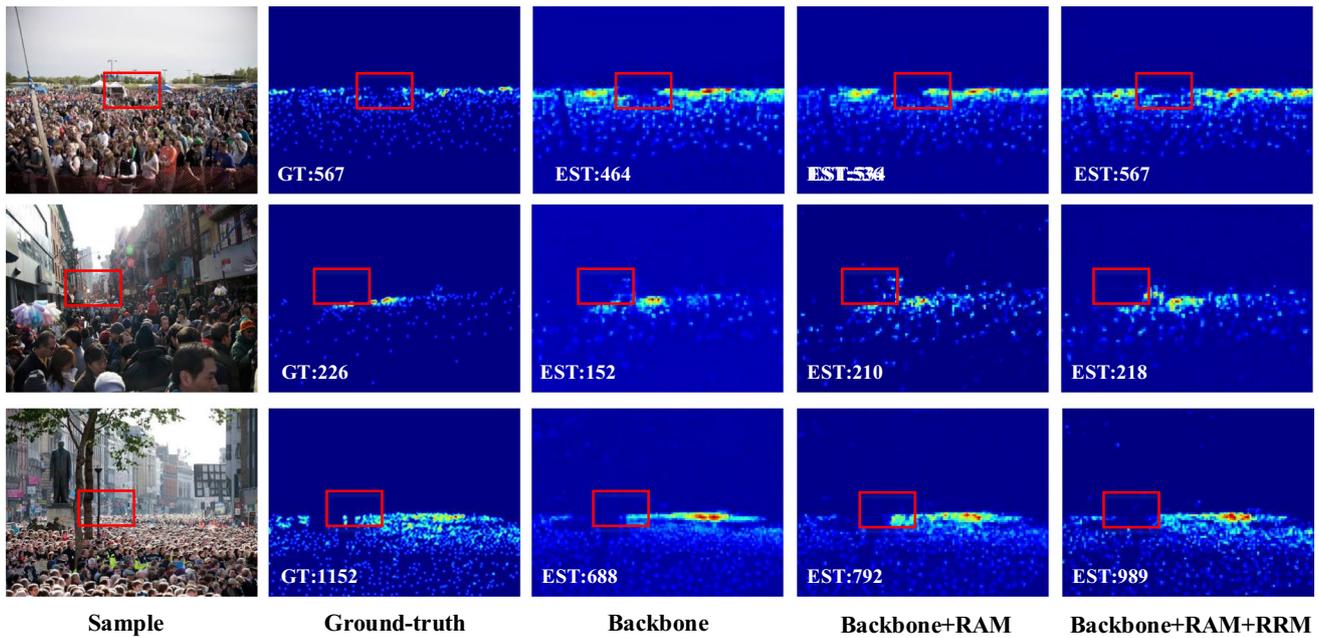


Fig. 8 Visually qualitative analysis on different crowd scenes

Table 4 The results produced by the different components of our method

Backbone	RAM	RRM	RAL	SHA		SHB	
				MAE	MSE	MAE	MSE
✓	–	–	–	68.0	113.4	10.6	16.0
✓	✓	–	–	65.9	108.3	8.7	14.6
✓	✓	✓	–	59.3	92.5	6.8	10.3
✓	✓	✓	✓	52.8	87.2	6.2	9.7

on the ShanghaiTech part A. Therefore, we can conclude that using RAL as the loss function can improve the counting performance more effectively. As shown in Fig. 6, we visually show the training process of our method with different loss functions, from which we can see that using RAL to train our method can converge faster and more smoothly.

Finally, we verify the impact of parameter λ in the loss function on the counting performance. As shown in Fig. 7, our method achieves the optimal performance in different datasets when λ is set to 0.01. Therefore, we finally choose $\lambda=0.01$ as the scale-specific weight to balance the impact of RAL on each level.

5 Analysis

In this section, we further analyze the differences between our method and existing methods. In addition, we also verify the generalization ability of our method in crowd localization.

5.1 Analysis of counting performance

We visualize some density maps generated by the proposed method and some representative methods in different crowd scenes, as shown in Fig. 9. Based on this figure, we further analyze the reasons why our method outperforms existing methods.

