Relational Attention Network for Crowd Counting

Anran Zhang[1], Jiayi Shen[1], Zehao Xiao[1], Fan Zhu[4], Xiantong Zhen[4], Xianbin Cao[1,2,3]*, Ling Shao[4]
[1]School of Electronic and Information Engineering, Beihang University, Beijing, China
[2]Key Laboratory of Advanced Technology of Near Space Information System (Beihang University), Ministry of Industry and Information Technology of China, Beijing, China
[3]Beijing Advanced Innovation Center for Big Data-Based Precision Medicine, Beijing, China
[4]Inception Institute of Artificial Intelligence, Abu Dhabi, UAE
zhanganran@buaa.edu.cn, shenjiayi@buaa.edu.cn, zhxiao@buaa.edu.cn,
fan.zhu@inceptioniai.org, zhenxt@gmail.com, xbcao@buaa.edu.cn, ling.shao@ieee.org

Abstract

Crowd counting is receiving rapidly growing research interests due to its potential application value in numerous real-world scenarios. However, due to various challenges such as occlusion, insufficient resolution and dynamic backgrounds, crowd counting remains an unsolved problem in computer vision. Density estimation is a popular strategy for crowd counting, where conventional density estimation methods perform pixel-wise regression without explicitly accounting the interdependence of pixels. As a result, independent pixel-wise predictions can be noisy and inconsistent. In order to address such an issue, we propose a Relational Attention Network (RANet) with a self-attention mechanism for capturing interdependence of pixels. The RANet enhances the self-attention mechanism by accounting both short-range and long-range interdependence of pixels, where we respectively denote these implementations as local self-attention (LSA) and global self-attention (GSA). We further introduce a relation module to fuse LSA and GSA to achieve more informative aggregated feature representations. We conduct extensive experiments on four public datasets, including ShanghaiTech A, ShanghaiTech B, UCF-CC-50 and UCF-QNRF. Experimental results on all datasets suggest RANet consistently reduces estimation errors and surpasses the state-of-the-art approaches by large margins.

1. Introduction

Crowd counting aims at obtaining the number of individuals in a specific scene and has a wide range of applications such as video surveillance, safety monitoring urban planning and behavior analysis. However, it is a highly challenging task due to occlusion, low image quality/resolution, perspective distortion, scale variations of objects [29, 30]. Figure 1 gives an example of a crowd image associated with density maps by different methods.

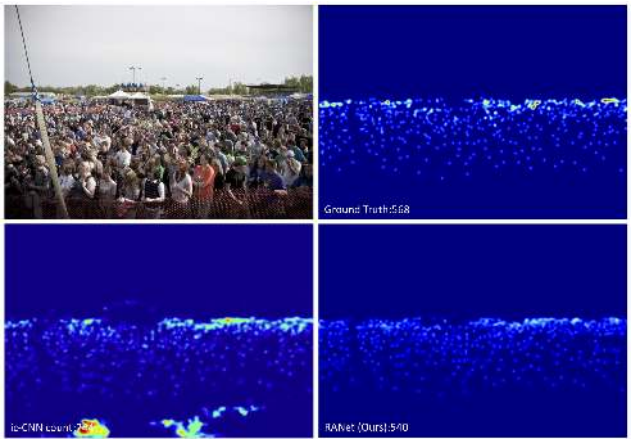

Figure 1. Density estimation results. Top Left: Input image. Top Right: Ground truth. Bottom Left: ic-CNN [28]. Bottom Right: Our RANet.

Recently, a lot of methods have been proposed to address the crowd counting problem, which mainly considers crowd counting as a pixel-wise regression problem. While the output of the unary regression for each pixel is produced independently from the outputs of the regression for other pixels, the labeling produced by the unary all regression alone is generally noisy and inconsistent. Previous works have shown that, for many pixel-level classification/regression problems, e.g., semantic segmentation, contour detection, and depth estimation, more accurate performance can be obtained by encoding interdependencies [15, 39, 13, 45]. [15] establishes pairwise potentials on all pairs of pixels in the image, enabling greatly refined segmentation and la-

*Corresponding author

beling. Pixel-centric relations [13] are collected to select the optimal affinity field size for each semantic category, which verifies the spatial structure of segmentation. This has shown the great effectiveness of modeling interdependencies of pixels for semantic segmentation which however remains largely unexplored for the task of crowd counting.

Recently, the development of deep convolutional neural networks (CNNs) has made remarkable progress in crowd counting [33, 44, 2, 12, 37]. Owing to the design of CNN structures, the convolutional operation of it is limited to having statistical efficiency by local regions [20, 36, 22]. Self-attention [34, 25, 5, 42], on the other hand, exhibits a better balance between the ability to model long-range dependencies and statistical efficiency. In contrast to the progressive behavior of convolutional operations, self-attention [42] captures long-range dependencies directly by computing interactions between any two positions and models the interdependencies of pixels. It is complementary to convolutions and helps with capturing long-range dependencies across image regions based on interdependencies of pixels.

In this paper, we propose the Relational Attention Network (RANet) for crowd counting by exploring both long- and short-range interdependencies of pixels. We extend the self-attention mechanism for complementary representations by attending to pixels both locally and globally, and we further introduce a relation module to learn correlations between attentive features, which achieves more informative representations.

Specifically, RANet incorporates a self-attention mechanism to capture interdependencies by modeling dependencies of pixels. Different from the previous work [34, 42] in which self-attention learns a weight coefficient for each neighbor by attending all positions to reconstruct the representation of each position, our RANet improve self-attention by introducing two attention modules, i.e., local self-attention (LSA) and global self-attention (GSA). To be more specific, LSA applies the self-attention on the original feature, only operating on spatially local neighbors which are closely correlated to the central position. Moreover, correlated pixels from long distances should also be taken into account for capturing interdependencies, while, to be efficient, GSA selects distinctive pixels in from the whole map by max pooling. Being computationally simple but effective, GSA provides complementary information to that of LSA, achieving more comprehensive representations.

