Boosting Crowd Counting via Multifaceted Attention

Hui LIN[1], Zhiheng MA[2], Rongrong JI[3], Yaowei WANG[4], Xiaopeng HONG[1,4,5]*
[1] School of Cyber Science and Engineering, Xi'an Jiaotong University
[2] Shenzhen Institute of Advanced Technology, Chinese Academy of Science
[3] Xiamen University, [4] Peng Cheng Laboratory, [5] Harbin Institute of Technology
linhuixjtu@gmail.com; zh.ma@siat.ac.cn; rrji@xmu.edu.cn;
wangyw@pcl.ac.cn; hongxiaopeng@ieee.org

Abstract

This paper focuses on the challenging crowd counting task. As large-scale variations often exist within crowd images, neither fixed-size convolution kernel of CNN nor fixed-size attention of recent vision transformers can well handle this kind of variations. To address this problem, we propose a Multifaceted Attention Network (MAN) to improve transformer models in local spatial relation encoding. MAN incorporates global attention *from vanilla transformer,* learnable local attention*, and* instance attention *into a counting model. Firstly, the local Learnable Region Attention (LRA) is proposed to assign attention exclusive for each feature location dynamically. Secondly, we design the Local Attention Regularization to supervise the training of LRA by minimizing the deviation among the attention for different feature locations. Finally, we provide an Instance Attention mechanism to focus on the most important instances dynamically during training. Extensive experiments on four challenging crowd counting datasets namely ShanghaiTech, UCF-QNRF, JHU++, and NWPU have validated the proposed method. Code: https://github.com/LoraLinH/Boosting-Crowd-Counting-via-Multifaceted-Attention.*

1. Introduction

Crowd counting plays an essential role in congestion estimation, video surveillance, and crowd management. Especially after the outbreak of coronavirus disease (COVID-19), real-time crowd detection and counting attract more and more attention.

In recent years, typical counting methods [20, 21, 41, 50] utilize the Convolution Neural Network (CNN) as backbone and regress density map to predict the total crowd count. However, due to the wide viewing angle of cameras and the 2D perspective projection, large-scale variations often exist in crowd images. Traditional CNNs with fixed-size convolution kernel are difficult to deal with these variations and the counting performance is severely limited. To alleviate this issue, multi-scale mechanism is designed, such as multi-scale blobs [48], pyramid networks [22], and multi-column networks. These methods introduce an intuitive local-structure inductive bias [43], suggesting that the respective field should be adaptive to the size of objects.

Lately, the blossom of Transformer models, which adopt the global self-attention mechanism, has significantly improved the performances of various natural language processing tasks. Nonetheless, it is not until ViT [10] introduces patch-dividing as a local-structure inductive bias that transformer models can compete with and even surpass CNN models in vision tasks. The development of vision transformer suggests that both global self-attention mechanism and local inductive bias are important for vision tasks.

The study about transformer based crowd counting is just in its preliminary stage [19, 49] and undergoes major challenges in introducing the local inductive bias to transformer models in crowded scenes. These models usually use fixed-size attention as ViT, which is limited in encoding the 2D local structure as pointed out by [10] and clearly inadequate to handle large-scale variations of crowd images. To solve this problem, in this paper, we improve both the structure and training scheme of vision transformers for crowd counting from the following three perspectives.

Firstly, in response to such limitations in local region encoding, we propose the learnable region attention (LRA) to emphasize the local context. Different from previous vision transformers that adopt fixed patch division schemes, LRA can flexibly determine which local region it should pay attention to for each feature location. As a result, the local attention module provides an efficient way of extracting the most relevant local information against the scale changes. Moreover, it further disengages from the dependence on the position embedding module of ViT, which has been proven

*Corresponding author.

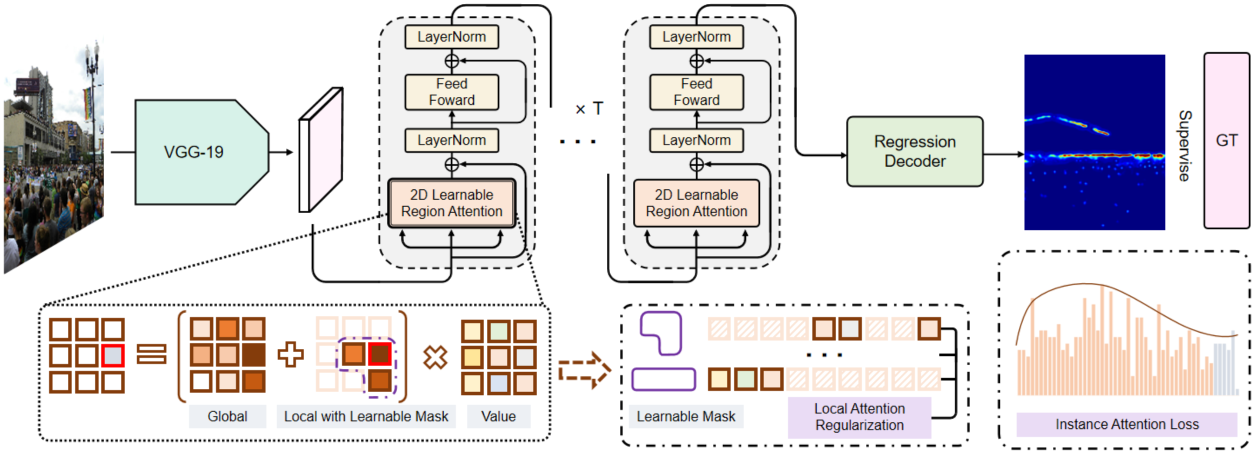


Figure 1. The framework of Multifaceted Attention Network. A crowd image is first fed into CNN. Then the flatten output feature map is transmitted into the transformer encoder with the *Learnable Region Attention*. Finally, a regression decoder predicts the density map. *Local Attention Regularization* and *Instance Attention Loss* (in lilac boxes) are optimized during the training process.

inefficient in encoding local space relations [10].

Secondly, we propose an efficient Local Attention Regularization (LAR) method to regularize the training of the LRA module. Inspired by the recent finding of human behaviors [5] that people often allocate similar attention resources to objects with similar real sizes regardless of their sizes in 2D images, we require the allocated attention *w.r.t.* each feature location to be similar. Based on this understanding, we design LAR to optimize the distribution of local attention by penalizing the deviation among them. LAR enforces the span of visual attention to be small on crowd area, and vice versa, for balanced and efficient allocations of attention.