As shown in the yellow marked regions in the figure, when dealing with crowd scenes with large crowd scale variation, the distribution of crowds in the density map generated by our method is closer to the ground-truth. It reflects the fact that our method can respond more effectively to rapid scale variation. Since the features at different levels are complementary to each other, their combination can improve the representation ability of multi-scale features. Thus, our method hierarchically combines these features, so that our method is more robust to scale variation.

As shown in the red marked regions in the figure, when dealing with crowd scenes with complex background, our

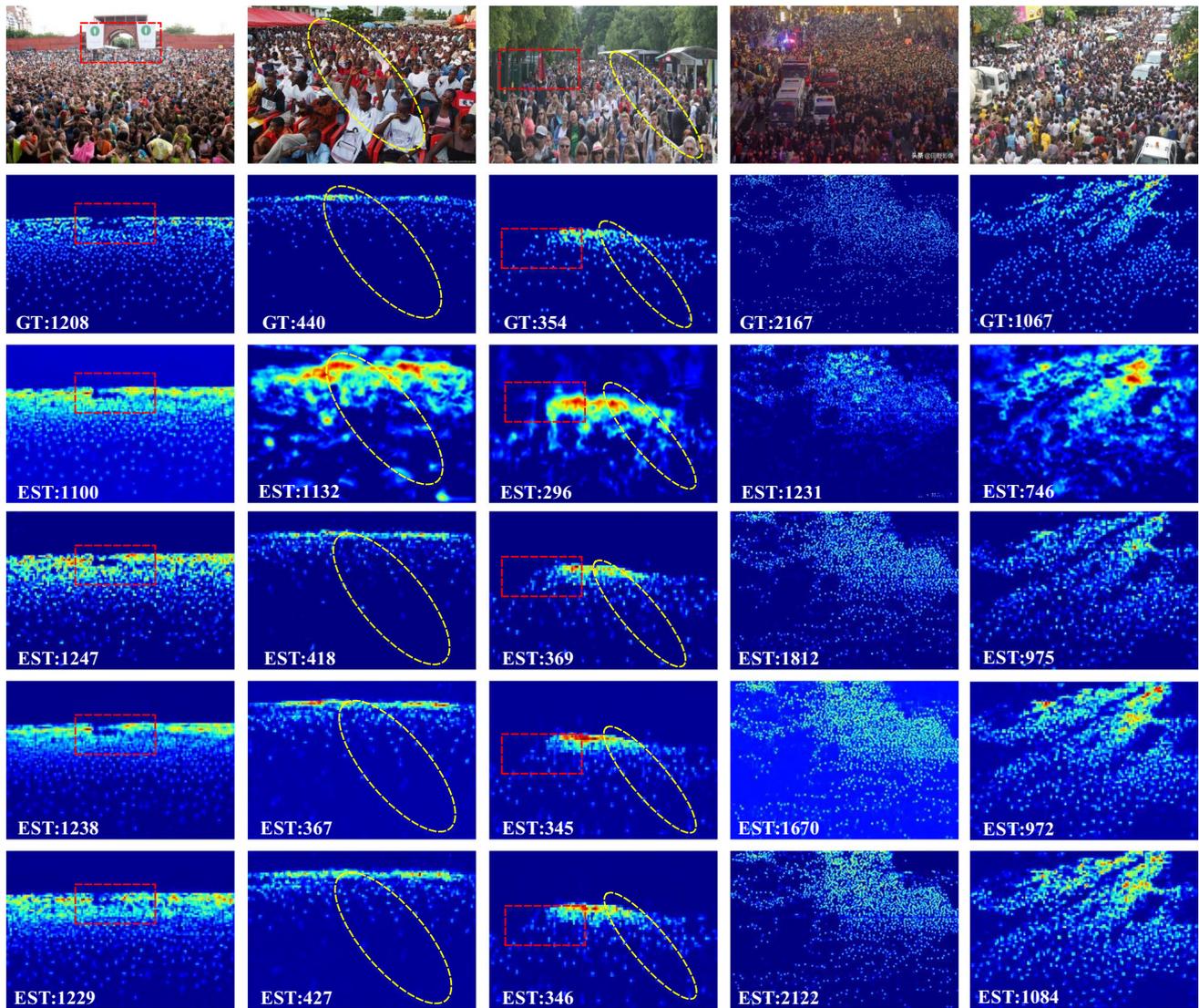


Fig. 9 Density maps generated by different methods. From top to bottom: sample, ground-truth, and the results of MCNN, SCAR, CAN, and HRANet