To fuse attentive features, we further introduce a relation module to learn correlations between attentive features from LSA and GSA to achieve a unified representation. Attentive features from LSA consider short-range dependencies while those from GSA model the long-range dependencies. Thus, it is highly desired to learn relations within different attentive features. The relation module is implemented as an intra-relation module and an inter-relation module, which can exploit correlations between two attentive features not only within the same position but also across different positions. The intra-relation module aggregates feature at the same positions from LSA and GSA. The inter-relation module can learn to infer hidden relations across the holistic attentive feature for each position, and needs no supervised information that exists in position relations.

To summarize, RANet provides an improved self-attention mechanism in conjunction with relational learning to achieve informative feature representations for crowd counting. More importantly, our RANet offers a general convolutional learning architecture for pixel-level classification/regression problems, which could be readily used for diverse visual tasks. The major contributions of this work are in three folds as follows:

- We propose extending the self-attention mechanism by attending both locally and globally. We develop local self-attention (LSA) and global self-attention (GSA) to capture the interdependencies both in the local neighborhood and in long-range distinctive regions.
- We provide a relation module to learn correlations between attentive features both within and across spatial locations. Compared to simple concatenation and summation, our relation module offers an effective learnable way for feature aggregation.
- The proposed RANet has greatly advanced the state-of-the-art performance on crowd counting four public benchmark datasets. Especially, on the challenging ShanghaiTechA and UCF-QNRF datasets with dense crowds, our method surpasses the best previous method by up to 10% and 15%, respectively, in terms of MAE.

2. Related work

We briefly review recent work on crowd counting as well as the attention mechanism and graphical models.

2.1. Crowd Counting

Early works addressing the crowd counting problem major follow the strand of counting by detection. These works estimate the number of pedestrians via head or body detection [7, 8, 16]. Low-level features are then used for feature representation in detection, e.g., Haar features [35], histogram oriented gradients (HOG) [6], salient omega shape [17] and texture elements [38, 27, 1, 24]. These methods extracted features of the whole pedestrian to train their classifiers and achieved successful results in low-density crowd scenes. While objects in extremely dense crowds are hardly detected because of severe occlusions.

To handle images of dense crowds, some methods [3, 4, 14] use a regression approach to avoid the harder detection

problem. They instead extracted local patch-level features and learned a regression function to directly estimate the total count for an input image patch. Various regression techniques such as linear regression [24], piecewise linear regression [3], ridge regression [4], Gaussian process regression and neural networks [21], have been used to build a mapping from extracted features to the count number.

In recent years, density map based methods have played the main role in tackling crowd counting. Compared with transitional regression-based methods, density map based approaches have rich location information embedded in the density map. To improve the robustness of the model to variations in crowd, many CNN models are developed to combine multi-level information [44, 33, 2, 12], which has shown great effectiveness in diverse tasks [18, 40, 46, 43]. MCNN [44] employed a multi-column architecture that is designed to capture scale variation and perspective with varied receptive fields in each column. Features from these columns are fused by the 1×1 convolutional layer to regress crowd density. CSRNet [19] replaced pooling operation with dilated kernels to fuse multi-scale contextual information. CP-CNN [33] proposed a contextual Pyramid CNNs that utilized various estimators to capture global and local contexts. Contextual information is fused with high-dimensional feature maps extracted from a multi-column CNN by a fusion-CNN to generate the final prediction. Recently, ic-CNN [28] is proposed by using a multi-stage method which combined the low-resolution density map of the previous stage together with extracted features to generate high-resolution density map. SANet [2] relies on different scales of convolutional kernels to address the scale variance problem. Different from these recent methods, our work for first the time models interdependencies for pixel-wise regression in crowd counting. And we capture the interdependence of pixels by our proposed improved self-attention models instead of multi-size kernels [44, 2] or using dilated kernels [19] for pixel-level regression problems. Moreover, to aggregate attentive features, we apply a relation module that effectively learns relations in attention models for enhancing the representational power.

2.2. Self-attention Mechanism

The attention mechanism has recently drawn increasing attention in diverse vision tasks [43, 34, 41, 9, 42]. The self-attention [34] mechanism calculates the response at a position in a sequence by attending to all positions within the same sequence. This attention is used to assign importance to each type of neighbor and reconstructs feature representation. In computer vision, SAGAN [42] introduces a self-attention mechanism into CNN, it is complementary to convolutions and helps with modeling long-range, multi-level dependencies across image regions. And [26] propose an image transformer model to add self-attention into an auto-regressive model for image generation. Our method is fundamentally different from previous methods and we propose improved self-attention modules.

2.3. Graphical Models

The self-attention mechanism is also related to the graphical models [43]. Self-attention [34] allows the model to make all positions in the fully connected graph models, and to learn a weight coefficient for each neighbor. GaAN [43] proposes a graph aggregator that can be trained end-to-end to extract the local and global features across the graph. GaAN [43] models the dependencies among neighbors, which can provide more modeling power in nature. Inspired by these works, in this paper we use self-attention to calculate pairwise similarity as weight. Specifically, our RANet uses position features for similarity computations for each neighbor and reconstruct each point as a weighted sum of its neighbors. The reconstructed feature is representational with rich pairwise dependencies for improving performance over pixel-level prediction for crowd counting.

3. Relational Attention Network for Crowd Counting

The proposed relational attention network is composed of an attention module and a relation module. The attention module is derived from the self-attention mechanism to leverage its strong ability of capturing interdependency among pixels. We extend self-attention into local self-attention (LSA) and global self-attention (GSA), which can efficiently capture both long and short-range dependencies of pixels. The relation module fuses the two attentions to achieve refined more informative feature representations.

3.1. Preliminaries

We address the crowd counting task by density estimation, which is a pixel-level regression problem. The crux is to achieve informative feature representation by aggregating features from neighbors for each pixel. Our relational attention network refines those features by combining the strengths of both self-attention and relational mechanisms.

The input feature is firstly transformed into an intermediate representation by using a linear transformation matrix. And each position in the intermediate representation learns feature similarity with all neighborhoods. The normalization factor is used to normalize the similarity matrix. Finally, the newly aggregated feature is added on the original feature.

