Attention-Guided Collaborative Counting

Hong Mo, Wenqi Ren, *Member, IEEE*, Xiong Zhang, Feihu Yan, Zhong Zhou, *Member, IEEE*,
Xiaochun Cao, *Senior Member, IEEE*, and Wei Wu

***Abstract*— Existing crowd counting designs usually exploit multi-branch structures to address the scale diversity problem. However, branches in these structures work in a competitive rather than collaborative way. In this paper, we focus on promoting collaboration between branches. Specifically, we propose an attention-guided collaborative counting module (AGCCM) comprising an attention-guided module (AGM) and a collaborative counting module (CCM). The CCM promotes collaboration among branches by recombining each branch's output into an independent count and joint counts with other branches. The AGM capturing the global attention map through a transformer structure with a pair of foreground-background related loss functions can distinguish the advantages of different branches. The loss functions do not require additional labels and crowd division. In addition, we design two kinds of bidirectional transformers (Bi-Transformers) to decouple the global attention to row attention and column attention. The proposed Bi-Transformers are able to reduce the computational complexity and handle images in any resolution without cropping the image into small patches. Extensive experiments on several public datasets demonstrate that the proposed algorithm performs favorably against the state-of-the-art crowd counting methods.**

***Index Terms*— Crowd counting, attention-guided collaborative counting model, bi-directional transformer.**

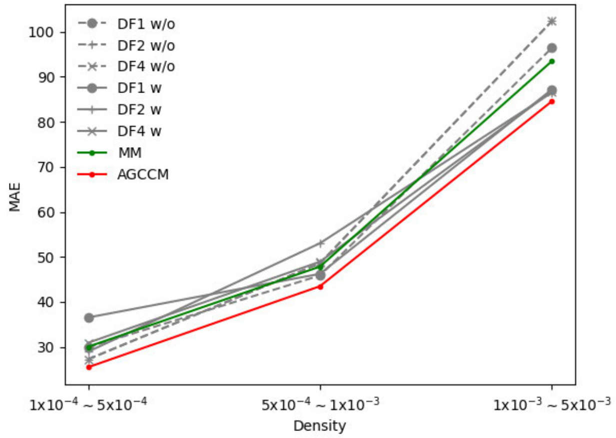


Fig. 1. The mean absolute error (MAE) of CSRNet [8] with different convolutions and multi-branch module with and without AGCCM on the SHA dataset [1]. SHA is divided into three subsets according to density. The density is the ratio of the people number in the image to the image resolution. DF1, DF2, and DF4, in turn, represent the dilated convolution with dilation rates of 1, 2, and 4. "w/o" and "w" means the dilated convolution without or with the deformable convolution.

I. Introduction

Crowd counting, a task aiming at computing the total number of people in the image, has recently become a focus in computer vision. However, due to the random crowd distribution and perspective distortion, human heads shown in the picture have different scales, making it more challenging to count people in the image.

It is well known that convolutions in different layers or with various kernel sizes are critical for capturing different scales. Many methods [1], [2], [3], [4], [5], [6], [7] capitalized on this common-sense have been proposed to deal with the scale variation problem. Zhang *et al.* [1] and Sam *et al.* [2] have contemporary works that first introduce a multi-branch structure with different receptive fields (large, medium, and small) to provide robustness counting for the large variation in crowd scales. Nevertheless, as demonstrated by [8] and [9], branches in the multi-branch structure work in a competitive rather than collaborative way, inhibiting each branch from achieving its best performances. Another line of works [3], [4], [5], explicitly divide the crowd into parts to narrow the large gap of head size variation. For example, Xu *et al.* [3] partition the crowd into groups in near- and far- view according to a depth map estimated by a multi-scale deep neural network. Then they utilize a detection-based approach to count people in the near-view group and estimate the density map in the far-view group. However, experiment results in [5] show that this kind of crowd division method is sensitive to the division results. Moreover, they need additional labels, for example, head size, which are hard to obtain.

This paper dedicates to promoting collaboration among branches in a multi-branch style network structures. We illustrate the MAE of CSRNet with different convolutions on the SHA dataset partitioned into different crowd densities in Figure 1. As the gray lines are shown in Figure 1, the model with different convolutions has advantages in different crowd

Manuscript received 3 January 2022; revised 20 July 2022 and 3 September 2022; accepted 7 September 2022. Date of publication 30 September 2022; date of current version 7 October 2022. This work was supported in part by the Natural Science Foundation of China under Grant 61872024, in part by the National Natural Science Foundation of China Grant 62172409, and in part by the Beijing Nova Program under Grant Z201100006820074. The associate editor coordinating the review of this manuscript and approving it for publication was Prof. Clinton Fookes. *(Corresponding author: Zhong Zhou.)*

Hong Mo, Zhong Zhou, and Wei Wu are with the State Key Laboratory of Virtual Reality Technology and Systems, Beihang University, Beijing 100191, China (e-mail: mandymo@buaa.edu.cn; zz@buaa.edu.cn; wuwei@buaa.edu.cn).

Wenqi Ren and Xiaochun Cao are with the School of Cyber Science and Technology, Sun Yat-sen University, Shenzhen Campus, Shenzhen 518107, China (e-mail: renwq3@mail.sysu.edu.cn; caoxiaochun@mail.sysu.edu.cn).

Xiong Zhang is with the Neolix Autonomous Vehicle, Beijing 100016, China (e-mail: zhangxiong@neolix.cn).

Feihu Yan is with the School of Electrical and Information Engineering, Beijing University of Civil Engineering and Architecture, Beijing 100044, China (e-mail: yanfeihu@bucea.edu.cn).

Digital Object Identifier 10.1109/TIP.2022.3207584

densities. However, the gap between each branch's MAE on different crowd densities is not large enough to make the multi-branch module distinguish expert area for each branch itself, which may lead to the competitive mechanism, as the green line shown in Figure 1. As a remedy, in the work of [10], Liu *et al.* try to promote collaborative representation of different information to facilitate multimodal crowd counting. Different from [10], we reach the collaborative counting from two aspects. The first one is to allocate an expert area for each branch, and the second strategy decouples the count of each branch into independent count and cooperative count with other branches. The red line in Figure 1 demonstrates that the multi-branch module combined with the proposed AGCCM achieves better performance in different crowd densities than each single branch.

Our main contributions are as follows:

- We creatively propose a collaborative counting mechanism (CCM) by recombining each branch's count into independent and joint counts with other branches. The CCM, with negligible computation, inspirits each branch to focus on its expert spatial area and share with other branches the count of its sub-optimal domain.
- We design an attention-guided model (AGM) to assist the independent and joint count area from a global view in an implicit way. Two novel bidirectional (row direction and column direction) transformers (Bi-Transformers) are proposed to achieve the global view. The Bi-Transformer can save computation and deal with input of various resolutions without splitting the image into small patches. Besides, we design a pair of loss functions related to foreground and background to implicitly guide the model to distinguish each branch's count, which does not necessitate additional annotations.

The proposed approach outperforms contemporary methods and demonstrates new state-of-the-art performances on several widely used benchmarks.

II. Related Works

This section briefly reviews the most related works, including multi-branch models and attention mechanism.

A. Multi-Branch Models

Due to the perspective deformation, the scale variation exists in many computer vision tasks, such as object detection [11], [12], semantic image segmentation [13], [14], and crowd counting [8], [15]. The multi-branch model design proves an effective strategy to solve the scale variation problem. We regard the multi-branch model as an ensemble learner, and each branch can be considered an individual learner. According to the combination object, we divide the multi-branch models into crowd ensembling models [1], [2], [3], [4], [5], feature ensembling models [15], [16], [17], [18], [19], [20], [21], and task ensembling models [9], [22], [23], [24].

The crowd ensembling models work in a 'divide and conquer' manner [25] to bridge the gap of various crowd density distribution. Zhang *et al.* [1] and Sam *et al.* [2] crop the image into nine patches without overlap to count them with different branches independently. References [3], [4], [5] divide the crowd into parts according to depth or head size. Then the divided crowd can be counted by branches which are excel on. However, these methods heavily rely on crowd division, while dividing the crowd based on a rigid number violates the arbitrariness of crowd distribution.

Feature ensembling models refer to fusing features between branches [15], [16], [17] or layers [18], [19], [20] to enhance the image representation ability for multi-size objects. Sindagi and Patel [16] and Liu *et al.* [15] concatenate the context information from different branches to extract informative feature maps. Sindagi and Patel [16] classify the crowd density into five classes to offer the global and local context information. Liu *et al.* [15] obtain contextual-aware features through spatial pyramid pooling the feature maps, then fuse them with scale-aware features gained by a local scale encoded perspective map. In [17], Cao *et al.* enhance the feature maps' representation ability and scale diversity by stacking a inception-like [26] scale aggregation module and generate a high-quality density map through transposed convolution and a local pattern consistency loss. There exist some other works [18], [19], [20] that integrate features from different layers. However, those works often require complex networks to extract multi-scale feature maps of the input.