method can easily adapt to cluttered crowd scenes. This reflects that our method can better focus on crowd regions. Due to the presence of region-aware attention mechanism, our method can further recalibrate the weights of different regions, thus suppressing the influence of background regions on the counting results.

In the same way, as shown in the last two columns in the figure, when dealing with crowd scenes with different crowd densities, our method estimates the number of pedestrians closer to the ground-truth, which is attributed to our Region-Aware Module (RAM). Our RAM can effectively extract adaptive features within different regions, so that our method can capture wider contextual information compared with existing methods.

5.2 Analysis of generalization ability

In this section, we verify the generalization ability of our method for performing crowd localization on the NWPU dataset [8].

In our implementation, we deploy our method on the IIM framework [62] which is proposed by the Gao et al. After that, we follow existing crowd localization methods and apply Recall (Rec.), instance-level Precision (Pre.), and F1-measure to validate our method. All of the above evaluation criteria are defined in [8].

As can be seen from the Fig. 11, our method significantly outperforms the average level. This is attributed to the fact that our method can generate high-resolution feature maps, which makes our method have more spatial information to



Fig. 10 The crowd localization performance of the proposed method. The two rows from top to bottom are the crowd images and the localization maps produced by the proposed method

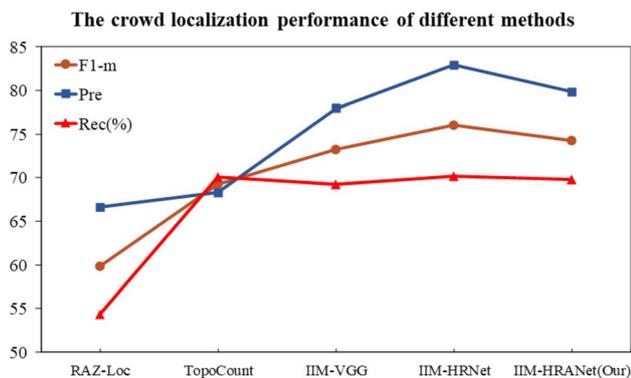


Fig. 11 Performance comparison of different methods for performing crowd localization

perform crowd localization. In addition, our region-aware attention mechanism further recalibrates the weights of different regions, so that our method can effectively avoid the interference of noisy information and accurately locate the heads of the pedestrian.

Additionally, to illustrate the performance of the proposed method in performing crowd localization, we visualize the crowd localization maps generated by our method, as shown in Fig. 10.

6 Conclusion

In this paper, we devise a Hierarchical Region-Aware Network (HRANet) for crowd counting, which addresses the crowd scale variation problem and the complex background problem from two aspects: more accurate focus on crowd regions and adequately considering local correlations on different regions. For the former, we propose a

Region-Aware Module (RAM) and a Region Recalibration Module (RRM). By integrating these two modules, our method can capture the influence of local regions on different image locations to further recalibrate the weights of different regions, thus suppressing the representation ability of background information. For the latter, we design a Region Awareness Loss (RAL), which can guide our model to learn global and local correlations in crowd density map, thus producing locally consistent density maps. Several sets of experiments on five crowd counting datasets show that proposed method can achieve more accurate counting results compared with the existing advanced methods. In addition, multiple sets of comparative experiments have fully demonstrated strong generalization ability of our method.

Acknowledgements This work was supported by the National Natural Science Foundation of China under Grant 61976127.

Declarations

Conflicts of interest Conflict of interest The authors declare that they have no conflict of interest.