Input Feature: The input to the self-attention layer is the feature map in convolutional neural networks. We define the feature maps as $\mathbf{X} = \{\mathbf{x}_i\}_{i=1}^{N}$, where $\mathbf{x}_i$ denotes the single position feature at different positions in a feature map $\mathbf{X}$ and $\mathbf{x_i} \in R^C$ where C is the number of channels

in the input feature and N is the number of pixels in the feature map.

Intermediate Representation: In order to obtain sufficient expressive power to transform the input features into higher-level features, intermediate representations are required. A linear transformation $\mathbf{W}$ makes this manifest by assigning each position i a distinct intermediate representation, $\mathbf{X}' = \{\mathbf{x}'_i\}_{i=1}^{N}$, where $\mathbf{x}'_i \in R^{C'}$ and C' is the number of channels in intermediate representations space.

$$\mathbf{X}' = \mathbf{W}\mathbf{X} \tag{1}$$

Feature Similarity: Our self-attention layer uses position features of intermediate representations for similarity computations, which is linearly transformed by $\mathbf{W}$ in Eq. 1. The feature similarity weight $\omega_{i,j}$ indicates the impact from other positions to position i. It is computed as Eq. 2:

$$\omega_{i,j} = F(\mathbf{x}'_i, \mathbf{x}'_j) \tag{2}$$

where F is the function of feature similarity, which is defined as $F(\mathbf{x}'_i, \mathbf{x}'_j) = \mathbf{x}'^{\top}_i \mathbf{x}'_j$ [34, 42].

Similarity Normalization: A softmax operation is applied to normalize feature similarity $F(\mathbf{x}'_i, \mathbf{x}'_j)$ in Eq. 3. It indicates that in the dimension j we can compare the most activation value for each position in the dimension i.

$$\gamma^{i,j} = \mathrm{softmax}(F(\mathbf{x}'_i, \mathbf{x}'_j)) \tag{3}$$

Feature Aggregation: By considering the normalized similarity in Eq. 3 and the intermediate representation in Eq. 1, $\mathbf{z}_i$ can be aggregated as:

$$\mathbf{z}_i = \sum_{j=1}^{N} \gamma^{i,j} \mathbf{x}'_j \tag{4}$$

where $\mathbf{z}_i$ is the aggregated attentive feature at position i in the output of the single self-attention layer. Eq. 4 computes the response at a position as a weighted sum of the features at all positions, which allows all positions in the feature map to attend on each position's aggregated representation. $\mathbf{z}_i$ needs a linear transformation to restore the representation space as introduced in Eq. 1.

3.2. LSA and GSA

In regular self-attention operations, it tends to be highly computational and redundant to calculate the aggregated feature of position i by attending all positions in the feature map as shown in Eq. 4. Our RANet improves the self-attention mechanism by attending locally to its neighbors and globally to distinctive regions, which is implemented as local self-attention (LSA) and global self-attention (GSA), respectively.

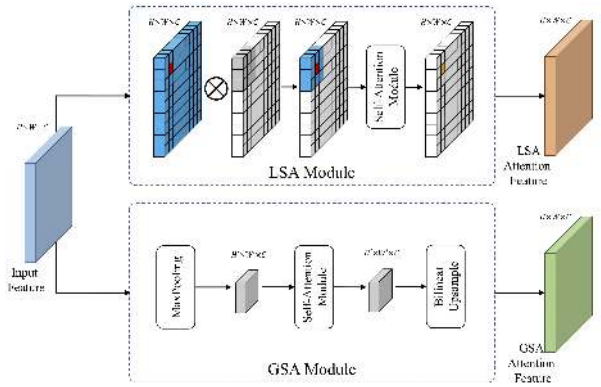

Figure 2. Upper branch: LSA learns a weight coefficient for each neighbors in the selected region to reconstruct each point as a weight sum of its neighbors. The gray part in this block is the selected local region. Bottom branch: GSA attends all positions to self-attention in a global way by max pooling.

We use LSA to consider local neighbors because the pixels that are spatially close to the current position would be more important for interdependencies compared to those far ways. While the global information would also contribute dependencies of pixels in the holistic feature map, and therefore we introduce GSA to model the long-range dependencies by selecting the most distinctive pixels using max pooling.

LSA: We define $\Omega(i)$ as the region centered at the position i spatially. For any position i, the pixel $k \in \Omega(i)$ yields attention coefficients $F(\mathbf{x}'_i, \mathbf{x}'_k)$ with sizes depending on their corresponding representations at the region $\Omega(i)$ in Eq. 5, where k enumerates all positions in the region $\Omega(i)$.

$$\mathbf{z}^l_i = \sum_{k \in \Omega(i)} \gamma^{i,k} \mathbf{x}'_k \tag{5}$$

Different from the convolutional operation in CNN, we calculate the weight coefficient for each neighbor based on the feature similarity $\gamma^{i,k}$ to reconstruct the representation of each position.

GSA: To capture full interdependencies of pixels, distant pixels to the current position also play an important role, though not all of them do. On the one hand, the distinctive neighbors in the holistic feature map contribute important correlations for each position. We introduce the global self-attention (GSA), in which we apply max pooling to select the most distinctive pixels to attend.