The task ensembling models utilize the auxiliary relation between tasks to improve the counting accuracy. Idress *et al.* [22] observe that counting, density map estimation, and people localization in a dense crowd image are inherently related, making the loss function for optimizing a deep CNN decomposable. Sindagi and Patel [23] incorporate a high-level prior population into density estimation through coarsely estimating the count. Shen *et al.* [9] utilize a GAN [27] to identity the sum-up count from local patches and count of the whole image. However, multi-task assist approaches usually need extra annotations, such as head bounding boxes, depth map and crowd density level.

In addition, all these above multi-branch structures are limited by the number of branches and lacks collaboration between branches as pointed by Li *et al.* [8] and Shen *et al.* [9]. As a complementary, this paper provide a collaboration counting method to fully play the strengths of each branch to achieve better performance.

B. Attention Mechanism

The basic idea of attention mechanism in computer vision is to make the system focus on the critical area rather than treat all pixels equally [28], [29], [30], [31]. Self-attention is a variant of the attention mechanism, which is the key structure of the transformer [32]. The transformer proposed by Vaswani *et al.* [32] for machine translation, is famous for its significant parallel capabilities and long term dependency. The self-attention in the transformer can capture the relations of each pair of words/pixels to form a global attention distribution. It has since become the state-of-the-art method in many natural language processing tasks [33], [34], [35], [36]. For example, Fan *et al.* [35] noticed the multi-branch module's

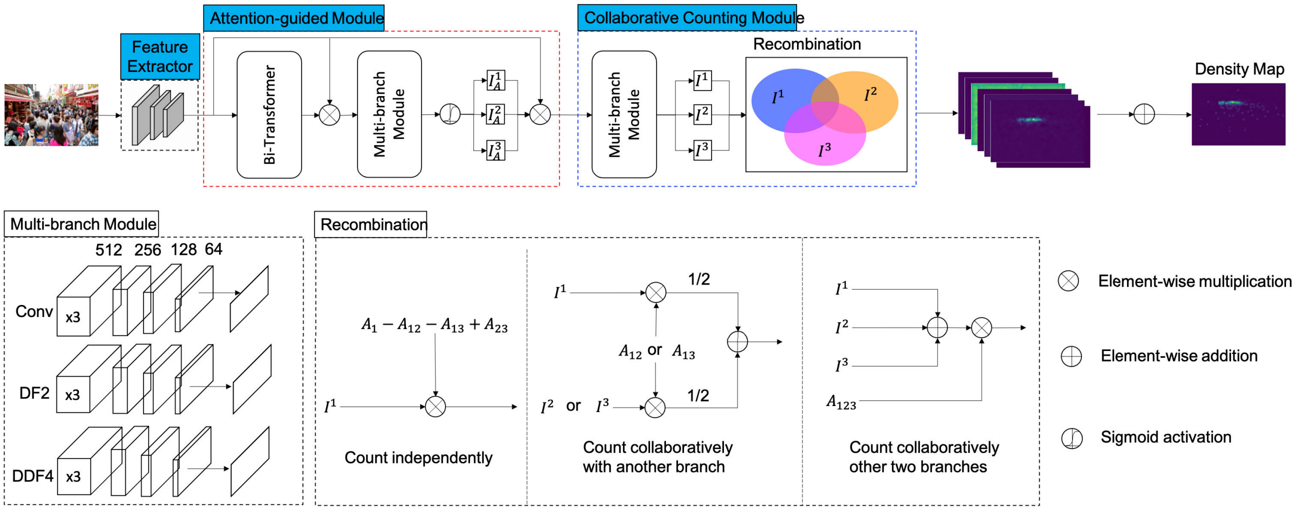


Fig. 2. The illustration of the framework. The framework contains three parts: feature extractor, attention-guided module (AGM), and collaborative counting module (CCM). The feature extractor is used to learn images feature representation. Then the global attention distributions (I_A^1, I_A^2, I_A^3) are generated by AGM and are used to divide expert area for each branch. Finally, feature maps multiply with three global attention distributions are respectively fed into each branch of the multi-branch module in CCM. The CCM recombines three outputs of the multi-branch module and generates the final predicted density map. The dashed box under CCM flag shows an example of recombination. A_1, $A_{12}(A_{13})$, and A_{123} represent the independent counting area of one branch, and collaboration counting area of two branches, and the cooperation area of three branches, respectively. ×3 indicates repeating the block three times. The number on top of each block is its channel. Conv, DF2, and DDF4 are means conventional convolution, dilated convolution with a dilation factor of 2, and dilated convolution with a dilation factor of 4 based on deformable convolution, respectively.

good performance in computer vision tasks. They introduced a multi-branch attentive transformer structure into kinds of neural language processing tasks and achieved significant performance improvement.

Computer vision tasks can be considered as a long sequence for their millions of pixels, bringing in huge computation and memory complexity of the self-attention module. References [37], [38], [39], [40] light weight attention to release the press from computation and memory complexity. Dosovitskiy *et al.* [39] overcame the high computation problem by splitting the image into a sequence of tokens with a fixed length. While the high computational cost and high complexity of self-attention are still unignorable when dealing with high-resolution images. In the work of [40], Liu *et al.* designed the Swin-Transformer, which realized linear computational complexity concerning input image size by utilizing a shifted window. However, the attention in Swin-Transformer is locally only related pixels in the same window. Besides, inspired by resnet [41], He *et al.* [42] introduced a dense connected transformer to deal with the vanishing gradient problem.

Moreover, the transformer has been used in many kinds of computer vision tasks and achieved significant improvement, such as image generation [37], image classification [39], [43], hyperspectral image classification [42], semantic segmentation [44], etc. Tian *et al.* [45] and Sajid *et al.* [46] also utilized transformer structures to capture global attention for more accurate crowd counting. However, they need to crop the image into small patches, which may cause apparent seams. Considering that the difference of reality head size is negligible, head size in the image is high related to its position. Therefore, we decompose the self-attention into a row- and column- attention to catch the global relation distributions. This kind of decomposition obviously reduces the computation complexity of the transformer.

III. Methods

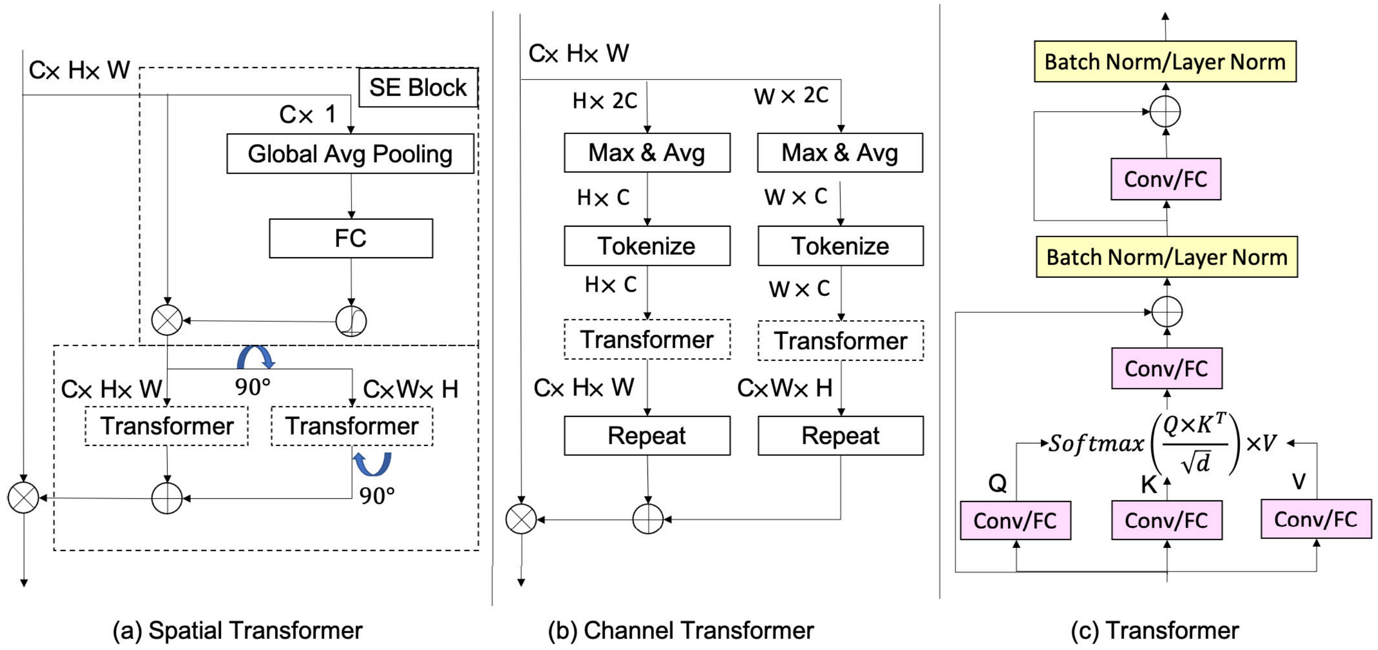
The architecture of the framework is shown in Figure 2. It includes a feature extractor, an attention-guided module (AGM), and a collaborative counting module (CCM). The AGM generates attention maps to allocate an expert area for each branch from a global view. The CCM recombines multi-branch module outputs to promote collaboration among branches.