References

1. Ji Q, Zhu T, Bao D (2020) A hybrid model of convolutional neural networks and deep regression forests for crowd counting. *Appl Intell*: 1–15
2. Wang Q, Han T, Gao J, Yuan Y (2021) Neuron linear transformation: Modeling the domain shift for crowd counting. *IEEE Trans Neural Netw Learn Syst*
3. Zhang G, Lu D, Liu H (2018) Strategies to utilize the positive emotional contagion optimally in crowd evacuation. *IEEE Transactions on Affective Computing* 11(4):708–721

4. Zhang G, Lu D, Liu H (2020) Iot-based positive emotional contagion for crowd evacuation. *IEEE Internet of Things Journal* 8(2):1057–1070
5. Thanasutives P, Fukui KI, Numao M, Kijisirikul B (2021) Encoder-decoder based convolutional neural networks with multi-scale-aware modules for crowd counting. In: 2020 25th international conference on pattern recognition (ICPR). IEEE, pp 2382–2389
6. Cheng ZQ, Li JX, Dai Q, Wu X, Hauptmann AG (2019) Learning spatial awareness to improve crowd counting. In: Proceedings of the IEEE/CVF international conference on computer vision, pp 6152–6161
7. Li L, Liu H, Han Y (2019) Arch formation-based congestion alleviation for crowd evacuation. *Transportation research part C: emerging technologies* 100:88–106
8. Wang Q, Gao J, Lin W, Li X (2020) Nwpu-crowd: A large-scale benchmark for crowd counting and localization. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 43(6):2141–2149
9. Liu L, Chen J, Wu H, Li G, Li C, Lin L (2021) Cross-modal collaborative representation learning and a large-scale rgbt benchmark for crowd counting. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, pp 4823–4833
10. Gao J, Wang Q, Li X (2019) Pcc net: Perspective crowd counting via spatial convolutional network. *IEEE Transactions on Circuits and Systems for Video Technology* 30(10):3486–3498
11. Gao G, Gao J, Liu Q, Wang Q, Wang Y (2020) Cnn-based density estimation and crowd counting: A survey. [arXiv:2003.12783](https://arxiv.org/abs/2003.12783)
12. Hu Y, Jiang X, Liu X, Zhang B, Han J, Cao X, Doermann D (2020) Nas-count: Counting-by-density with neural architecture search. In: *Computer vision-ECCV 2020: 16th european conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XXII* 16. Springer, pp 747–766
13. Jiang H, Jin W (2019) Effective use of convolutional neural networks and diverse deep supervision for better crowd counting. *Applied Intelligence* 49(7):2415–2433
14. Babu Sam D, Surya S, Venkatesh Babu, R (2017) Switching convolutional neural network for crowd counting. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp 5744–5752
15. Zhang A, Shen J, Xiao Z, Zhu F, Zhen X, Cao X, Shao L (2019) Relational attention network for crowd counting. In: Proceedings of the IEEE/CVF international conference on computer vision, pp 6788–6797
16. Jiang X, Zhang L, Xu M, Zhang T, Lv P, Zhou B, Yang X, Pang Y (2020) Attention scaling for crowd counting. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, pp 4706–4715
17. Sindagi VA, Patel VM (2019) Ha-ccn: Hierarchical attention-based crowd counting network. *IEEE Transactions on Image Processing* 29:323–335
18. Chen X, Bin Y, Sang N, Gao C (2019) Scale pyramid network for crowd counting. In: 2019 IEEE winter conference on applications of computer vision (WACV). IEEE, pp 1941–1950
19. Wang W, Liu Q, Wang W (2021) Pyramid-dilated deep convolutional neural network for crowd counting. *Appl Intell*: 1–13