Finally, we make an attempt to apply the attention mechanism to the instance (i.e., the point annotations) level in images and propose the *Instance Attention* module. As the point annotations as provided in popular crowd benchmarks are spare and can only occupy a very small portion of the entire human heads, there are unavoidable annotation errors. To alleviate this issue, we use *Instance Attention* to focus on the most important instances dynamically during training.

In summary, we propose a counting model with multifaceted attention, termed as Multifaceted Attention Network (MAN), to address large-scale variations in crowd images. The contributions are further summarized as follows:

- We propose the local Learnable Region Attention to allocate an attention region exclusive for each feature location dynamically.
- We design a local region attention regularization method to supervise the training of LRA.
- We introduce an effective instance attention mechanism to select the most important instances dynamically during training.
- We perform extensive experiments on popular datasets including ShanghaiTech, UCF-QNRF, JHU++, and NWPU, and show that the proposed method makes a solid advance in counting performance.

2. Related Works

2.1. Crowd Counting

Existing crowd counting methods can be categorized into three types: detection, regression and point supervision. Detection based methods [15, 18] construct detection models to predict bounding boxes for every person in the image. The final predicted count is represented by the number of boxes. However, its performance is limited by the occlusion in congested areas and the need of additional annotations.

Regression based methods [13, 50] predict count by regressing to a pseudo density map generated based on point annotations. More improvements such as multi-scale mechanisms [2, 22, 30, 48], perspective estimation [44, 46] and auxiliary task [17, 51] further promote the development of crowd counting.

Recently, many works propose to avoid the inaccurate generation of pseudo maps and directly use point supervisions. BL [21] designs the loss function based on Bayesian theory, calculating the deviation of expectation for each crowd. And further works [14, 23, 38] focus on optimal transport and measure the divergence without depending on the assumption of Gaussian distribution.

2.2. Transformer

The transformer [36] has rapidly been used in wide range of machine learning area. [9] proposes Bidirectional Encoder Representations from Transformers (BERT) to enable deep bidirectional pre-training for language representations.

[26] makes use of transformer to achieve strong natural language understanding through generative pre-training. [8] introduces a generalization of the Transformer model which extends theoretical capabilities.

The Vision Transformer (ViT) [10] firstly applies a transformer architecture for vision tasks and demonstrates outstanding performances. DETR [3] further boosts the efficiency of vision transformer focusing on object detection. More recently, these advancements have boosted the effective applications of transformer in various tasks. [32,42,53] adopt transformers on instance or semantic segmentation. [34, 52, 54] improve accuracy and efficiency for object detection. For tracking task, great properties of transformer are also leveraged in [4, 33, 39].

2.3. Variable Attention

The self-attention module is a key component in many deep learning models and especially in different kinds of transformers. In order to better leverage on the ability of relative information extraction, some previous works endow the attention module with the variable property. [47] proposes flexible self-attention module which computes attention weights over words with the dynamic weight vector. Disan [27] introduces multi-dimensional attention and directional self-attention to perform a feature-wise selection and the context-aware representations. Longformer [1] utilizes dilated sliding window attention to combine local and global information. And [25] enables more focused attentions by dynamic differentiable windows.

In vision tasks, Swin Transformer [19] designs shifted attention windows with overlap to achieve cross-window connections. The study [35] develops blocked local attention and attention downsampling to improve speed, memory usage, and accuracy. [45] proposes focal self-attention to capture both local and global interactions in vision transformers. A 2-D version of sliding window attention as Longformer [1] is introduced to achieve a linear complexity w.r.t. the number of tokens [49].

We extend the previous variable attentions to learnable one, under the premise of large scale variations in crowd images. Our proposed 2D Learnable Region Attention (LRA) breaks the constraint of traditional fixed-size local attention windows in vision tasks and is robust to scale variations.

3. The Proposed Method

In this section, we will elaborate the Multifaceted Attention Network, which consists of three major components: the Learnable Region Attention (LRA), the Local Attention Regularization (LAR), and the Instance Attention Loss.

3.1. Framework Overview

Figure 1 presents an overview of the framework. For each image I, we first use VGG-19 [28] as our backbone to extract the features $F \in \mathbb{R}^{C\times W\times H}$, where C, W, and H are the channel, width and height, respectively. Then the feature map is flattened and transmitted into transformer encoder with the proposed LRA to learn features F' against various scales. Afterwards, a regression decoder is utilized to predict the final density map $D \in \mathbb{R}^{W'\times H'}$ from F'. Finally, We use an Local Attention Regularization dedicated to supervise the training of the LRA module and an Instance Attention Loss to constrain the training of the total network.

3.2. Global Attention

Traditional transformer network [36] adopts self-attention layer in the encoder. It is able to connect all pairs of input and output positions to consider the global relations of current features. It is computed by:

$$Att(Q,K,V) = \mathcal{S}(\frac{(QW^Q)(KW^K)^T}{\sqrt{d}})(VW^V), \quad (1)$$

where $\mathcal{S}$ is the softmax function and $\frac{1}{\sqrt{d}}$ is a scaling factor based on the vector dimension d. $W^Q, W^K, W^V \in \mathbb{R}^{d\times d}$ are weight matrices for projections. Q, K, V, which are derived from source features, stand for the query, key, and value vectors, respectively.

However, since it regards the input as a disordered sequence and indiscriminately considers all correlations among features, the global attention model is position-agnostic [7]. Therefore, we propose a local learnable region attention to consider spatial information and enable more focus attentions.

3.3. Region Attention

As fixed-size convolution kernel and predesigned attentions [19, 49] are insufficient to learn cross-scale spatial information, we seek to design a mechanism by which each feature will be learnable to attend to a most suitable local region. In specific, as a rectangular region can be identified by two vertices, we begin with a region filter mechanism to obtain the exclusive region for each position.

We first define two filter functions of position $\mathbf{p} = (x_p, y_p)$ where $0 \le x_p < W, 0 \le y_p < H$ as:

$$\begin{aligned} fil^{bl}(\mathbf{p} \mid \mathbf{b}) &= \begin{cases} 1, & \text{if } x_b \le x_p < W, y_b \le y_p < H \\ 0, & \text{otherwise} \end{cases}, \\ fil^{ur}(\mathbf{p} \mid \mathbf{u}) &= \begin{cases} 1, & \text{if } 0 \le x_p \le x_u, 0 \le y_p \le y_u \\ 0, & \text{otherwise} \end{cases}. \end{aligned} \quad (2)$$

Given two predicted vertices bottom left (bl) and upper right (ur): $\mathbf{b} = (x_b, y_b)$, $\mathbf{u} = (x_u, y_u)$ for a specific feature, the filter regions for it can be calculated by:

$$R^{bl} = [fil^{bl}(\mathbf{p} \mid \mathbf{b})]_p^{W\times H}, \quad R^{ur} = [fil^{ur}(\mathbf{p} \mid \mathbf{u})]_p^{W\times H}. \quad (3)$$

After that, calculated by Hadamard product between two filter regions $R = R^{bl} \circ R^{ur}$, the region map $R \in \mathbb{R}^{W \times H}$ can finally be expressed as:

$$R(\mathbf{p}) = \begin{cases} 1, & \text{if } x_b \le x_p \le x_u, y_b \le y_p \le y_u \\ 0, & \text{otherwise} \end{cases}. \tag{4}$$

Especially, when adopting global attention, R can be represented as an all-ones matrix.