Specially, we first transform $\mathbf{X}'$ into $g(\mathbf{X}')$, where $g(\mathbf{X}')$ is a subsampled version of $\mathbf{X}'$ by max pooling. $g(\mathbf{X}') = \{g(\mathbf{x}'_p)\}_{p=1}^{N/N'}$, where $g(\mathbf{x}'_p)$ denotes the single position feature at position p in the subsampled feature map and N/N' denotes the number of pixels after max pooling. For each position p, we attend all positions in $g(\mathbf{x}')$ to calculate the

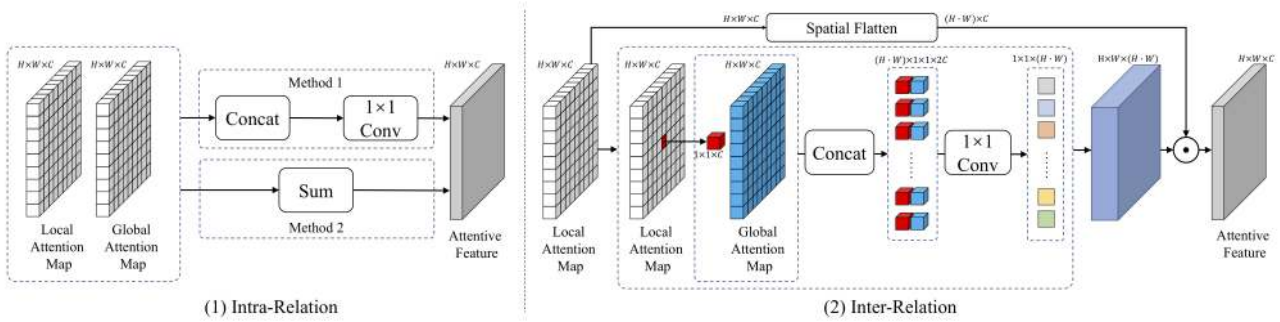

Figure 3. Relational Module: Intra-Relation and Inter-Relation. (1) Intra-Relation aggregates attentive feature from position i in the local attentive feature to position i in the global attentive feature. For the corresponding positions, we can use concatenation and summation to aggregate them. (2) Inter-Relation aggregates attentive feature from position i in the local attentive feature to all positions in the global attentive feature, and uses a relation module to infer their relations.

similarity $F(g(\mathbf{x}'_p), g(\mathbf{x}'_q))$ with *softmax* normalization, and then use Eq. 4 to calculate the aggregated feature $\mathbf{z}^g_p$.

$$g(\mathbf{z}^g_p) = \sum_{q=1}^{N/N'} g(\gamma^{p,q}) g(\mathbf{x}'_q) \tag{6}$$

which is the aggregated attentive feature for position p. GSA can reduce the amount of pairwise computation from N to N/N', and attend all the most activation value of positions in each local region for self-attention.

3.3. Relation Module

Attentive features from LSA and GSA contain different information for each position. It is important to fuse those attentive features from LSA and GSA into a more comprehensive way. Traditional methods (e.g., sum, or concatenation) aggregate information in a simple way, which may be insufficient without taking into account their correlations. We explore the deep relations in attentive features for better aggregating them to provide a learned method in relational attentive features. We introduce intra-relation and inter-relation to fuse those attentive features.

Intra-Relation: We firstly consider the intra-relation from position i in LSA to position i in GSA as shown in Eq. 7. $\mathbf{z}^l$ and $\mathbf{z}^g$ are introduced in Section. 3.2.

$$\mathbf{z} = \mathrm{ReLU}(W_c[\mathbf{z}^l, \mathbf{z}^g]) \tag{7}$$

where $[,]$ denotes element-wise (e.g., concatenation or element-wise sum operation), W_c is the convolutional operation that projects the concatenated vector to a scalar. $\mathbf{z}$ denotes aggregated information from position i in LSA and position i in GSA as shown in Figure 2.

The element-wise sum operation contains both LSA and GSA information before convolutional operation, and concatenation operation uses high-dimension feature representation across channels in CNN. The latter operation enables the network to learn more comprehensive representations, evidenced in our experimental section.

Inter-Relation: This can be inferred from the fact that intra-relation contributes point-to-point aggregated operation for two self-attention models. While it brings nothing between position i in LSA and position j in GSA. For this problem, we introduce inter-relation to model the correlation between them in Eq. 9.

As LSA only provides local aggregated information for position i, and GSA computes the self-attention in a global way. Our inter-relation provides insight by relating different aggregated self-attention information and brings both local and global features into position i. We learn the aggregated feature for each position i by inferring the relation $r(\mathbf{z}^l_i, \mathbf{z}^g_j)$ between position i in LSA and all the position j in GSA, where

$$r(\mathbf{z}^l_i, \mathbf{z}^g_j) = \mathrm{ReLU}(W_\beta[\mathbf{z}^l_i, \mathbf{z}^g_j]). \tag{8}$$

And then we combine all the positions in LSA with their relations to get the aggregated feature in Eq. 9. W_β is convolutional operation, which is implemented as, e.g., 1×1 convolution in space. This operation is shown in Figure 3 in an intermediate representation space.

$$\mathbf{z}_i = \sum_j \mathrm{ReLU}(W_\beta[\mathbf{z}^l_i, \mathbf{z}^g_j])\mathbf{z}^l_j \tag{9}$$

4. Implementation Details

In this section, we provide details about the implementation of our relational attention network.

4.1. Network Architecture

We apply the stacked Hourglass [23] in our RANet, different from recent crowd counting works [33, 19, 31]. We use the stacked Hourglass with intermediate supervision, and we add bilinear upsampling layers to ensure that the output resolution is the same as the input resolution. Here,

we apply attention modules with relation modules in the decoder of each hourglass module, and to explore the interperformance on modeling dependencies by RANet.

4.2. Training Details

During training, patches with the fixed size are cropped at random locations of original images, then they are randomly horizontal flipped for data augmentation. Density map estimation amounts to computing per-pixel density at each location in the image, thus preserving spatial information about the distribution of the crowd. We generate our ground truth density maps by fixed-size Gaussian kernels. It is required to convert these points to a density map. If there is a point at pixel x_i, it can be represented with a delta function. The ground truth density map Y is generated by using each delta function with a normalized Gaussian kernel G:

$$D(x) = \sum_{x_i \in S} \delta(x - x_i) * G_\sigma \tag{10}$$

We train the RANet in an end-to-end manner. The network parameters are randomly initialized by a Xavier with a mean zero and a standard deviation of 0.01. Adam optimizer with a small learning rate of $1e-3$ is used to train the model, and the network is trained with a batch size of 8. The implementation of our method is based on the Pytorch framework. At test time we do not extract image crops and instead we feed the whole image to the network.