A. Feature Extractor

The input image is first fed into a feature extractor to obtain image representation. Considering that VGG16 is widely used in the network structure of the crowd counting task, we also adopt the top 13 layers of VGG16 as the image feature extractor to reduce unnecessary variables when compared with other methods. Additionally, only the first three pooling layers are saved to ensure the output is in a relatively high resolution. Taking H and W as the input image's height and width, the feature maps are in a resolution of $\frac{1}{8}$H × $\frac{1}{8}$W.

B. Attention-Guided Module

As mentioned before, the AGM can provide attention distributions from a global view. We realize the global view by introducing a transformer variant – bidirectional transformer (Bi-Transformer). The AGM includes a Bi-Transformer followed by a multi-branch module. Besides, we utilize a pair of loss functions related to background and foreground to force the model to implicitly distinguish each branch's expert area.



(a) Spatial Transformer

(b) Channel Transformer

(c) Transformer

Fig. 3. The framework of the two kinds of Bi-Transformers: (a) Spatial-Transformer and (b) channel-transformer. Spatial-transformer keeps the spatial structure and converts self-attention into asynchronously channel attention and spatial attention. Channel-transformer reforms the self-attention into two parallel transformers by sacrificing the individual differences in intra-column and intra-row of the input. $\oplus$ and $\odot$ mean element-wise operation. Max & Avg represents the concatenated maximum and mean of vectors in the same column or row.

1) Bi-Transformers: Multi-head attention is a core structure of transformer. Multi-head attention uses multiple queries $Q = [q_1, \cdots, q_N]$ to compute the selection of multiple inputs in parallel. Each attention focuses on a different part of the input. When queries comes from inputs, it is called self-attention. To an input image/features $I \in \mathbb{R}^{H\times W\times C}$, the number of query is $N = H\times W$. C is the channel number of the input. Three weight matrices are learned to achieve the linear representation of query vector (Q), the key vector (K), and the value vector (V) as follows,

$$\begin{aligned} Q &= I \times W_Q, \\ K &= I \times W_K, \\ V &= I \times W_V. \end{aligned} \tag{1}$$

The $W_Q, W_K, W_V \in \mathbf{R}^{C\times d}$. Then, the output of the attention can be calculated as,

$$\text{Attention}(Q, K, V) = \text{Softmax}(\frac{QK^T}{\sqrt{d}})V. \tag{2}$$

Therefore, the complexity and parameters of the attention module are $O(N^2 \times d)$ and $O(4d^2)$, respectively. The attention weights are divided by $\sqrt{d}$ to stabilize gradients during training. At present, the mainstream visual task processing images are mostly megapixels or above. This makes the computation of the attention mechanism not negligible, which should be first considered when introducing transformer into visual tasks. For example, the swin-transformer [40] cropped the input into small patches. To reduce the computational complexity, swin-transformer only computes self-attention within local windows. The windows are composed of a predefined number of patches. However, self-attention within local windows can not provide long range dependencies.

Supposing the similarity between vectors calculated through dot product, then the similarity between adjacent pixels can be blurred into the similarity between rows and columns. Based on this kind of similarity, we decompose the self-attention into the sum of the row- and column- attention and design two types of bidirectional transformers called Bi-Transformer. The row- and column- attention can be calculated through matrix multiply. This kind of variation keeping spatial relation to a degree is named spatial transformer. On the other hand, we further simplify the bidirectional transformer structure. Assuming that the population distribution of each row (column) follows a Gaussian distribution, we can represent it by the mean and maximum value of each row or column. Then the relation between each row (column) is more related to the channel. We called this simplified version is channel transformer. The detail the two transformer variants are displayed in Figure 3.

Spatial-transformer contains a SE block [29] and two parallel transformers, as displayed in Figure 3(a). We use the SE block to generate channel-wise feature maps. The channel-wise feature maps and their transpose are fed into the two parallel transformers separately to attain row attention and column attention. Symmetrical transposition operation is performed after one of the transformers, so that the output of two parallel transformers can be added. In the channel-transformer, we concatenate the mean and maximum value of each row/column to represent each row/column. Then, each row/column goes through a fully connected layer to linearly transform into the same dimension as the original input. The channel-transformer also contains two parallel transformers.

TABLE I

DETAILS OF TWO KINDS OF TRANSFORMERS IN SPATIAL-TRANSFORMER AND CHANNEL-TRANSFORMER

Spatial-Transformer	Channel-Transformer
(1×1×64)	$d = 256$
(3×3×128)	$d = 512$
(3×3×256)	
(3×3×512)	
(3×3×1024)	$d = 1024$
(3×3×512)	$d = 512$

TABLE II

COMPARISON OF COMPUTATION COMPLEXITY AND PARAMETERS OF ATTENTION MODULE IN DIFFERENT TRANSFORMERS

Module	Complexity	#Parameters
Conventional	$O((\mathrm{H}^2 \times \mathrm{W}^2) \times d)$	$4d^2$
Channel	$O((\mathrm{H}^2 + \mathrm{W}^2) \times d)$	$8d^2$
Spatial	$O((\mathrm{H} + \mathrm{W})(\mathrm{H} \times \mathrm{W}) \times d)$	$4k^2 \times d$

Figure 3(c) shows the two kinds of transformer in the proposed channel-transformer and spatial-transformer. The transformer structure of the channel-transformer is the same as the standard transformer, which is realized by full connection and layer normalization. In the spatial-transformer, a convolution operation is used to preserve the spatial continuity of pixels in the image. The dimensions of attention (Q, K, V) in row-direction and column-direction in the spatial-transformer is W^2 and H^2, respectively, and in the channel-transformer is equal to the channel number C.

Detailed parameters in format of (kernel size × kernel size × channel) are shown in Table I in the order from input to output. Row and column attention in AGM based on spatial-transformer is achieved by transpose operation and matrix multiplication. In channel-transformer, the representation of each vector is only related to the number of channels in the input feature maps, and operation in spatial-transformer has no relation to resolution. Thus, our Bi-Transformers can process images of any resolution without cropping the input into small pathces.

We present the computation complexity and number of parameters of the attention mechanism in the three kinds of transformers in Table II. Let dot product function $f(a,b) =$ <a, b> to calculate the similarity of a and b. Suppose $a, b \in \mathbb{R}^d$, then the dot product of a and b consists of d multiplications and $d - 1$ addition. For conventional transformer, it contains $\mathrm{H}^2 \times \mathrm{W}^2$ pixel pairs to be calculated. Thus, the computation complexity of self-attention in conventional transformer is $O((\mathrm{H}^2 \times \mathrm{W}^2) \times d)$. The attention of spatial and channel transformer can be respectively calculated according to

$$f(x^t_{i,j}, x^t_{m,n}) = \underbrace{f(X^t_{i,:}, X^t_{m,:})}_{\text{Row attention}} + \underbrace{f(X^t_{:,j}, X^t_{:,n})}_{\text{Column attention}}, \tag{3}$$

and

$$f(x_{i,j}, x_{m,n}) = \underbrace{f(\widehat{X_{i,:}}, \widehat{X_{m,:}})}_{\text{Row attention}} + \underbrace{f(\widehat{X_{:,j}}, \widehat{X_{:,n}})}_{\text{Column attention}}. \tag{4}$$

The $f(X^t_{i,:}) \in \mathbb{R}^{\mathrm{W}}$ and the $f(X^t_{:,m}) \in \mathbb{R}^{\mathrm{H}}$ represent the i-th row and the m-th column vector at the t-th dimension, where $t \in [1, 2, \ldots, d]$. Spatial transformer includes H×W $f(x_{i,j}, x_{m,n})$ calculations, and computation complexity of each $f(x_{i,j}, x_{m,n})$ is $O((\mathrm{H} + \mathrm{W}) \times d)$. In Eq. 4, $\widehat{X_{i,:}} \in \mathbb{R}^d$ indicates the embedding of the i-th row information. Channel transformer computes each pair of rows and each pair of columns attention. The computation complexity of a conventional transformer is a quartic function of image size. The spatial transformer and channel transformer reduce complexity to cubic and quadratic functions of image size, respectively. Moreover, we note that the number of parameters in the channel transformer is double that of the standard transformer. This is because the channel transformer essentially contains two parallel conventional transformers. The number of arguments in the spatial-transformer is reduced to a linear function of the channel number $4k^2 \times d$, where k is the kernel size.