20. Abualigah L, Diabat A, Mirjalili S, Abd Elaziz M, Gandomi AH (2021) The arithmetic optimization algorithm. *Computer Methods in Applied Mechanics and Engineering* 376:113609
21. Dinh PH (2021) Multi-modal medical image fusion based on equilibrium optimizer algorithm and local energy functions. *Appl Intell*: 1–16
22. Dinh PH (2021) Combining gabor energy with equilibrium optimizer algorithm for multi-modality medical image fusion. *Biomedical Signal Processing and Control* 68:102696
23. Dinh PH (2021) A novel approach based on three-scale image decomposition and marine predators algorithm for multi-modal medical image fusion. *Biomedical Signal Processing and Control* 67:102536
24. Dinh PH (2021) A novel approach based on grasshopper optimization algorithm for medical image fusion. *Expert Systems with Applications* 171:114576
25. Ahmadianfar I, Bozorg-Haddad O, Chu X (2020) Gradient-based optimizer: A new metaheuristic optimization algorithm. *Information Sciences* 540:131–159
26. Dinh PH (2021) An improved medical image synthesis approach based on marine predators algorithm and maximum gabor energy. *Neural Comput Appl*: 1–19
27. Jena JJ, Satapathy SC (2021) A new adaptive tuned social group optimization (sgo) algorithm with sigmoid-adaptive inertia weight for solving engineering design problems. *Multimed Tools Appl*: 1–35
28. Rong L, Li C (2021) 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
29. Liu L, Qiu Z, Li G, Liu S, Ouyang W, Lin L (2019) Crowd counting with deep structured scale integration network. In: Proceedings of the IEEE/CVF international conference on computer vision, pp 1774–1783
30. Wan J, Kumar NS, Chan AB (2021) Fine-grained crowd counting. *IEEE Transactions on Image Processing* 30:2114–2126
31. Shen Z, Xu Y, Ni B, Wang M, Hu J, Yang X (2018) Crowd counting via adversarial cross-scale consistency pursuit. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp 5245–5254
32. Zhang Y, Zhou D, Chen S, Gao S, Ma Y (2016) 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
33. Sindagi VA, Patel VM (2017) Generating high-quality crowd density maps using contextual pyramid cnns. In: Proceedings of the IEEE international conference on computer vision, pp 1861–1870
34. Chen Z, Cheng J, Yuan Y, Liao D, Li Y, Lv J (2019) Deep density-aware count regressor. [arXiv:1908.03314](https://arxiv.org/abs/1908.03314)
35. Guo D, Li K, Zha ZJ, Wang M (2019) Dadnet: Dilated-attention-deformable convnet for crowd counting. In: Proceedings of the 27th ACM international conference on multimedia, pp 1823–1832
36. Cheng ZQ, Li JX, Dai Q, Wu X, He JY, Hauptmann, AG (2019) Improving the learning of multi-column convolutional neural network for crowd counting. In: Proceedings of the 27th ACM international conference on multimedia, pp 1897–1906
37. Zhang A, Yue L, Shen J, Zhu F, Zhen X, Cao X, Shao L (2019) Attentional neural fields for crowd counting. In: Proceedings of the IEEE/CVF international conference on computer vision, pp 5714–5723
38. Sindagi VA, Patel VM (2019) Multi-level bottom-top and top-bottom feature fusion for crowd counting. In: Proceedings of the IEEE/CVF international conference on computer vision, pp 1002–1012
39. Jiang X, Xiao Z, Zhang B, Zhen X, Cao X, Doermann D, Shao L (2019) 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
40. Zhang L, Shi M, Chen Q (2018) Crowd counting via scale-adaptive convolutional neural network. In: 2018 IEEE winter conference on applications of computer vision (WACV). IEEE, pp 1113–1121