It is worth noticing that following the above-mentioned filter mechanism, the accuracy of each exclusive region entirely depends on only two discrete points, which is not learnable and lacks flexibility.

Therefore, to explore more on local relations and improve the effectiveness of learning ability, we redesign the region filter mechanism based on coverage probability projections.

3.4. Learnable Region Attention

First, given the query vector and key vector $Q, K \in \mathbb{R}^{WH \times d}$, we replace the two binary filter regions by first introducing two predicted coverage probability maps as follows:

$$\begin{aligned} C^1 &= \mathcal{S}((QW_1^Q)(KW_1^K)^T), \\ C^2 &= \mathcal{S}((QW_2^Q)(KW_2^K)^T), \end{aligned} \tag{5}$$

where $W_1^Q, W_1^K, W_2^Q, W_2^K \in \mathbb{R}^{d \times d}$ are trainable parameter matrices and $C^1, C^2 \in \mathbb{R}^{WH \times WH}$.

To obtain a 2D learnable attention map, C^1 and C^2 are reshaped to an order-3 tensor with a size of $\mathbb{R}^{WH \times W \times H}$. For each $i \in WH$ along the first axis of C^1 and C^2, there are two corresponding probability maps $\mathbf{C}_i^1, \mathbf{C}_i^2 \in \mathbb{R}^{W \times H}$. That is, $\mathbf{C}_i^1 = C^1(i, :, :)$ and $\mathbf{C}_i^2 = C^2(i, :, :)$. We then redesign the filter region maps by following the Cumulative Distribution Function (CDF) *w.r.t.* two different directions, namely from bottom left (bl) to upper right (ur) and opposite. More specifically, given a 2D probability function $\mathbf{C}_i$, for any position $\mathbf{p} = (x_p, y_p)$ where $0 \le x_p < W, 0 \le y_p < H$, we have

$$\begin{aligned} F_{\text{CDF}}^{bl}(\mathbf{p} \mid \mathbf{C}_i) &= \sum_{x_j \le x_p} \sum_{y_j \le y_p} \mathbf{C}_i(x_j, y_j), \\ F_{\text{CDF}}^{ur}(\mathbf{p} \mid \mathbf{C}_i) &= \sum_{x_j \ge x_p} \sum_{y_j \ge y_p} \mathbf{C}_i(x_j, y_j). \end{aligned} \tag{6}$$

Let $\hat{R}_i^{bl}(\mathbf{C}_i) = \left[F_{\text{CDF}}^{bl}(\mathbf{p} \mid \mathbf{C}_i)\right]_p^{W \times H}$ and $\hat{R}_i^{ur}(\mathbf{C}_i) = \left[F_{\text{CDF}}^{ur}(\mathbf{p} \mid \mathbf{C}_i)\right]_p^{W \times H}$, the learnable region map $\hat{R}_i \in \mathbb{R}^{W \times H}$ is given by:

$$\hat{R}_i = \hat{R}_i^{bl}\left(\mathbf{C}_i^1\right) \circ \hat{R}_i^{ur}\left(\mathbf{C}_i^2\right), \tag{7}$$

where $\circ$ is the Hadamard product.

Nonetheless, since we compute those two probability maps by softmax function, the two cumulative distributions $\hat{R}_i^{bl}, \hat{R}_i^{ur}$ may have a large number of zero values. Then the final region map, which is the product of above two regions, will be trivial, as illustrated by the first row of Figure 2. Therefore, we perform a reverse to guarantee no matter which the cumulative direction is chosen, $\hat{R}_i\left(\mathbf{C}_i^1\right)$ and $\hat{R}_i\left(\mathbf{C}_i^2\right)$ will have nontrivial overlap, as shown by Figure 2. The complete region map becomes:

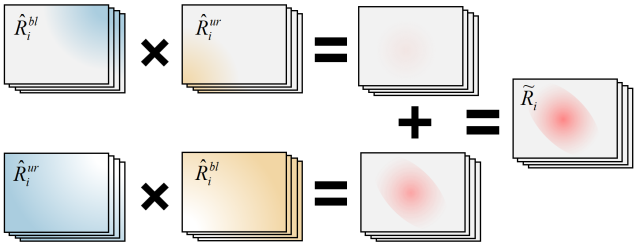


Figure 2. An example of the learnable region filter mechanism. See texts for details.

$$\widetilde{R}_i = \hat{R}_i^{bl}\left(\mathbf{C}_i^1\right) \circ \hat{R}_i^{ur}\left(\mathbf{C}_i^2\right) + \hat{R}_i^{ur}\left(\mathbf{C}_i^1\right) \circ \hat{R}_i^{bl}\left(\mathbf{C}_i^2\right). \tag{8}$$

After obtaining the learnable region map $\widetilde{R}$, we combine it into attention module for learnable local attention. With $W_{loc}^Q, W_{loc}^K \in \mathbb{R}^{d \times d}$ being specific parameter matrices of local attention, the output can be computed by:

$$Att_{lra} = \mathcal{S}\left(\frac{(QW_{loc}^Q)(KW_{loc}^K)^T \circ \widetilde{R}}{\sqrt{d}}\right)(VW^V). \tag{9}$$

Compared to R in Eq. 4 which relies on discrete vertex points, $\widetilde{R}$ is in a form of parameter arrays which are differentiable. The proposed learnable region attention mechanism is thus trainable and more flexible at determining the attentional regions.