2) Loss Function of AGM: A pair of loss functions about foreground and background are proposed to supervise the attention-guided module (AGM) to learn each branch's independent and collaboration areas. The loss consists of two parts, the first one makes the three attention maps consistent with the foreground, and the second constrains the intersection region of the three attention maps to 0.

We first estimate the foreground density map $\mathrm{P_F}$ and background density map $\mathrm{P_B}$ by,

$$\mathrm{P_F} = I^1_A + I^2_A + I^3_A, \tag{5}$$

and

$$\mathrm{P_B} = (1 - I^1_A) \times (1 - I^2_A) \times (1 - I^3_A). \tag{6}$$

Due to the large difference in the ratio of foreground and background, we design two image adaptive focal loss [47],

$$\begin{aligned}\mathrm{FL}(\mathrm{P_F}, \mathrm{G_F}) = &-\alpha_1 \times \mathrm{P_F}^{\gamma} \times (1 - \mathrm{G_F}) \times \log(1 - \mathrm{P_F}) \\ &- (1 - \alpha_1) \times (1 - \mathrm{P_F})^{\gamma} \times \mathrm{G_F} \times \log \mathrm{P_F},\end{aligned} \tag{7}$$

and

$$\begin{aligned}\mathrm{FL}(\mathrm{P_B}, \mathrm{G_B}) = &-\alpha_2 \times \mathrm{P_B}^{\gamma} \times (1 - \mathrm{G_B}) \times \log(1 - \mathrm{P_B}) \\ &- (1 - \alpha_2) \times (1 - \mathrm{P_B})^{\gamma} \times \mathrm{G_B} \times \log \mathrm{P_B},\end{aligned} \tag{8}$$

where $\mathrm{G_F}$ and $\mathrm{G_B}$ represent the ground truth of foreground density map and background density map, respectively. It is worth noting that in the actual experimental operation, to avoid the overflow of the foreground loss function, the exact expression of $\mathrm{P_F}$ is the sigmoid activation of the sum of each branch's output of the multi-branch module. We fixed γ as 2 and set the weight α_1/α_2 of the niche category to 0.75. For example, when the background area is larger than the foreground area, we set α_2 to 0.25 to decrease the contribution of background loss; otherwise, α_2 is set to 0.75 to increase the background loss. The sum of the α_1 and α_2 is one ($\alpha_1 + \alpha_2 = 1$) because the foreground and background are complementary. Finally, the total loss of the AGM is,

$$\ell_{\mathrm{mask}} = \mathrm{FL}(\mathrm{P_F}, \mathrm{G_F}) + \mathrm{FL}(\mathrm{P_B}, \mathrm{G_B}), \tag{9}$$

where λ is an external parameter.

C. Collaborative Counting Module

The collaborative counting module contains a multi-branch module and recombination. The structure of multi-branch module in both AGM and CCM is absolutely the same.

1) Multi-Branch Module: We compose the multi-branch model relying on the performance of each solo branch at different crowd densities. As a result, we select three kinds of convolution that can perform optimally when working independently and achieve complementary performance at different densities. The selected convolutions for three branches are conventional convolution (Conv), dilated convolution with a dilation factor of 2 (DF2), dilated convolution with a dilation factor of 4 based on deformable convolution (DDF4), respectively. The first and second branches have superior performance at small crowd scale belongs to $(1 \times 10^{-4}, 5 \times 10^{-4})$ and $(5 \times 10^{-4}, 1 \times 10^{-3})$, respectively. The third branch works best at the dense crowd with a high density ranges from 1×10^{-3} to 5×10^{-3}. Kernel size in each convolution is fixed at 3×3 and channels are displayed in Figure 2 above each block in the multi-branch module. All these convolutions are followed by a ReLU activation function and Batch Normalization. At last, we use convolution with kernel size 1×1 and channel 1 to generate density maps. The multi-branch module inside the AGM takes the feature maps with global attention distribution as input and outputs three attention maps (I_A^1, I_A^2, I_A^3) after a sigmoid activation. In CCM, the multi-branch module takes three feature maps with three different attention (I_A^1, I_A^2, I_A^3) as input and outputs three density maps (I^1, I^2, I^3).

2) Recombination: Since the division of each branch's dominant area is not obvious, we propose a recombination step to aggregate density information of the three outputs (I^1, I^2, I^3) from the second multi-branch module. The relationship of the three density maps can be described by a Wenn diagram in the blue dotted box in Figure 2. Each branch's count can be classified into three forms: independent count, cooperation count with another branch, and cooperation count with all these three branches. We treat the counting of all branches involved in the collaborative area evenly.

$$\begin{aligned} \mathrm{O}_1 = & \underbrace{(A_1 - A_{12} - A_{13} + A_{123}) \times I^1}_{\text{Count Independently}} \\ & + \underbrace{(A_{12} - A_{123}) \times \frac{I^1}{2}}_{\text{Count Collaboratively with DF2}} + \underbrace{(A_{13} - A_{123}) \times \frac{I^1}{2}}_{\text{Count Collaboratively with DDF4}} \\ & + \underbrace{A_{123} \times \frac{I^1}{3}}_{\text{Count Collaboratively with DF2 and DDF4}} \end{aligned} \tag{10}$$

Taking the first branch (Conv) as an example, its count can be calculated according to Equation 10. We use A_1, A_{12}, A_{13}, A_{123} to denote counting areas. Such as, $A_{12} = \sigma(I^1) \times \sigma(I^2)$ represents the cooperation area of branch Conv and DF2. The final prediction can be represented as $(\mathrm{O}_1 + \mathrm{O}_2 + \mathrm{O}_3)$.

IV. Experimental Results & Dataset

In this section, we first give the description of the implementation details, and then present the comparison between state-of-the-arts and our model on three datasets, namely ShanghaiTech [1], ShanghaiTechRGBD [48], and JHU-Crowd++ [49]. Extensive ablation study is then conducted to clarify the contribution of each component in our model. The code will be avilable tho the public.

A. Evaluation Metric and Labels

We use Mean Absolute Error (MAE)

$$\mathrm{MAE} = \frac{1}{\mathrm{T}} \sum^{\mathrm{T}} \left| \sum \mathrm{P_D} - \sum \mathrm{G_D} \right|, \tag{11}$$

and Root Mean Square Error (RMSE)

$$\mathrm{RMSE} = \frac{1}{\mathrm{T}} \sum^{\mathrm{T}} \left| \sum \mathrm{P_D} - \sum \mathrm{G_D} \right|^2, \tag{12}$$

as the evaluation metrics like in [2], [8],and [50]. We use $\mathrm{P_D}$ and $\mathrm{G_D}$ to represent the predicted and ground truth density maps, respectively. Symbol T denotes the number of the examples.

This works refers to three labels, the ground truth of density maps $\mathrm{G_D}$, foreground $\mathrm{G_F}$, and background $\mathrm{G_B}$. The $\mathrm{G_D}$ is a blurring head distribution achieved through a bivariate gaussian function, as displayed in the following equation.

$$\mathrm{G_D} = \sum_{i=1} \delta(x - x_i, y - y_i) \times G_{(\sigma_x=5,\sigma_y=5)} \tag{13}$$

$G_{(\sigma_x=5,\sigma_y=5)}$ is a binary Gaussian centered at the annotated head position (x_i, y_i) with a variance of (5, 5). The ground truth of foreground,

$$\mathrm{G_F} : \mathbb{I}_{\mathrm{G_D}}(x, y) = \begin{cases} 1 & \text{if} \quad \mathrm{G_D}(x, y) > 0 \\ 0 & \text{otherwise}, \end{cases} \tag{14}$$

and background,

$$\mathrm{G_B} : \mathbb{I}_{\mathrm{G_D}}(x, y) = \begin{cases} 1 & \text{if} \quad \mathrm{G_D}(x, y) = 0 \\ 0 & \text{otherwise}, \end{cases} \tag{15}$$

are two indication functions on $\mathrm{G_D}$.

B. Datasets

1) ShanghaiTech [1]: SHA and SHB are two parts of the ShanghaiTech dataset. Images in SHA are in arbitrary resolution, which is randomly crawled from the Internet. It contains 482 pictures and a total of 241,677 points near head center annotations. SHB is collected on the busy streets of Shanghai with a resolution fixed at 1024×768. SHB includes 716 images with 88,488 annotations. Relative to SHB, most images in SHA have a large number of people.

2) JHU-Crowd++ [49]: It is one of the largest crowd counting datasets with rich annotations at both image-level and head-level. JHU-Crowd++ contains 4,372 images in average resolution of 910×1430 and 1,515,005 annotations. It provides annotations, including heads locations and corresponding occlusion level, blur level and size level, bounding boxes, scene labels (such as marathon, mall, railway station, stadium, etc.), and the weather labels (rain, snow, and fog).