4.3. Evaluation Metrics

Following previous works for crowd counting, we use the Mean Absolute Error (MAE) and Root Mean Squared Error (RMSE) to evaluate the performance of our proposed method. MAE indicates the accuracy of the predicted result and RMSE measures the robustness. If the predicted count for the image i is C_i and the ground truth count is $C_i^{'}$, the MAE and RMSE can be computed as:

$$\text{MAE} = \frac{1}{n}\sum_{i=1}^{N}|C_i - C_i^{'}| \text{ and } \text{RMSE} = \sqrt{\frac{1}{n}\sum_{i=1}^{N}|C_i - C_i^{'}|^2} \tag{11}$$

where n is number of test samples.

5. Experiments

We conduct extensive experiments on four benchmark datasets and compare the proposed RANet with state-of-art methods. The proposed RANet achieves the best performance on all datasets and largely surpasses previous methods by substantial margins. Extensive ablation studies have shown the great effectiveness of the proposed attention and relation modules.

Table 1. Evaluation of attention and relation modules (RM).

Method	MAE	RMSE
LSA w/o RM	63.6	113.7
GSA w/o RM	62.3	107.2
LSA & GSA with Intra-Relation (Sum)	63.4	113.0
LSA & GSA with Intra-Relation (Concat)	62.2	103.4
LSA & GSA with Inter-Relation	**59.4**	**102.0**

5.1. Datasets

ShanghaiTech A and B: The ShanghaiTech dataset [44] contains 1198 images, with a total of 330,165 annotated people. ShanghaiTech is a challenging dataset because it contains both high-density and low-density crowd. This dataset is divided into two parts: Part A with 482 images and Part B with 716 images. Images in Part A are randomly crawled from the Internet, most of them have a large number of people. Part B is taken from the busy streets of metropolitan areas in Shanghai. Part A contains high-density crowds, and Part B contains low-density crowds. There are tremendous occlusions for most people in each image, and the scale of the people is variable.

UCF-CC-50: The UCF-CC-50 dataset introduced in [10] contains 50 images of varying resolutions, with a wide range of densities. It is the first dataset for dense crowd images. Similar to the other counting scenes, the scenes in these images also belong to a series of different events: concerts, protests, stadiums, marathons and pilgrimages. Each image has a different resolution, and the image resolution of this dataset is rather large with an average resolution of 2101×2888. There is a large variation in crowd counts with the number of people in the image ranging from 94 to 4543. The limited number of images makes it a challenging dataset for deep learning methods.

UCF-QNRF: The UCF-QNRF [11] is the latest released dataset with one of the highest number of high-count crowd images and annotations in 2018. The number of people in the image ranges from 49 to 12865. The UCF-QNRF has the most number of high-count crowd images and annotations in all datasets. The new UCF-QNRF dataset contains buildings, vegetation, sky and roads as they are present in realistic scenarios captured in the wild, which makes it more challenging for counting.

5.2. Ablation Study

In this section, we compare the performance of our attention modules (i.e., LSA and GSA) and relation modules (i.e., intra-relation and inter-relation). We follow previous work [28, 2, 19], by conducting the ablation studies on the ShanghaiTech A dataset.