Then the global attention is defined by sharing same refined value vectors with LRA:

$$Att_{glb} = \mathcal{S}\left(\frac{(QW_{glb}^Q)(KW_{glb}^K)^T}{\sqrt{d}}\right)(VW^V). \tag{10}$$

Finally, the output of the complete attention module is a combination of global attention and our proposed learnable region attention (LRA):

$$Att = Att_{glb} + Att_{lra}. \tag{11}$$

3.5. Local Attention Regularization

We take inspiration from the recent finding from the study of human behaviors that the human visual system usually pays similar attention to the objects with similar real sizes [5]. To mimic such a phenomenon, we design the local region attention regularization module for supervising the training of the local learnable region attention module. The goal is to balance the distributions of local attention and

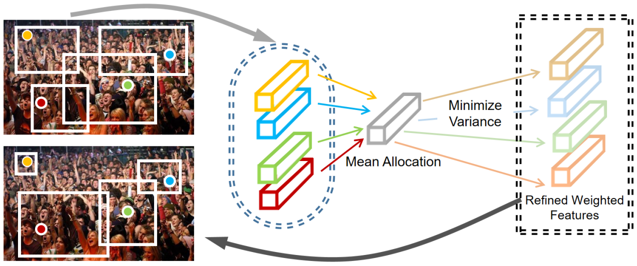


Figure 3. Overview of the Local Attention Regularization. It refines the LRA by keeping the consistency of allocated attention resources in each proposed region.

penalize the deviation among the attention allocated to local regions.

More specifically, given a predicted learnable region map $\widetilde{R}_i \in \mathbb{R}^{W \times H}$ and a feature map $F \in \mathbb{R}^{C \times W \times H}$, we compute the attention-weighted features, which can be formulated as a double tensor contraction of the second and the third mode of F with the first and second mode of $\widetilde{R}_i$ [6]:

$$E_i = [F]_{(1,[2,3])} \star [\widetilde{R}_i]_{([1,2])}. \tag{12}$$

$[\cdot]_{(\cdot)}$ indicates the mode for the tensor contraction operator $\star$ [24] and we finally have $E_i\,(c) = \sum_{w=1}^{W} \sum_{h=1}^{H} F\,(c, w, h) \cdot \widetilde{R}_i\,(w, h)\,.$

To keep the consistency of allocations of attention resources in each region, we regularize the local attention by minimizing the variance among weighted features:

$$\mathcal{R}_{lra} = \sum_{i=0}^{WH} \mathcal{G}(E_i, \overline{E}), \tag{13}$$

where $\overline{E}$ is the mean allocation of attention resources in all local attention regions. The deviation penalty $\mathcal{G}$ between two weighted features is given by:

$$\mathcal{G}(E_i, E_j) = 1 - \frac{E_i^T E_j}{||E_i|| \cdot ||E_j||}. \tag{14}$$

The scheme of Local Attention Regularization is shown in Figure 3.

3.6. Instance Attention Loss

For optimizing the entire network, we provide the Instance Attention Loss. As the ground truth as provided in popular crowd benchmarks is in a form of spare point annotations and only occupies a very small portion of human heads, this kind of human-labeled annotations inevitably exist spatial error.

To alleviate negative influences by annotation noises, we impose a dynamic selection mechanism named Instance Attention, considering that the trained model sometimes predicts more correct signals than annotations. The mechanism is designed based on an attention mask $\mathbf{m} = [m^j]_j^N$ to select supervisions. The Instance Attention Loss is defined as:

$$\mathcal{L}_{IA} = \sum_{j}^{N} m^j \cdot \epsilon^j, \tag{15}$$

where $\mathbf{e} = [\epsilon^j]_j^N$ are deviations between predictions and labels. For example, in MSE Loss, N equals to the size of density map, while in Bayesian Loss [21] (BL), N equals to the number of annotated points. Considering the performance and robustness, we finally choose BL as the deviation function:

$$\epsilon^j = |1 - \sum_{\mathbf{p}} Prob_j(\mathbf{p}) \cdot D_{\mathbf{p}}|, \tag{16}$$

where j is j_{th} annotated point. D is the final predicted density map. $Prob_j(\mathbf{p})$ represents for the posterior of the occurrence of the j_{th} annotation given the position $\mathbf{p}$.

The instance attention mask in Eq. 15 provides a mechanism to select or weigh the instances. We regard the deviation ϵ^j between predictions and labels as a kind of uncertainty of labels. If ϵ^j is too large, there is a contradiction under the label of the instance. In this case, we shall reduce its importance or exclude this instance in back-propagation dynamically. For efficient computation, we adopt $\mathbf{m}$ as binary vectors. We first get the indices that sort the deviations in ascending order $\vec{\mathbf{k}} = \text{sortID}\,(\mathbf{e})$. Then $\mathbf{m}$ is given by:

$$m^j = \begin{cases} 1, & \text{if } j \in \left\{ \vec{\mathbf{k}}(1), \vec{\mathbf{k}}(2), ..., \vec{\mathbf{k}}([\delta N]) \right\} \\ 0, & \text{otherwise} \end{cases}, \tag{17}$$

where δ is the threshold. Clearly, in normal cases, $\mathbf{m} = [1]^N$ and $\delta = 1.0$. In the experiments, we set $\delta = 0.9$, which means only 90% annotated points with the lowest deviations from prediction will be involved in supervision.

Finally, the overall loss function of MAN is defined by:

$$\mathcal{L} = \mathcal{L}_{IA} + \lambda \mathcal{R}_{lra}. \tag{18}$$

4. Experiments

4.1. Implement Details

Network Structure: We adopt VGG-19 as our CNN backbone network which is pre-trained on ImageNet. We refer to [36] for the structure of transformer encoder and replace the attention module by our proposed LRA. Specifically, as LRA is spatial-aware, the feature map is directly fed into the encoder without position encoding. Our regression decoder consists of an upsampling layer and three convolution layers with activation ReLU function. The kernel sizes of first two layers are 3×3 and that of last is 1×1.

Dataset	ShanghaiTech A		UCF-QNRF		JHU++		NWPU	
Method	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE
MCNN [50] (CVPR 16)	110.2	173.2	277	426	188.9	483.4	232.5	714.6
CP-CNN [30] (ICCV 17)	73.6	106.4	-	-	-	-	-	-
CSRNet [13] (CVPR 18)	68.2	115.0	-	-	85.9	309.2	121.3	387.8
SANet [2] (ECCV 18)	67.0	104.5	-	-	91.1	320.4	190.6	491.4
CA-Net [16] (CVPR 19)	61.3	100.0	107.0	183.0	100.1	314.0	-	-
CG-DRCN-CC [29] (PAMI 20)	60.2	94.0	95.5	164.3	71.0	278.6	-	-
DPN-IPSM [22] (ACMMM 20)	58.1	91.7	84.7	147.2	-	-	-	-
DM-Count [38] (NIPS 20)	59.7	95.7	85.6	148.3	-	-	88.4	388.6
UOT [23] (AAAI 21)	58.1	95.9	83.3	142.3	60.5	252.7	87.8	387.5
S3 [14] (IJCAI 21)	57.0	96.0	80.6	139.8	59.4	244.0	83.5	346.9
P2PNet [31] (ICCV 21)	**52.7**	**85.1**	85.3	154.5	-	-	77.4	362.0
GL [37] (CVPR 21)	61.3	95.4	84.3	147.5	59.9	259.5	79.3	346.1
BL [21] (ICCV 19)	62.8	101.8	88.7	154.8	75.0	299.9	105.4	454.2
MAN (Ours)	56.8	90.3	**77.3**	**131.5**	**53.4**	**209.9**	**76.5**	**323.0**

Table 1. Comparisons with the state of the arts on ShanghaiTech A, UCF-QNRF, JHU-Crowd++, and NWPU. BL [21] serves as our baseline. The best performance is shown in **bold** and the second best is shown in underlined.