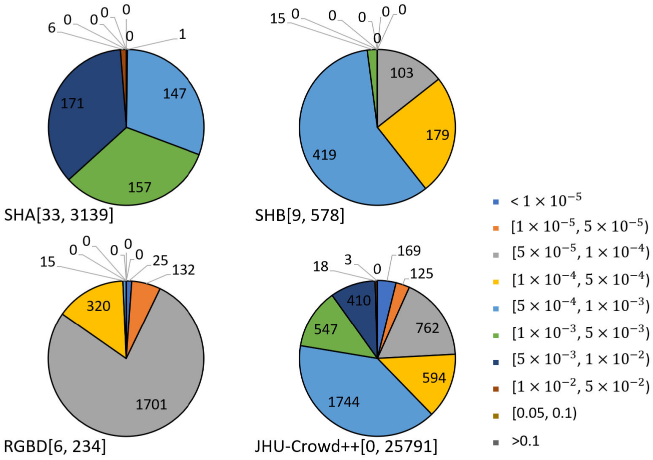


Fig. 4. Crowd scale statistics on three public datasets. The number next to the dataset name is its average density, the ratio of average count to average resolution.

3) RGBD [48]: Images' resolution of RGBD is fixed at 1080 × 1920. It contains 2,193 images and 144,512 head annotation. Besides, it provide depth map.

In addition, we show the statistical distribution of crowd density in each data set in Figure 4. The crowd density ranges from 0 to 1 and is divided into ten levels as $[0, 1 \times 10^{-5})$, $[1 \times 10^{-5}, 5 \times 10^{-5})$, $\cdots$, $[0.1, 1)$. Figure 4 shows the average density of part A is the largest, and its image distribution under each density is relatively uniform. RGBD has the smallest average density. The density involved in JHU-Crowd++ is the most extensive. Moreover, the crowd density of images in SHB and JHU-Crowd++ favors being between 5×10^{-4} and 1×10^{-3}.

C. Implementation Details

We use a squared L2 norm loss,

$$\ell_{\text{density}} = \frac{1}{\mathrm{N}} \sum (\mathrm{P_D} - \mathrm{G_D})^2, \tag{16}$$

and ℓ_{mask}, displayed in Equation 9, to constraint the density map and the attention map, respectively. The total loss is

$$\ell = \ell_{\text{density}} + \lambda \times \ell_{\text{mask}}. \tag{17}$$

The pre-assigned parameter λ is applied to adjust the proportion of the attention map in the training process. We optimize the loss by Adam [51] with a learning rate fixed at 0.0001. The parameters of the feature extractor are initialized with well-trained top 13 layers of VGG-16 [52] and others are initialized by Gaussian distribution with a 0.01 standard deviation. We train all the proposed models on all datasets with fixed batch size 16 and epoch 1000. To abundant the dataset, we first randomly scale the source image by a factor between 0.5 and 2. If the short side of the zoomed image is less than 512, we re-scale it to 512. Then we randomly flipped the image horizontally and cropped 512 × 512 patches. In the test stage, we take the source image as the input.

TABLE III

COMPARISON WITH STATE-OF-THE-ART CROWD COUNTING METHODS ON SHANGHAITECH DATASET. **AGCCM$^{\mathbf{S}}$** AND **AGCCM$^{\mathbf{C}}$** INDICATE MODELS COMBINED WITH ATTENTION-GUIDED COLLABORATIVE COUNTING MODULES BASED ON SPATIAL-TRANSFORMER AND CHANNEL-TRANSFORMER, RESPECTIVELY

Methods	SHA		SHB	
	MAE	RMSE	MAE	RMSE
AMRNet[53]	61.59	98.36	7.02	11.0
AGMF [54]	63.8	108.5	7.3	11.8
AMSNet[55]	56.7	93.4	6.7	10.2
MAN [56]	61.9	99.6	7.4	11.3
KDMG ([57])	63.8	99.2	7.8	12.7
ADSCNet [58]	55.4	99.7	6.4	11.3
ASNet [59]	57.78	90.13	-	-
RPNet [60]	61.2	96.9	8.1	11.6
LibraNet [61]	55.9	97.1	7.3	11.3
EFDC-18 [62]	55.4	91.3	6.9	10.3
DKPNet [63]	55.6	91.0	6.6	10.9
UEP [64]	54.64	91.15	6.38	10.88
P2PNet [65]	**52.74**	**85.06**	6.25	9.9
AGCCM^{S}	52.75	85.5	**5.98**	**9.72**
AGCCM^{C}	52.94	85.69	6.06	10.31

TABLE IV

COMPARISON WITH STATE-OF-THE-ART METHODS ON JHU-CROWD++

Methods	JHU-Crowd++	
	MAE	RMSE
DM-Count [66]	68.4	283.3
NoiseCC[67]	67.7	258.5
GLF[68]	59.9	259.5
KDMG[57]	69.7	268.3
AGCCM^{S}	**58.56**	**215.09**
AGCCM^{C}	58.97	247.05

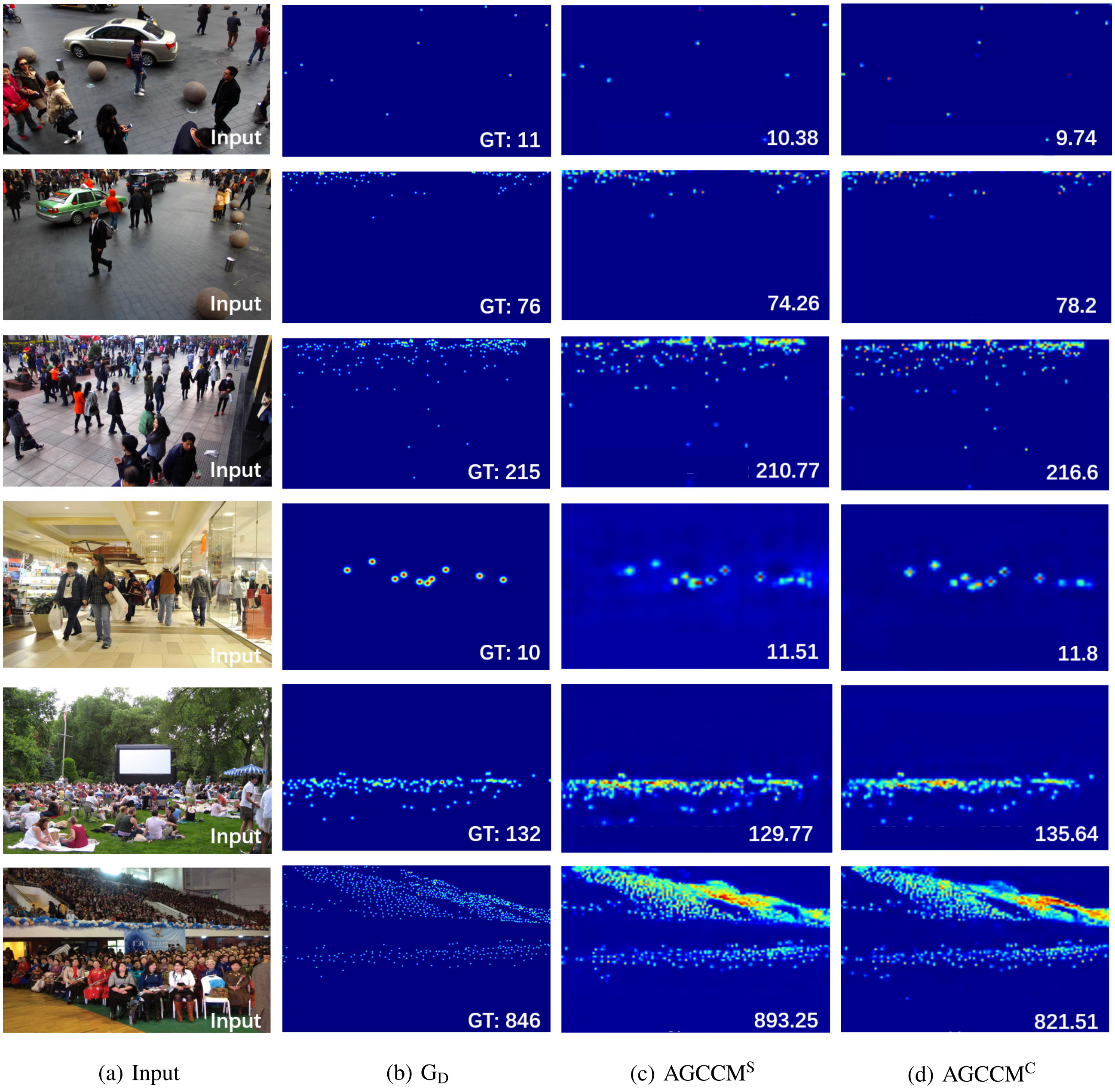
TABLE V

COMPARISON WITH STATE-OF-THE-ART METHODS ON RGBD

Methods	RGBD	
	MAE	RMSE
CSR [8]	4.91	7.11
RDNet[48]	4.96	7.22
AGD [4]	4.18	6.75
CSR+IDAM[10]	4.38	7.06
AGCCM^{S}	4.36	6.25
AGCCM^{C}	**3.90**	**5.86**

D. Comparisons With State-of-the-Arts

In Table III ∼ V, we compare our results to those of the method that returns the best results for each one of the 4 public datasets, as currently reported in the literature. They are those of [65], [65], [68], and [4], respectively. In each case, we reprint the results as given in these papers and add those of models based on multi-branch module combined with recombination and attention-guided module realized according to spatial-transformer and channel-transformer, respectively, as described in Section III. On SHA, our methods are slightly inferior to [65], ranking second and third. On the second and third dataset, the two models consistently and clearly outperform all other methods. As shown in Tabel V, models with AGCCM based on spatial-transformer and channel-transformer improve 7.4% and 13.2% in terms of the RMSE metric, respectively.