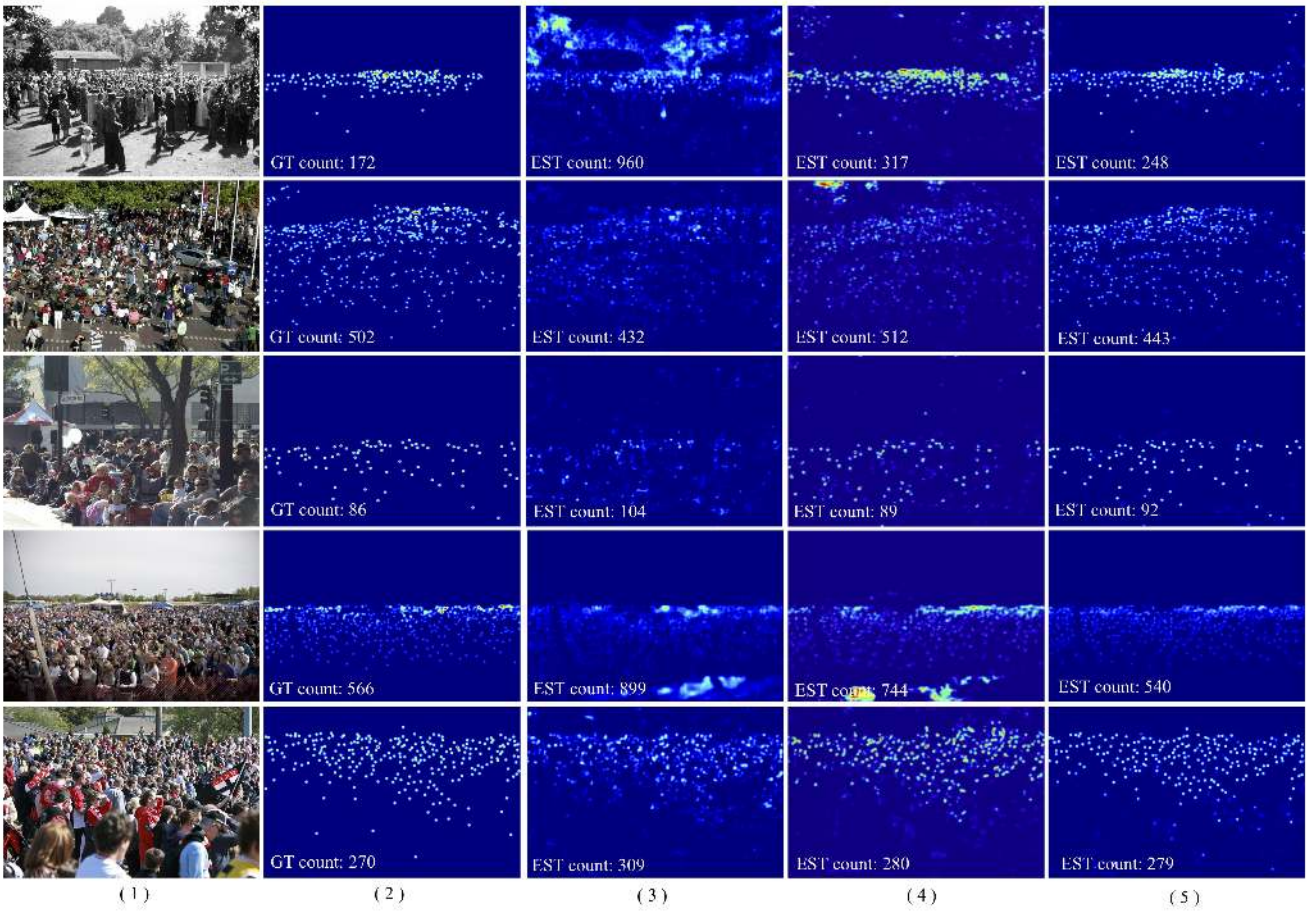

Figure 4. **Qualitative results on the Shanghaitech Part A dataset**. The five columns show that: (1) the input image,(2) the ground truth annotation map, (3) MCNN [44], (4) ic-CNN [28] , (5) RANet.

Effect of LSA and GSA. The results of LSA and GSA are summarized in Table 1. Both LSA and GSA perform well in terms of both MAE and RMSE. The results demonstrate the effectiveness of the proposed attention modules on capturing long- and short-range interdependencies of pixels.

Effect of relation modules. Our relation module offers an effective way to fuse attentive features from the attention module. The improvements of intra-relation and inter-relation are shown in Table 1. The results of intra-relation show that with a learnable concatenation operation, the performance has been largely improved by 3.5% in terms of RMSE compared with the performance of GSA, while sum operation provides negative results. This indicates that our learnable concatenation enables the network to learn more comprehensive representation for aggregating attentive features. The inter-relation achieves an improvement of 4.7% and 4.9% in terms of MAE and RMSE, respectively. This result demonstrates the effectiveness of our relation modules to fuse attentive features in a learnable way. Inter-relation learns to infer the hidden relations of positions, from each position in LSA to whole positions in GSA, which enables the network to learn parameters.

Qualitative results. Our RANet is evaluated and compared to the other seven recent state-of-the-art methods and comparison results are shown in Table 2. It indicates that our method achieves the highest performance in terms of both MAE and RMSE in all datasets compared to other methods. And our method surpasses the state-of-the-art method (SANet [2]) by up to 10% in terms of MAE. As in the previous work, SANet [2] and ic-CNN [28] conducted their ablation studies on ShanghaiTech A. Particularly, ic-CNN [28] provides more variable examples than SANet [2] with occlusion, perspective distortion and scale variations in ablations. MCNN [44] proposes the ShanghaiTech dataset and is one of the most representative methods in density estimation based crowd counting. To evaluate the quality of the generated density map, we compare our method to MCNN [44] and ic-CNN [28] using Part A dataset. Samples

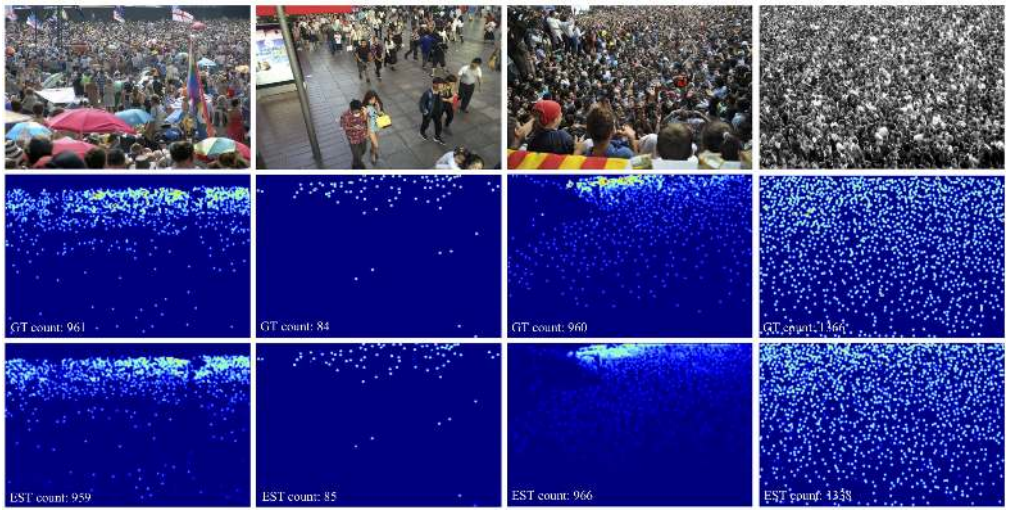

Figure 5. Estimated density maps from left to right: (1) ShanghaiTech A, (2) ShanghaiTech B, (3) UCF-QNRF, (4) UCF-CC-50.

Table 2. Performance comparison with state-of-the methods.

	ShanghaiTech PartA		ShanghaiTech PartB		UCF-CC-50		UCF-QNRF	
Method	MAE	RMSE	MAE	RMSE	MAE	RMSE	MAE	RMSE
MCNN [44]	110.2	173.2	26.4	41.3	377.6	509.1	277	426
Cascaded-MTL [32]	101.3	152.4	20.0	31.1	322.8	397.9	252	514
Switching-CNN [31]	90.4	135.0	21.6	33.4	318.1	439.2	228	445
CP-CNN [33]	73.6	106.4	20.1	30.1	295.8	320.9	-	-
ic-CNN [28]	68.5	116.2	10.7	16.0	260.9	365.5	-	-
CSRNet [19]	68.2	115.0	10.6	16.0	266.1	397.5	-	-
SANet [2]	67.0	104.5	8.4	13.6	258.4	334.9	-	-
Idrees et al [11]	-	-	-	-	-	-	132	191
RANet(Ours)	**59.4**	**102.0**	**7.9**	**12.9**	**239.8**	**319.4**	**111**	**190**

of the test cases can be found in Figure 4, which shows that our RANet can address the problem with occlusion, perspective distortion and scale variations. RANet performs better on counting the number of people in an image than MCNN [44] and ic-CNN [28]. Compared with ground truth, RANet also shows better localized predictions and is closer to the ground truth.