Training Details: We first adopt random scaling and horizontal flipping for each training image. Then we randomly crop image patches with a size of 512 × 512. As some images in ShanghaiTech A contain smaller resolution, the crop size for this dataset changes to 256 × 256. We also limit the shorter side of each image within 2048 pixels in all datasets. We use Adam algorithm [12] with a learning rate 10^{-5} to optimize the parameters. We set the number of encoder layers T as 4 and the loss balanced parameter λ as 100.

4.2. Datasets and Evaluation Metrics

Experiments for evaluation are conducted on four largest crowd counting datasets: ShanghaiTech [50], UCF-QNRF [11], JHU-Crowd++ [29] and NWPU-CROWD [40]. These datasets are described as follows:

ShanghaiTech A [50] contains 482 images with 244,167 annotated points. 300 images are divided for training and the remaining 182 images are for testing. Images are randomly chosen from the Internet.

UCF-QNRF [11] includes 1,535 high resolution images collected from the Web, with 1.25 million annotated points. There are 1,201 images in the training set and 334 images in the testing set. UCF-QNRF has a wide range of people count with the minimum and maximum are 49 and 12,865, respectively.

JHU-Crowd++ [29] contains 4,372 images with 1.51 million annotated points. 2,772 images are chosen for training and the rest 1,600 images are for testing. The images are collected from several sources on the Internet using different keywords and specifically chosen for adverse weather conditions.

NWPU-CROWD [40] includes 5,109 images with 2.13 million annotated points. 3,109 images are divided into training set; 500 images are in validation set; and the remaining 1,500 images are in testing set. Compared with other datasets, it has the largest density range from 0 to 20,033 and contains various illumination scenes.

Evaluation Metrics: We evaluate counting methods by two commonly used metrics: Mean Absolute Error (MAE) and Mean Squared Error (MSE). They are defined as follows:

$$MAE = \frac{1}{M}\sum_{i=1}^{M} \left|N_i^{gt} - N_i\right|,$$

$$MSE = \sqrt{\frac{1}{M}\sum_{i=1}^{M}(N_i^{gt} - N_i)^2}.$$

where M is the number of sample images. N_i^{gt} and N_i are ground truth and estimated count of i_{th} image respectively. MAE measures the accuracy of methods more and MSE measures the robustness more. The lower of both represents the better performance [50].

4.3. Comparison with state-of-the-art methods

We evaluate our model on above four datasets and list thirteen recent state-of-the-arts methods for comparison. BL [21] serves as our baseline. The quantitative results of counting accuracy are listed in Table 1. As the result shows, our MAN performs great accuracy on all the four benchmark datasets. MAN improves MAE and MSE values of second best method S3 [14] from 80.6 to 77.3 and from 139.8 to 131.5, respectively. On JHU++, it improves these two values from 59.4 to 53.4 and from 244.0 to 209.9, respectively.

Compared to BL, MAN significantly boosts its counting accuracy on all four datasets. The improvements are

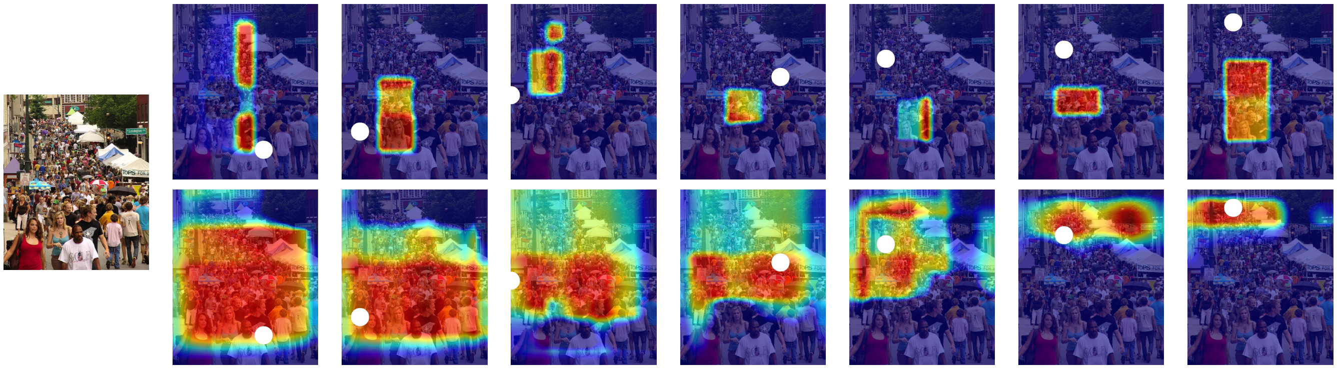


Figure 4. Comparison of regions proposed by our Learnable Region Attention (LRA) without (first row) and with (second row) LAR. Each white circular mark indicates the location of a feature which the mask is corresponding to. For a clearer visualization, we subtracted the mean mask region map corresponding to each head. The attention region with LAR becomes gradually narrower compared to the first row as the scale of focus crowd is smaller.

Transformer	LRA	LAR	IAL	MAE	MSE
				88.7	154.8
✓				85.2	149.5
			✓	84.7	150.9
	✓			84.2	150.8
✓			✓	83.0	146.2
✓	✓			82.9	144.2
✓	✓		✓	81.5	137.9
✓	✓	✓		80.5	140.4
✓	✓	✓	✓	77.3	131.5

Table 2. Ablation study. Transformer is the vanilla form [36]. LRA, LAR and IAL are short for the Learnable Region Attention, Local Attention Regularization and Instance Attention Loss respectively. All experiments are performed on UCF-QNRF. The full model combining all proposed modules performs best.