(a) Input (b) G_D (c) $AGCCM^S$ (d) $AGCCM^C$

Fig. 5. Some examples of $AGCCM^S$ and $AGCCM^C$ on RGBD and JHU-Crowd++. The top and bottom three samples are from RGBD and JHU-Crowd++, respectively. From left to right are the input image, the ground truth of density map, the predicted density map of $AGCCM^S$ and $AGCCM^C$, in order.

Tables III∼V also show that AGCCM based on spatial-transformer performs better than based on channel-transformer except for the RGBD dataset. As displayed in Figure 4, the RGBD dataset is leaner to sparse density when compared to the other three datasets. Sparsity may lead to large spatial variations. On the other hand, the channel transformer neglect the difference intra rows or columns, which may release this kind of spatial variation. Therefore, the channel transformer performs better than the spatial transformer on RGBD. Another abnormal phenomenon is that our methods slightly performed worse on SHA compared to P2PNet while achieving better results in SHB. As is known to all, SHA has a more dense density than SHB, and the contextual information of the crowd correlated well in a dense area, but the correlation might be poorer for the low-density areas of the crowd. Our transformer-based framework is adaptive for this situation with its long-term dependencies. Thus, our model performs better on the SHB dataset than P2PNet.

TABLE VI
RESULTS OF MODEL COMBINED WITH AGM OR CCM

AGM^S	AGM^C	CCM	SHA		SHB	
			MAE	RMSE	MAE	RMSE
			58.52	94.83	6.84	12.87
		√	56.91	90.8	6.53	10.36
√			54.21	91.05	6.46	11
	√		54.79	86.65	6.3	10.6
√		√	**52.75**	**85.5**	**5.98**	**9.72**
	√	√	52.94	85.69	6.06	10.31

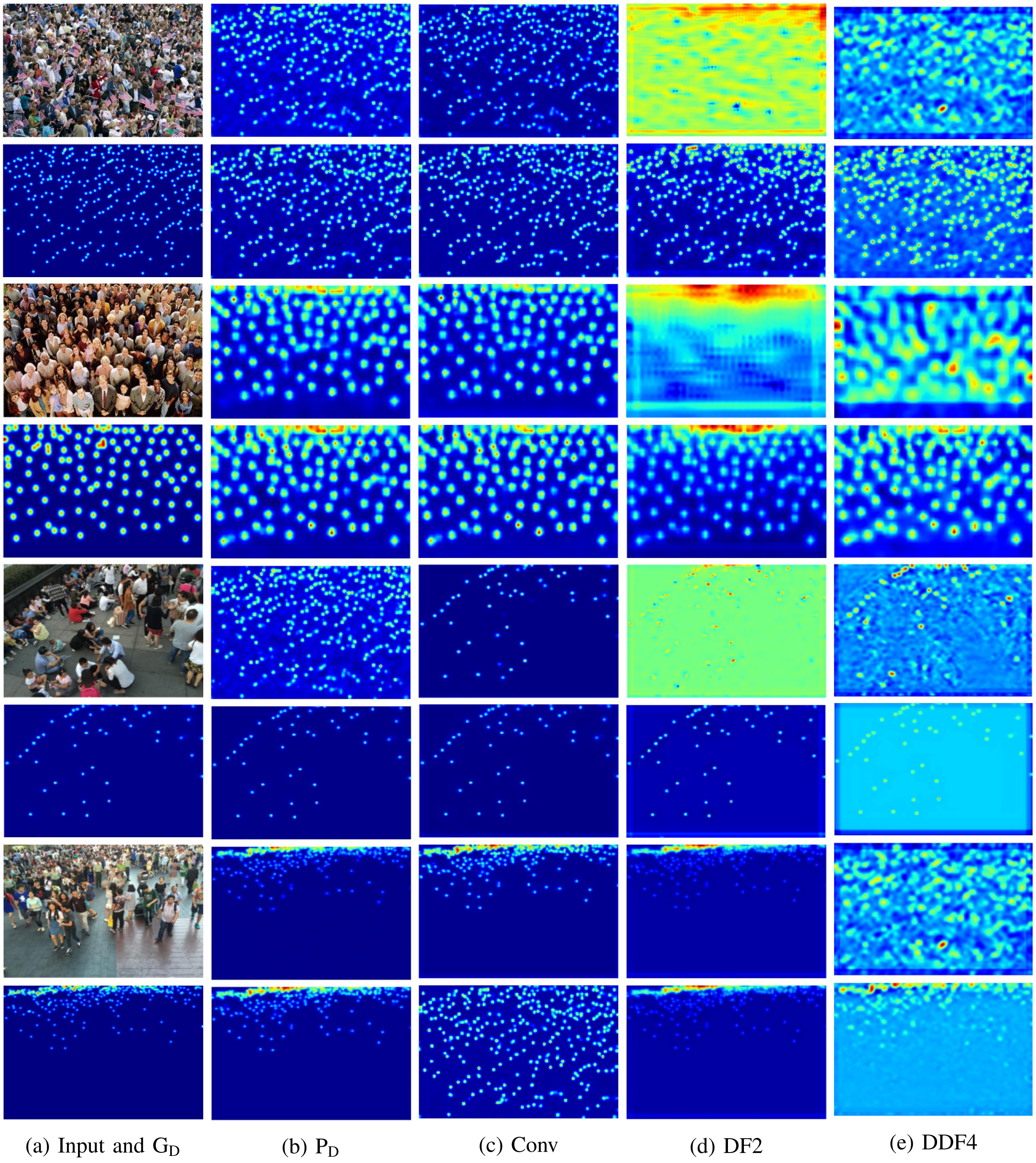


Fig. 6. Some samples' outputs of the multi-branch model without and with our AGCCM on ShanghaiTech dataset. Each sample shows the output of multi-branch module without AGCCM^S in the first row, and with AGCCM in the second row. From left to right, columns (b)~(e) exhibits the output of P_D, and outputs of branch Conv, DF2, and DDF4. The top and bottom three samples are from part_A and part_B, respectively.

E. Visualization

Figure 5 shows the predict density maps of the AGCCM^S and AGCCM^C on RGBD and JHU-Crowd++. The top and bottom three samples are from RGBD and JHU-Crowd++, respectively. The three chosen pictures have small to large densities from top to down. The comparison among the outputs and the ground truth demonstrates the reliability of our model. Moreover, we compare each branch's output of multi-branch model with and without AGCCM^S in Figure 6. Density maps, including the final prediction and each branch's output of the model with AGCCM, show stronger anti-noise ability in contradistinction to without. We regard this background noise filtering as additional welfare of assigning feature maps with different attention regions for each branch, the same as the phenomenon in [50] and [5]. Another remarkable phenomenon is that each branch's output of the model without AGCCM count the crowd in obviously different areas, while it is almost the same in the model with AGCCM but with another counting

TABLE VII

MAE of Multi-Branch Module Combined With AGM or CCM on Images With Different Crowd Density

Conv	DF2	DDF4	AGM^S	AGM^C	CCM	$[1\times10^{-4}\sim5\times10^{-4}]$	$[5\times10^{-4}\sim1\times10^{-3}]$	$[1\times10^{-3}\sim5\times10^{-3}]$
√						30.89	45.86	96.42
	√					27.23	48.57	102.36
		√				30.9	48.92	86.73
						29.67	47.9	95.17
√	√	√				30.02	47.83	93.48
√	√	√			√	33.64	44.52	88.04
√	√	√	√			27.57	47.81	89.63
√	√	√		√		30.68	47.22	82.36
√	√	√	√		√	25.47	43.47	84.57
√	√	√		√	√	24.56	43.01	86.31

contribution degree in other areas. Taking the first sample as an example, branch DF2 in the two models is more inclined to count people far from the camera. The difference is that branch DF2 of the model combined with the collaboration mechanism has a more definite collaborative counting with other branches. Therefore, we believe that our collaboration mechanism guides each branch to play its advantages in its expert area and promotes the teamwork of each branch in the collaboration area.