5.3. Comparison to other methods

We compare our RANet with previous methods for crowd counting on all datasets in Table 2, and show some sample images generated by RANet in Figure 5. We produce state-of-the-art on all four challenging datasets. As shown in Table 2, on the most popular benchmark - ShanghaiTech A, our proposed RANet achieves the highest counting accuracy and significantly improves the previously best performance from 67.0 to 59.4 in terms of MAE. Moreover, the lower RMSE - 102.0 - achieved by our RANet also indicates that it can better count the number of crowds. Especially, on UCF-QNRF which is the latest released dataset with one of the highest number of count crowd images and annotations, our RANet surpasses the best previous method by up to 15% in terms of MAE. These results show that the effectiveness of RANet by using relation modules in attention modules to model interdependencies of pixels for crowd counting.

6. Conclusion

In this paper, we have presented the Relational Attention Network (RANet) for crowd counting. RANet incorporates global and local self-attention mechanisms to capture both long- and short-range interdependence of pixels. It also provides a novel and effective way to fuse attentive features by inferring their relations in an end-to-end trainable fashion. RANet integrates attention mechanisms and relational modules to enhance feature representation for crowd counting, which achieves new state-of-the-art performence on four benchmarks.

Acknowledgment This paper was supported by National Key Research and Development Program of China under Grant 2016YFB1200100, National Key Scientific Instrument and Equipment Development Project under Grant 61827901, and Natural Science Foundation of China under Grant 91538204, 91738301, 61871016, 61571147.

References

[1] Gabriel J Brostow and Roberto Cipolla. Unsupervised bayesian detection of independent motion in crowds. In *Computer Vision and Pattern Recognition, 2006 IEEE Computer Society Conference on*, volume 1, pages 594–601. IEEE, 2006.

[2] Xinkun Cao, Zhipeng Wang, Yanyun Zhao, and Fei Su. Scale aggregation network for accurate and efficient crowd counting. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 734–750, 2018.

[3] Antoni B Chan, Zhang-Sheng John Liang, and Nuno Vasconcelos. Privacy preserving crowd monitoring: Counting people without people models or tracking. In *Computer Vision and Pattern Recognition, 2008. CVPR 2008. IEEE Conference on*, pages 1–7. IEEE, 2008.

[4] Ke Chen, Chen Change Loy, Shaogang Gong, and Tony Xiang. Feature mining for localised crowd counting. In *BMVC*, volume 1, page 3, 2012.

[5] Jianpeng Cheng, Li Dong, and Mirella Lapata. Long short-term memory-networks for machine reading. *arXiv preprint arXiv:1601.06733*, 2016.

[6] Navneet Dalal and Bill Triggs. Histograms of oriented gradients for human detection. In *Computer Vision and Pattern Recognition, 2005. CVPR 2005. IEEE Computer Society Conference on*, volume 1, pages 886–893. IEEE, 2005.

[7] Piotr Dollar, Christian Wojek, Bernt Schiele, and Pietro Perona. Pedestrian detection: An evaluation of the state of the art. *IEEE transactions on pattern analysis and machine intelligence*, 34(4):743–761, 2012.

[8] Pedro F Felzenszwalb, Ross B Girshick, David McAllester, and Deva Ramanan. Object detection with discriminatively trained part-based models. *IEEE transactions on pattern analysis and machine intelligence*, 32(9):1627–1645, 2010.

[9] Yuanjun Huang, Xianbin Cao, Xiantong Zhen, and Jungong Han. Attentive temporal pyramid network for dynamic scene classification. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, pages 8497–8504, 2019.

[10] Haroon Idrees, Imran Saleemi, Cody Seibert, and Mubarak Shah. Multi-source multi-scale counting in extremely dense crowd images. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2547–2554, 2013.

[11] Haroon Idrees, Muhmmad Tayyab, Kishan Athrey, Dong Zhang, Somaya Al-Maadeed, Nasir Rajpoot, and Mubarak Shah. Composition loss for counting, density map estimation and localization in dense crowds. *arXiv preprint arXiv:1808.01050*, 2018.

[12] Xiaolong Jiang, Zehao Xiao, Baochang Zhang, Xiantong Zhen, Xianbin Cao, David Doermann, and Ling Shao. Crowd counting and density estimation by trellis encoder-decoder networks. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2019.

[13] Tsung-Wei Ke, Jyh-Jing Hwang, Ziwei Liu, and Stella X Yu. Adaptive affinity fields for semantic segmentation. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 587–602, 2018.

[14] Dan Kong, Douglas Gray, and Hai Tao. A viewpoint invariant approach for crowd counting. In *Pattern Recognition, 2006. ICPR 2006. 18th International Conference on*, volume 3, pages 1187–1190. IEEE, 2006.

[15] Philipp Krähenbühl and Vladlen Koltun. Efficient inference in fully connected crfs with gaussian edge potentials. In *Advances in neural information processing systems*, pages 109–117, 2011.

[16] Bastian Leibe, Edgar Seemann, and Bernt Schiele. Pedestrian detection in crowded scenes. In *null*, pages 878–885. IEEE, 2005.

[17] Min Li, Zhaoxiang Zhang, Kaiqi Huang, and Tieniu Tan. Estimating the number of people in crowded scenes by mid based foreground segmentation and head-shoulder detection. In *Pattern Recognition, 2008. ICPR 2008. 19th International Conference on*, pages 1–4. IEEE, 2008.

[18] Peizhao Li, Anran Zhang, Lei Yue, Xiantong Zhen, and Xianbin Cao. Multi-scale aggregation network for direct face alignment. In *2019 IEEE Winter Conference on Applications of Computer Vision (WACV)*, pages 2156–2165. IEEE, 2019.