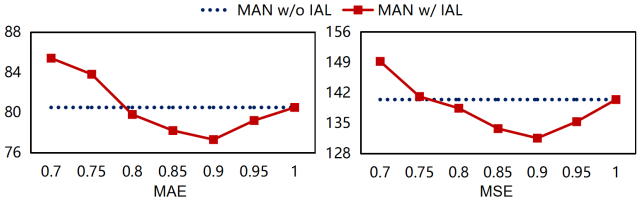


Figure 5. Effect of threshold δ. The dotted line represents our network without proposed Instance Attention Loss. When $0.8 \leq \delta < 1$, the results are better than supervision by all annotations.

9.6% and 11.3% for MAE and MSE on ShanghaiTech A, 12.9% and 15.1% on UCF-QNRF, 28.8% and 30.0% on JHU-Crowd++, and 27.4% and 28.9% on NWPU-CROWD. Visualizations of our MAN are shown in Figure 6.

4.4. Key Issues and Discussion

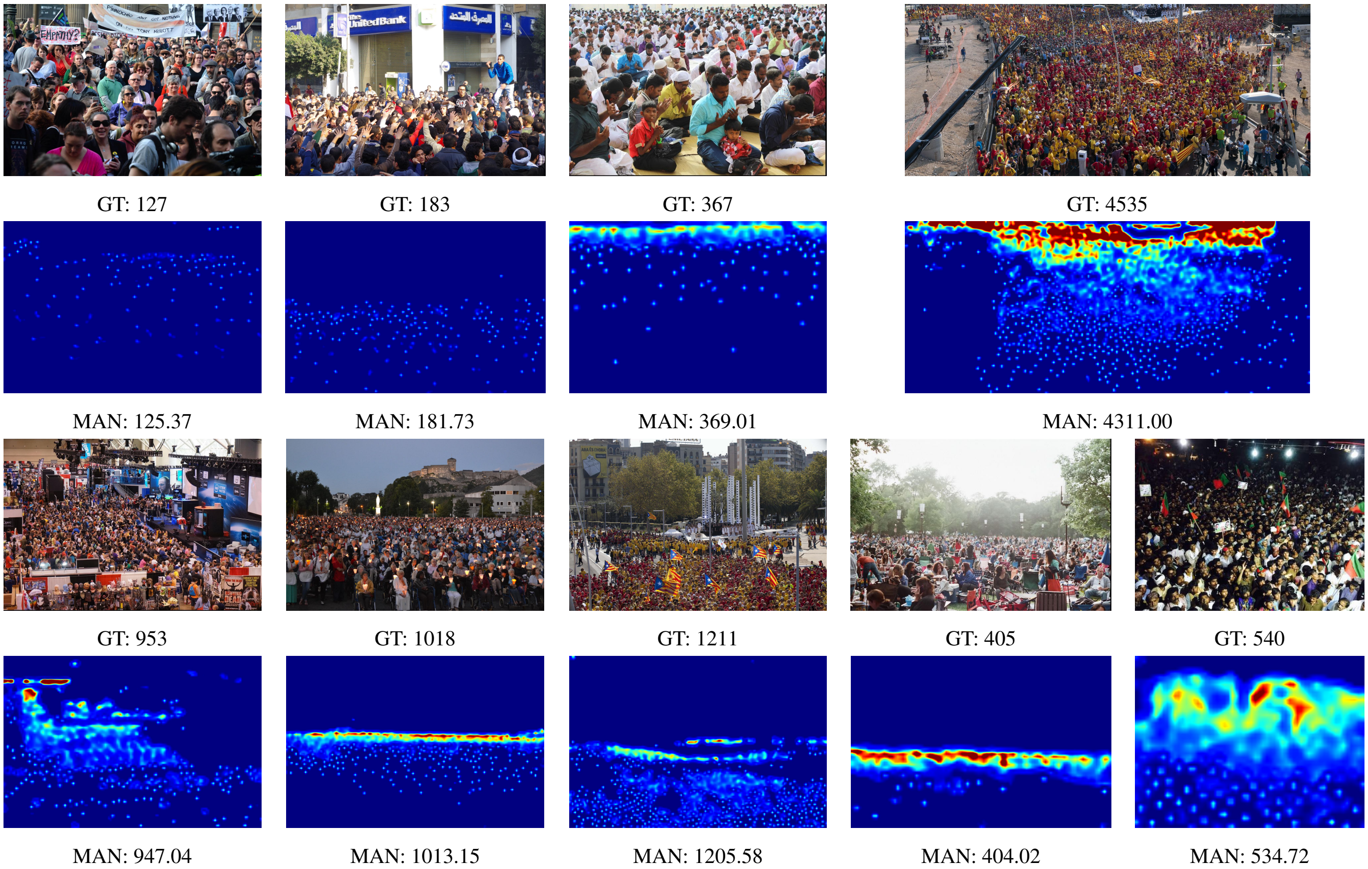
Ablation Studies We perform the ablation study on UCF-QNRF and provide quantitative results in Table 2. We start with the baseline BL [21] and then test the contribution of vanilla transformer encoder [36]. MAE and MSE are reduced by 3.9% and 3.4%, respectively. By adding IAL, the performance from baseline is improved by 4.5% and 2.5%. The combination of transformer and IAL further boosts the counting accuracy. Then, we replace the vanilla attention module by our proposed LRA, the performance is improved by 2.7% and 3.5% without IAL and by 1.8% and 5.7% with IAL. However, it is worth noticing that when we only adopt LRA without global attention, the performance will drop, indicating both global and local information are important. Finally, when the LAR is adopted, the best performance is achieved, boosting the counting accuracy of BL by 12.9% and 15.1% for MAE and MSE, respectively.

Effect of δ: We hold experiments to understand the parameter selection of proposed Instance Attention Loss. We compare the counting accuracy under different thresholds on UCF-QNRF, which result is shown in Figure 5.

We set Multifaceted Attention Network without Instance Attention Loss as the baseline, which also means $\delta = 1$. We observe that the counting accuracy reaches best when $\delta = 0.9$, representing the model cuts off 10% annotations with largest deviations from the prediction.

As δ is selected smaller, the accuracy of MAN declines obviously. It can be explained by the insufficient use of ground truth and that the model is weakly supervised. Then, when we focus on $0.8 \leq \delta < 1$, the results are much better than supervision by all annotations. This may suggest that there are about 20% annotations which will negatively influence the performance of model in counting when adopted in training. And by our Instance Attention Loss, it reduces this negative influence conveniently and effectively.