V. Ablation Study

This paper proposes an attention-guided collaborative counting model to promote the collaborative counting of branches in a multi-branch structure. It contains an AGM, a CCM, and a pair of loss functions. We utilize the model composed of VGG16 and our multi-branch module as the base framework and verify the performance of each module by adding them into the framework.

A. Effectiveness of AGM

The proposed AGM has two forms, one is based on spatial transformer, and the other is based on channel transformer. To exhibit the role of AGM, we show the results of a multi-branch model with or without CCM to combined with the two kinds of transformers in Table VI. It is obvious that the performance of the multi-branch model is improved when combined with AGM. For example, the MAE of the multi-branch module without and with CCM are reduced by 7.37% and 7.31%, respectively, when combining with AGM^S.

B. Effectiveness of CCM

Table VI shows that the CCM can obviously improve the performance of the multi-branch module. For example, combining with CCM, the MAE of a multi-branch module without and with AGM^S are reduced by 2.75% and 2.69%, respectively.

Besides, we show the performance of the multi-branch model with or without CCM on crowds with different densities in Table VII. The top three rows show models with only one branch's results on different crowd densities, and the following row exhibits the average of the three one-branch models. It shows the multi-branch model without any other strategies on different crowd densities is almost equal to the three branches' average. Moreover, combined with only attention-guided module or collaborative counting module cannot reach each branch's best performance simultaneously, but combined with both can.

TABLE VIII

Results of AGCCM With Different λ

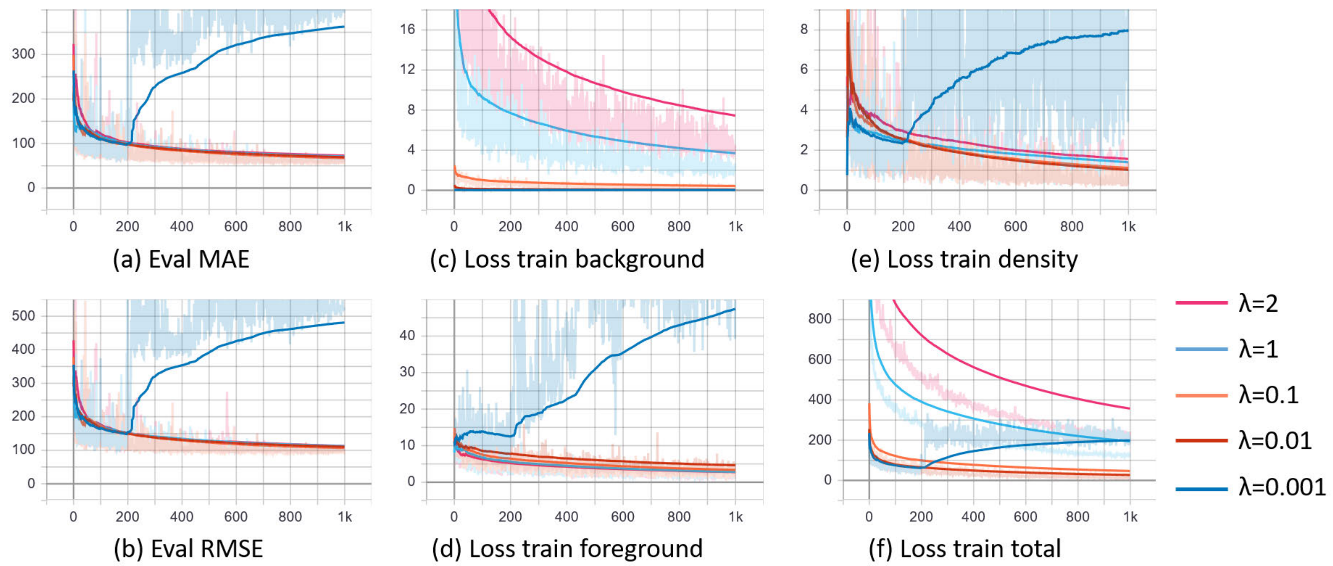
Methods	λ	SHA		SHB	
		MAE	RMSE	MAE	RMSE
AGCCM^S	2	55.04	92.47	6.37	10.45
	1	54.01	89.83	6.2	10.28
	0.1	**52.75**	**85.5**	**5.98**	**9.72**
	0.01	53.75	88.34	6.13	11.37
	0.001	62.94	96.48	7.93	13.54
AGCCM^C	2	55.87	88.87	6.24	11.1
	1	**52.94**	**85.69**	**6.06**	**10.31**
	0.1	53.96	89.13	6.07	10.54
	0.01	54.67	86.99	6.29	10.49
	0.001	59.09	97.83	7.41	12.44

C. Robustness of λ

In Equation 17, an external parameter λ is introduced to balance the loss of attention maps and density maps. We exhibit MAE of models with spatial-transformer and channel-transformer under different λ in Table VIII. The two models are robust to λ range between 0.1 and 1. Besides, the model related to spatial-transformer and channel-transformer act best at λ be 0.1 and 1, respectively, and We fixed them to train other datasets. Noticeably, when the λ is very small, the performance of both models drops sharply. In order to further describe the impact of λ, we show the training process of the model with AGM based on spatial-transformer under different λ in Figure 7. When $\lambda = 0.001$, the counting error on the test dataset, the training loss of density map, and total training loss consistently first decrease and then rise abnormally, while foreground loss fluctuates within a certain range inside about 200 epochs then rises significantly. Obviously, the significantly rising of the density loss and the counting error is caused by the surge of foreground error. We believe that this is because the attention map not only acts as a background filter (as analyzed in Section V) but also plays vital role in distinguishing expert areas for each branch.

D. Bi-Transformers vs. Other Transformer Variants

To verify the effectiveness of Bi-Transformers, we provide an algorithmic comparison with ViT [39] and Swin



(a) Eval MAE (c) Loss train background (e) Loss train density

(b) Eval RMSE (d) Loss train foreground (f) Loss train total

Fig. 7. Train process of AGCCM^S with different λ on SHA. The x-axis is the number of iterations. The y-axis on Eval shows MAE and RMSE, respectively.

TABLE IX

RESULTS OF AGCCM BASED ON DIFFERENT TRANSFORMER VARIANTS

Transformer	SHA		SHB	
	MAE	RMSE	MAE	RMSE
ViT[39]	55.89	89.99	6.58	11.31
Swin[40]	54.86	87.7	6.61	10.8
AGCCM^S	**52.75**	**85.5**	**5.98**	**9.72**
AGCCM^C	52.94	85.69	6.06	10.31

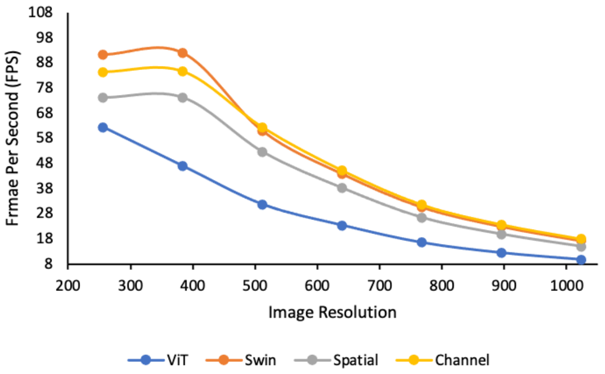
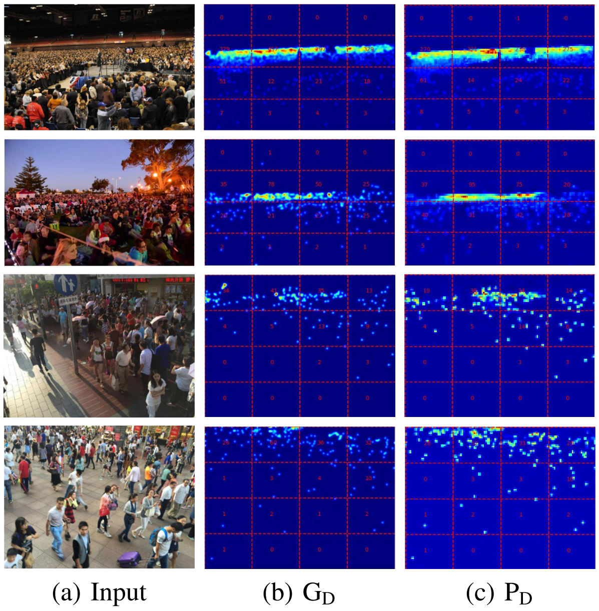


Fig. 8. Illustration of FPS on different resolution images of the model based on different transformer variants.