[19] Yuhong Li, Xiaofan Zhang, and Deming Chen. Csrnet: Dilated convolutional neural networks for understanding the highly congested scenes, 2018.

[20] Wenjie Luo, Yujia Li, Raquel Urtasun, and Richard Zemel. Understanding the effective receptive field in deep convolutional neural networks. In *Advances in neural information processing systems*, pages 4898–4906, 2016.

[21] AN Marana, L da F Costa, RA Lotufo, and SA Velastin. On the efficacy of texture analysis for crowd monitoring. In *Computer Graphics, Image Processing, and Vision, 1998. Proceedings. SIBGRAPI'98. International Symposium on*, pages 354–361. IEEE, 1998.

[22] Xin Miao, Xiantong Zhen, Xianglong Liu, Cheng Deng, Vassilis Athitsos, and Heng Huang. Direct shape regression networks for end-to-end face alignment. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 5040–5049, 2018.

[23] Alejandro Newell, Kaiyu Yang, and Jia Deng. Stacked hourglass networks for human pose estimation. In *European Conference on Computer Vision*, pages 483–499. Springer, 2016.

[24] Nikos Paragios and Visvanathan Ramesh. A mrf-based approach for real-time subway monitoring. In *Computer Vision and Pattern Recognition, 2001. CVPR 2001. Proceedings of the 2001 IEEE Computer Society Conference on*, volume 1, pages I–I. IEEE, 2001.

[25] Ankur P Parikh, Oscar Täckström, Dipanjan Das, and Jakob Uszkoreit. A decomposable attention model for natural language inference. *arXiv preprint arXiv:1606.01933*, 2016.

[26] Niki Parmar, Ashish Vaswani, Jakob Uszkoreit, Łukasz Kaiser, Noam Shazeer, Alexander Ku, and Dustin Tran. Image transformer. *arXiv preprint arXiv:1802.05751*, 2018.

[27] Vincent Rabaud and Serge Belongie. Counting crowded moving objects. In *Computer Vision and Pattern Recognition, 2006 IEEE Computer Society Conference on*, volume 1, pages 705–711. IEEE, 2006.

[28] Viresh Ranjan, Hieu Le, and Minh Hoai. Iterative crowd counting. *arXiv preprint arXiv:1807.09959*, 2018.

[29] David Ryan, Simon Denman, Sridha Sridharan, and Clinton Fookes. An evaluation of crowd counting methods, features and regression models. *Computer Vision and Image Understanding*, 130:1–17, 2015.

[30] Sami Abdulla Mohsen Saleh, Shahrel Azmin Suandi, and Haidi Ibrahim. Recent survey on crowd density estimation and counting for visual surveillance. *Engineering Applications of Artificial Intelligence*, 41:103–114, 2015.

[31] Deepak Babu Sam, Shiv Surya, and R Venkatesh Babu. Switching convolutional neural network for crowd counting. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, volume 1, page 6, 2017.

[32] Vishwanath A Sindagi and Vishal M Patel. Cnn-based cascaded multi-task learning of high-level prior and density estimation for crowd counting. In *Advanced Video and Signal Based Surveillance (AVSS), 2017 14th IEEE International Conference on*, pages 1–6. IEEE, 2017.

[33] Vishwanath A Sindagi and Vishal M Patel. Generating high-quality crowd density maps using contextual pyramid cnns. In *2017 IEEE International Conference on Computer Vision (ICCV)*, pages 1879–1888. IEEE, 2017.

[34] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. Attention is all you need, 2017.

[35] Paul Viola and Michael J Jones. Robust real-time face detection. *International journal of computer vision*, 57(2):137–154, 2004.

[36] Xiaolong Wang, Ross Girshick, Abhinav Gupta, and Kaiming He. Non-local neural networks. *arXiv preprint arXiv:1711.07971*, 10, 2017.

[37] Ze Wang, Zehao Xiao, Kai Xie, Qiang Qiu, Xiantong Zhen, and Xianbin Cao. In defense of single-column networks for crowd counting. In *British Machine Vision Conference (BMVC)*, 2018.

[38] Xinyu Wu, Guoyuan Liang, Ka Keung Lee, and Yangsheng Xu. Crowd density estimation using texture analysis and learning. In *Robotics and Biomimetics, 2006. ROBIO'06. IEEE International Conference on*, pages 214–219. IEEE, 2006.

[39] Dan Xu, Elisa Ricci, Wanli Ouyang, Xiaogang Wang, and Nicu Sebe. Multi-scale continuous crfs as sequential deep networks for monocular depth estimation. In *Proceedings of CVPR*, volume 1, 2017.

[40] Fisher Yu, Dequan Wang, Evan Shelhamer, and Trevor Darrell. Deep layer aggregation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2403–2412, 2018.

[41] Lei Yue, Xin Miao, Pengbo Wang, Baochang Zhang, Xiantong Zhen, and Xianbin Cao. Attentional alignment networks. In *British Machine Vision Conference (BMVC)*, 2018.

[42] Han Zhang, Ian Goodfellow, Dimitris Metaxas, and Augustus Odena. Self-attention generative adversarial networks. *arXiv preprint arXiv:1805.08318*, 2018.

[43] Jiani Zhang, Xingjian Shi, Junyuan Xie, Hao Ma, Irwin King, and Dit-Yan Yeung. Gaan: Gated attention networks for learning on large and spatiotemporal graphs. *arXiv preprint arXiv:1803.07294*, 2018.

[44] Yingying Zhang, Desen Zhou, Siqin Chen, Shenghua Gao, and Yi Ma. Single-image crowd counting via multi-column convolutional neural network. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 589–597, 2016.

[45] Xiantong Zhen, Mengyang Yu, Xiaofei He, and Shuo Li. Multi-target regression via robust low-rank learning. *IEEE transactions on pattern analysis and machine intelligence*, 40(2):497–504, 2017.

[46] Jiewan Zheng, Xianbin Cao, Baochang Zhang, Xiantong Zhen, and Xiangbo Su. Deep ensemble machine for video classification. *IEEE transactions on neural networks and learning systems*, 30(2):553–565, 2018.