Visualizations of LRA: Figure 4 presents a comparison of the region mask $\widetilde{R}_i$ in Learnable Region Attention (LRA) without (first row) and with (second row) LAR, where the location of corresponding feature i is marked by a white circle. Supervised by LAR, LRA is able to balance the allo-



GT: 127 GT: 183 GT: 367 GT: 4535

MAN: 125.37 MAN: 181.73 MAN: 369.01 MAN: 4311.00

GT: 953 GT: 1018 GT: 1211 GT: 405 GT: 540

MAN: 947.04 MAN: 1013.15 MAN: 1205.58 MAN: 404.02 MAN: 534.72

Figure 6. Visualizations on UCF-QNRF. The first and third rows are input images while the second and forth rows are the corresponding density maps predicted by our MAN. The warmer color means higher density.

cations of attention resources. As can be seen, the attention region becomes gradually narrower in the second row as the scale of focus crowd is smaller and the number of people needed attention in each region is about the same. Such a dynamic attention scheme is more in accord with human's efficient deployment of attention resources and justifies the usefulness of our LRA and LAR.

Running Cost Evaluation: Table 3 reports a comparison of model size, floating point operations (FLOPs) computed on one 384×384 input image, inference time for 1024×1024 images. It can be easily observed that the model size and inference time of MAN are closed to those of VGG19+Trans and much smaller than those of ViT-B. Moreover, MAN and VGG19-Trans are with a marginal difference in FLOPs. It thus shows that the proposed components are lightweight compared with vanilla transformers.

	ViT-B [10]	Bayesian [21]	VGG19+Trans	**MAN**
Model Size (M)	86.0	21.5	29.9	30.9
GFLOPs	55.4	56.9	58.0	58.2
Inference time	21.3	10.3	10.8	11.3

Table 3. Comparison of the model size (M), FLOPs and Inference time (s / 100 images). Trans stands for the vanilla encoder. The computational cost of MAN only increases a little.

5. Conclusion

This paper is aimed to enhance the ability of transformers in spatial local context encoding for crowd counting. We contribute to the structure of transformers by proposing a Learnable Region Attention module. We also improve the training pipeline by designing Local Attention Regularization to balance the attention allocated for each proposed region and introducing the Instance Attention Loss to reduce the influences of label noise. The proposed Multifaceted Attention Network has achieved state-of-the-art performances on four crowd counting datasets. We consider future directions for applying model to a wider range of vision tasks.

Acknowledgment

This work is funded by National Key Research and Development Project of China under Grant No. 2019YFB1312000 and by National Natural Science Foundation of China under Grant No. 62076195 and U20B2052.

References

[1] Iz Beltagy, Matthew E Peters, and Arman Cohan. Longformer: The long-document transformer. *arXiv preprint*, 2020. 3

[2] Xinkun Cao, Zhipeng Wang, Yanyun Zhao, and Fei Su. Scale aggregation network for accurate and efficient crowd counting. In *ECCV*, 2018. 2, 6

[3] Nicolas Carion, Francisco Massa, Gabriel Synnaeve, Nicolas Usunier, Alexander Kirillov, and Sergey Zagoruyko. End-to-end object detection with transformers. In *ECCV*, 2020. 3

[4] Xin Chen, Bin Yan, Jiawen Zhu, Dong Wang, Xiaoyun Yang, and Huchuan Lu. Transformer tracking. In *CVPR*, 2021. 3

[5] Andrew J Collegio, Joseph C Nah, Paul S Scotti, and Sarah Shomstein. Attention scales according to inferred real-world object size. *Nature human behaviour*, 2019. 2, 4

[6] Pierre Comon. Tensors: a brief introduction. *IEEE Signal Processing Magazine*, 31(3):44–53, 2014. 5

[7] Stéphane d'Ascoli, Hugo Touvron, Matthew Leavitt, Ari Morcos, Giulio Biroli, and Levent Sagun. Convit: Improving vision transformers with soft convolutional inductive biases. *arXiv preprint arXiv:2103.10697*, 2021. 3

[8] Mostafa Dehghani, Stephan Gouws, Oriol Vinyals, Jakob Uszkoreit, and Lukasz Kaiser. Universal transformers. In *ICLR*, 2018. 3

[9] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint*, 2018. 2

[10] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, et al. An image is worth 16x16 words: Transformers for image recognition at scale. In *ICLR*, 2020. 1, 2, 3, 8

[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. In *ECCV*, 2018. 6

[12] Diederik P Kingma and Jimmy Lei Ba. Adam: Amethod for stochastic optimization. 6

[13] Yuhong Li, Xiaofan Zhang, and Deming Chen. Csrnet: Dilated convolutional neural networks for understanding the highly congested scenes. In *CVPR*, 2018. 2, 6

[14] Hui Lin, Xiaopeng Hong, Zhiheng Ma, Xing Wei, Yunfeng Qiu, Yaowei Wang, and Yihong Gong. Direct measure matching for crowd counting. In *IJCAI*, 2021. 2, 6

[15] Jiang Liu, Chenqiang Gao, Deyu Meng, and Alexander G Hauptmann. Decidenet: Counting varying density crowds through attention guided detection and density estimation. In *CVPR*, 2018. 2

[16] Weizhe Liu, Mathieu Salzmann, and Pascal Fua. Context-aware crowd counting. In *CVPR*, 2019. 6

[17] Yan Liu, Lingqiao Liu, Peng Wang, Pingping Zhang, and Yinjie Lei. Semi-supervised crowd counting via self-training on surrogate tasks. In *ECCV*, 2020. 2

[18] Yuting Liu, Miaojing Shi, Qijun Zhao, and Xiaofang Wang. Point in, box out: Beyond counting persons in crowds. In *CVPR*, 2019. 2

[19] Ze Liu, Yutong Lin, Yue Cao, Han Hu, Yixuan Wei, Zheng Zhang, Stephen Lin, and Baining Guo. Swin transformer: Hierarchical vision transformer using shifted windows. *arXiv preprint*, 2021. 1, 3

[20] Zhiheng Ma, Xiaopeng Hong, Xing Wei, Yunfeng Qiu, and Yihong Gong. Towards a universal model for cross-dataset crowd counting. In *ICCV*, 2021. 1

[21] Zhiheng Ma, Xing Wei, Xiaopeng Hong, and Yihong Gong. Bayesian loss for crowd count estimation with point supervision. In *ICCV*, 2019. 1, 2, 5, 6, 7, 8

[22] Zhiheng Ma, Xing Wei, Xiaopeng Hong, and Yihong Gong. Learning scales from points: A scale-aware probabilistic model for crowd counting. In *ACMMM*, 2020. 1, 2, 6

[23] Zhiheng Ma, Xing Wei, Xiaopeng Hong, Hui Lin, Yunfeng Qiu, and Yihong Gong. Learning to count via unbalanced optimal transport. In *AAAI*, 2021. 2, 6