41. Wang Q, Gao J, Lin W, Yuan Y (2019) Learning from synthetic data for crowd counting in the wild. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, pp 8198–8207
42. Ma L, Li H, Meng F, Wu Q, Ngan KN (2018) Global and local semantics-preserving based deep hashing for cross-modal retrieval. *Neurocomputing* 312:49–62
43. Ma L, Li H, Meng F, Wu Q, Ngan KN (2020) Discriminative deep metric learning for asymmetric discrete hashing. *Neurocomputing* 380:115–124
44. Ma L, Li X, Shi Y, Huang L, Huang Z, Wu J (2021) Learning discrete class-specific prototypes for deep semantic hashing. *Neurocomputing* 443:85–95
45. Ma L, Li X, Shi Y, Wu J, Zhang Y (2020) Correlation filtering-based hashing for fine-grained image retrieval. *IEEE Signal Processing Letters* 27:2129–2133
46. Ronneberger O, Fischer P, Brox T (2015) U-net: Convolutional networks for biomedical image segmentation. In: International conference on medical image computing and computer-assisted intervention. Springer, pp 234–241
47. Idrees H, Saleemi I, Seibert C, Shah M (2013) 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
48. Wang Z, Bovik AC, Sheikh HR, Simoncelli EP (2004) Image quality assessment: from error visibility to structural similarity. *IEEE Transactions on Image Processing* 13(4):600–612
49. Li, Y., Zhang, X., Chen, D (2018) 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
50. Idrees H, Tayyab M, Athrey K, Zhang D, AlMaadeed S, Rajpoot N, Shah M (2018) Composition loss for counting, density map estimation and localization in dense crowds. In: Proceedings of the European Conference on Computer Vision (ECCV), pp 532–546
51. Zhang C, Li H, Wang X, Yang X (2015) Cross-scene crowd counting via deep convolutional neural networks. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp 833–841
52. Zou Z, Liu Y, Xu S, Wei W, Wen S, Zhou P (2020) Crowd counting via hierarchical scale recalibration network. [arXiv:2003.03545](https://arxiv.org/abs/2003.03545)
53. Liu W, Salzmann M, Fua P (2019) Context-aware crowd counting. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, pp 5099–5108
54. Miao Y, Lin Z, Ding G, Han J (2020) Shallow feature based dense attention network for crowd counting. *Proceedings of the AAAI conference on artificial intelligence* 34:11765–11772
55. Liu L, Jiang J, Jia W, Amirholipour S, Wang Y, Zeibots M, He X (2020) Denet: A universal network for counting crowd with varying densities and scales. *IEEE Transactions on Multimedia* 23:1060–1068
56. Chen X, Yan H, Li T, Xu J, Zhu F (2021) Adversarial scale-adaptive neural network for crowd counting. *Neurocomputing* 450:14–24
57. Liu X, Sang J, Wu W, Liu K, Liu Q, Xia X (2021) Density-aware and background-aware network for crowd counting via multi-task learning. *Pattern Recognition Letters* 150:221–227
58. Zhu A, Duan G, Zhu X, Zhao L, Huang Y, Hua G, Snoussi H (2021) Cdadnet: Context-guided dense attentional dilated network for crowd counting. *Signal Processing: Image Communication* 98:116379
59. Xia Y, He Y, Peng S, Hao X, Yang Q, Yin B (2021) Edenet: Elaborate density estimation network for crowd counting. *Neurocomputing* 459:108–121
60. Gao J, Wang Q, Yuan Y (2019) Scar: Spatial/channel-wise attention regression networks for crowd counting. *Neurocomputing* 363:1–8
61. Cao X, Wang Z, Zhao Y, Su F (2018) Scale aggregation network for accurate and efficient crowd counting. In: Proceedings of the European conference on computer vision (ECCV), pp 734–750
62. Gao J, Han T, Yuan Y, Wang Q (2020) Learning independent instance maps for crowd localization. [arXiv:2012.04164](https://arxiv.org/abs/2012.04164)



Jinyang Xie was born in Shandong, China, in 1995. He received the B. S. degree from Shandong Normal University, China, in 2019. He is currently pursuing a master's degree in the School of Information Science and Engineering, Shandong Normal University. His research interest includes computer vision, crowd counting and deep learning.



Zhonghui Li is a senior researcher of Jinan Rail Transit Group Co.,Ltd. Jinan, China. He received a Ph.D. degree in Management Science and Engineering from Shandong Normal University in 2021. His current research interests include industrial agglomeration and operations management of Rail transit.



Lingyu Gu was born in Shandong, China, in 1996. She received the B. S. degree from Taishan University, China, in 2019. She is currently pursuing a master's degree in the School of Information Science and Engineering, Shandong Normal University. Her research interest includes computer vision, crowd counting and deep learning.



Lei Lyu is an associate professor of School of Information Science and Engineering, Shandong Normal University, Jinan, China. He received a Ph.D. degree in computer application technology from University of Chinese Academy in 2013. His current research interests include computer vision and artificial intelligence.