[24] Kristina Naskovska, André L. F. de Almeida, and Martin Haardt. Using double contractions to derive the structure of slice-wise multiplications of tensors with applications to semi-blind mimo ofdm, 2020. 5

[25] Thanh-Tung Nguyen, Xuan-Phi Nguyen, Shafiq Joty, and Xiaoli Li. Differentiable window for dynamic local attention. *arXiv preprint arXiv:2006.13561*, 2020. 3

[26] Alec Radford, Karthik Narasimhan, Tim Salimans, and Ilya Sutskever. Improving language understanding by generative pre-training. 2018. 3

[27] Tao Shen, Tianyi Zhou, Guodong Long, Jing Jiang, Shirui Pan, and Chengqi Zhang. Disan: Directional self-attention network for rnn/cnn-free language understanding. In *AAAI*, 2018. 3

[28] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. *arXiv preprint*, 2014. 3

[29] Vishwanath Sindagi, Rajeev Yasarla, and Vishal MM Patel. Jhu-crowd++: Large-scale crowd counting dataset and a benchmark method. *PAMI*, 2020. 6

[30] Vishwanath A Sindagi and Vishal M Patel. Generating high-quality crowd density maps using contextual pyramid cnns. In *ICCV*, 2017. 2, 6

[31] Qingyu Song, Changan Wang, Zhengkai Jiang, Yabiao Wang, Ying Tai, Chengjie Wang, Jilin Li, Feiyue Huang, and Yang Wu. Rethinking counting and localization in crowds: A purely point-based framework. In *ICCV*, 2021. 6

[32] Robin Strudel, Ricardo Garcia, Ivan Laptev, and Cordelia Schmid. Segmenter: Transformer for semantic segmentation. *arXiv preprint*, 2021. 3

[33] Peize Sun, Yi Jiang, Rufeng Zhang, Enze Xie, Jinkun Cao, Xinting Hu, Tao Kong, Zehuan Yuan, Changhu Wang, and Ping Luo. Transtrack: Multiple-object tracking with transformer. *arXiv preprint*, 2020. 3

[34] Zhiqing Sun, Shengcao Cao, Yiming Yang, and Kris M Kitani. Rethinking transformer-based set prediction for object detection. In *ICCV*, 2021. 3

[35] Ashish Vaswani, Prajit Ramachandran, Aravind Srinivas, Niki Parmar, Blake Hechtman, and Jonathon Shlens. Scaling local self-attention for parameter efficient visual backbones. In *CVPR*, 2021. 3

[36] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *Advances in neural information processing systems*, 2017. 2, 3, 5, 7

[37] Jia Wan, Ziquan Liu, and Antoni B Chan. A generalized loss function for crowd counting and localization. In *CVPR*, 2021. 6

[38] Boyu Wang, Huidong Liu, Dimitris Samaras, and Minh Hoai Nguyen. Distribution matching for crowd counting. *NIPS*, 2020. 2, 6

[39] Ning Wang, Wengang Zhou, Jie Wang, and Houqiang Li. Transformer meets tracker: Exploiting temporal context for robust visual tracking. In *CVPR*, 2021. 3

[40] Qi Wang, Junyu Gao, Wei Lin, and Xuelong Li. Nwpu-crowd: A large-scale benchmark for crowd counting and localization. *PAMI*, 2020. 6

[41] Yabin Wang, Zhiheng Ma, Xing Wei, Shuai Zheng, Yaowei Wang, and Xiaopeng Hong. Eccnas: Efficient crowd counting neural architecture search. *ACM TOMM*, 2022. 1

[42] Yuqing Wang, Zhaoliang Xu, Xinlong Wang, Chunhua Shen, Baoshan Cheng, Hao Shen, and Huaxia Xia. End-to-end video instance segmentation with transformers. In *CVPR*, 2021. 3

[43] Yufei Xu, Qiming Zhang, Jing Zhang, and Dacheng Tao. Vitae: Vision transformer advanced by exploring intrinsic inductive bias. *arXiv preprint*, 2021. 1

[44] Zhaoyi Yan, Yuchen Yuan, Wangmeng Zuo, Xiao Tan, Yezhen Wang, Shilei Wen, and Errui Ding. Perspective-guided convolution networks for crowd counting. In *ICCV*, 2019. 2

[45] Jianwei Yang, Chunyuan Li, Pengchuan Zhang, Xiyang Dai, Bin Xiao, Lu Yuan, and Jianfeng Gao. Focal self-attention for local-global interactions in vision transformers. *arXiv preprint*, 2021. 3

[46] Yifan Yang, Guorong Li, Zhe Wu, Li Su, Qingming Huang, and Nicu Sebe. Reverse perspective network for perspective-aware object counting. In *CVPR*, 2020. 2

[47] Deunsol Yoon, Dongbok Lee, and SangKeun Lee. Dynamic self-attention: Computing attention over words dynamically for sentence embedding. *arXiv preprint*, 2018. 3

[48] Lingke Zeng, Xiangmin Xu, Bolun Cai, Suo Qiu, and Tong Zhang. Multi-scale convolutional neural networks for crowd counting. In *ICIP*, 2017. 1, 2

[49] Pengchuan Zhang, Xiyang Dai, Jianwei Yang, Bin Xiao, Lu Yuan, Lei Zhang, and Jianfeng Gao. Multi-scale vision longformer: A new vision transformer for high-resolution image encoding. *arXiv preprint*, 2021. 1, 3

[50] Yingying Zhang, Desen Zhou, Siqin Chen, Shenghua Gao, and Yi Ma. Single-image crowd counting via multi-column convolutional neural network. In *CVPR*, 2016. 1, 2, 6

[51] Muming Zhao, Jian Zhang, Chongyang Zhang, and Wenjun Zhang. Leveraging heterogeneous auxiliary tasks to assist crowd counting. In *CVPR*, 2019. 2

[52] Minghang Zheng, Peng Gao, Xiaogang Wang, Hongsheng Li, and Hao Dong. End-to-end object detection with adaptive clustering transformer. *arXiv preprint*, 2020. 3

[53] Sixiao Zheng, Jiachen Lu, Hengshuang Zhao, Xiatian Zhu, Zekun Luo, Yabiao Wang, Yanwei Fu, Jianfeng Feng, Tao Xiang, Philip HS Torr, et al. Rethinking semantic segmentation from a sequence-to-sequence perspective with transformers. In *CVPR*, 2021. 3

[54] Xizhou Zhu, Weijie Su, Lewei Lu, Bin Li, Xiaogang Wang, and Jifeng Dai. Deformable detr: Deformable transformers for end-to-end object detection. In *ICLR*, 2020. 3