(a) Input (b) G_D (c) P_D

Fig. 9. Some failure cases of AGCCM^C on part_A and part_B.

Transformer [40]. We set the following parameters in ViT and Swin to ensure a fair comparison. The dimension of embedding and MLP is 512 and 1024, respectively. The head number is 1. Besides, we set patch size in ViT as 2 × 2 and window size in Swin as 7 × 7. The ViT and Swin Transformer block code is respectively based on [39][1] and [40].[2] In addition, we learn from [44] converting ViT to a dense prediction model by reshaping the patch encoding to a 2D feature map and a bilinear upsampling of the feature map to the original input size. Table IX exhibits the results of AGCCM based on different transformer variants. When changing the bi-transformer to ViT or Swin, the MAE increased. We think that is mainly because the attention in ViT and Swin lacks distribution within patches and between windows, respectively. Another phenomenon worth noting is that when increasing the patch size to 4 × 4, the training process of AGCCM based on ViT does not converge. It may be caused by the absence of attention distribution within the patch and a high upsample rate.

Besides, we compare the frame per second (FPS) of our framework based on different transformer variants. The results illustrated in Figure 8 show that the FPS of the channel transformer and the spatial transformer is higher than ViT and almost equal to Swin.

E. Failure Cases

Figure 9 displays some examples having higher MAE than the average on the whole dataset. We show the ground truths

[1] https://github.com/lucidrains/vit-pytorch

[2] https://github.com/microsoft/Swin-Transformer

and the estimated numbers of individuals under each block in (b) and (c), respectively. It can be found that a block containing more objects tends to have more counts error. We can see people in these blocks with high error in the original image that tends to occupy a small area in the picture, even too tiny to distinguish by a human. The low quality of the input image appears as a more straightforward problem than the various head scales. Still, it turns out to be even more challenging because of the reconstruction errors [66] when the middle representation of the ground truth density map is based on fixed kernel size.

VI. Conclusion

This paper proposes a collaborative mechanism including an attention-guided module (AGM) and collaborative counting module (CCM). The proposed AGM assists in allocating each branch's expert area by assigning weight maps from a global view. Specially, we design two bidirectional transformers (i.e., spatial-transformer and channel-transformer) to achieve global attention distributions, which enable any resolution input without patch cropping. The CCM encourages each branch to focus on its expertise area and sharing counts on its sub-optimal area. In addition, our loss implicitly guides the model to distinguish the advantageous areas of each branch without additional label or crowd division. Experiments show that the proposed collaborative mechanism effectively promotes collaboration between branches. Moreover, AGCCM based on the proposed Bi-Transformer has a comparable speed to based Swin and superior performance.

References

[1] Y. Zhang, D. Zhou, S. Chen, S. Gao, and Y. Ma, "Single-image crowd counting via multi-column convolutional neural network," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 589–597.

[2] D. B. Sam, S. Surya, and R. V. Babu, "Switching convolutional neural network for crowd counting," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 4031–4039.

[3] M. Xu *et al.*, "Depth information guided crowd counting for complex crowd scenes," *Pattern Recognit. Lett.*, vol. 125,, pp. 563–569, Jul. 2019.

[4] X. Pan, H. Mo, Z. Zhou, and W. Wu, "Attention guided region division for crowd counting," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process. (ICASSP)*, May 2020, pp. 2568–2572.

[5] H. Mo *et al.*, "Background noise filtering and distribution dividing for crowd counting," *IEEE Trans. Image Process.*, vol. 29, pp. 8199–8212, 2020.

[6] J. Liu, C. Gao, D. Meng, and A. G. Hauptmann, "DecideNet: Counting varying density crowds through attention guided detection and density estimation," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 5197–5206.

[7] Y. Wang, S. Hu, G. Wang, C. Chen, and Z. Pan, "Multi-scale dilated convolution of convolutional neural network for crowd counting," *Multimedia Tools Appl.*, vol. 79, no. 1, pp. 1057–1073, 2020.

[8] Y. Li, X. Zhang, and D. Chen, "CSRNet: Dilated convolutional neural networks for understanding the highly congested scenes," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 1091–1100.

[9] Z. Shen, Y. Xu, B. Ni, M. Wang, J. Hu, and X. Yang, "Crowd counting via adversarial cross-scale consistency pursuit," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 5245–5254.

[10] L. Liu, J. Chen, H. Wu, G. Li, C. Li, and L. Lin, "Cross-modal collaborative representation learning and a large-scale RGBT benchmark for crowd counting," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2021, pp. 4823–4833.

[11] P. Dollár, C. Wojek, B. Schiele, and P. Perona, "Pedestrian detection: An evaluation of the state of the art," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 34, no. 4, pp. 743–761, Apr. 2012.

[12] T.-Y. Lin, P. Dollar, R. Girshick, K. He, B. Hariharan, and S. Belongie, "Feature pyramid networks for object detection," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 2117–2125.

[13] R. M. Haralick and L. G. Shapiro, "Image segmentation techniques," *Comput. Vis., Graph., Image Process.*, vol. 29, no. 1, pp. 100–132, 1985.

[14] J. Wang *et al.*, "Deep high-resolution representation learning for visual recognition," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 43, no. 10, pp. 3349–3364, Oct. 2021.

[15] W. Liu, M. Salzmann, and P. Fua, "Context-aware crowd counting," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2019, pp. 5099–5108.

[16] V. A. Sindagi and V. M. Patel, "Generating high-quality crowd density maps using contextual pyramid CNNs," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Oct. 2017, pp. 1861–1870.

[17] X. Cao, Z. Wang, Y. Zhao, and F. Su, "Scale aggregation network for accurate and efficient crowd counting," in *Proc. Eur. Conf. Comput. Vis. (ECCV)*, 2018, pp. 734–750.

[18] V. Ranjan, H. Le, and M. Hoai, "Iterative crowd counting," in *Proc. Eur. Conf. Comput. Vis. (ECCV)*, 2018, pp. 270–285.

[19] L. Zhang, M. Shi, and Q. Chen, "Crowd counting via scale-adaptive convolutional neural network," in *Proc. IEEE Winter Conf. Appl. Comput. Vis. (WACV)*, Mar. 2018, pp. 1113–1121.

[20] X. Jiang *et al.*, "Crowd counting and density estimation by trellis encoder–decoder networks," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2019, pp. 6133–6142.

[21] X. Zhang *et al.*, "DCNAS: Densely connected neural architecture search for semantic image segmentation," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2021, pp. 13956–13967.

[22] H. Idrees *et al.*, "Composition loss for counting, density map estimation and localization in dense crowds," in *Proc. Eur. Conf. Comput. Vis. (ECCV)*, 2018, pp. 532–546.

[23] V. A. Sindagi and V. M. Patel, "CNN-based cascaded multi-task learning of high-level prior and density estimation for crowd counting," in *Proc. 14th IEEE Int. Conf. Adv. Video Signal Based Surveill. (AVSS)*, Aug. 2017, pp. 1–6.

[24] X. Zhang *et al.*, "Hand image understanding via deep multi-task learning," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2021, pp. 11281–11292.

[25] L. Zhang *et al.*, "Nonlinear regression via deep negative correlation learning," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 43, no. 3, pp. 982–998, Mar. 2021.

[26] C. Szegedy *et al.*, "Going deeper with convolutions," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2015, pp. 1–9.

[27] I. Goodfellow *et al.*, "Generative adversarial networks," *Commun. ACM*, vol. 63, no. 11, pp. 139–144, 2020.

[28] V. Mnih *et al.*, "Recurrent models of visual attention," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 27, 2014, pp. 1–9.

[29] J. Hu, L. Shen, and G. Sun, "Squeeze-and-excitation networks," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 7132–7141.

[30] Z. Shen *et al.*, "Human-aware motion deblurring," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2019, pp. 5572–5581.

[31] W. Wang, J. Shen, J. Xie, M. Cheng, H. Ling, and A. Borji, "Revisiting video saliency prediction in the deep learning era," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 43, no. 1, pp. 220–237, Jan. 2021.

[32] A. Vaswani *et al.*, "Attention is all you need," in *Proc. Adv. Neural Inf. Process. Syst.*, 2017, pp. 5998–6008.

[33] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "BERT: Pre-training of deep bidirectional transformers for language understanding," 2018, *arXiv:1810.04805*.

[34] X. Qiu, T. Sun, Y. Xu, Y. Shao, N. Dai, and X. Huang, "Pre-trained models for natural language processing: A survey," *Sci. China Technol. Sci.*, vol. 63, no. 10, pp. 1872–1897, 2020.

[35] Y. Fan *et al.*, "Multi-branch attentive transformer," 2020, *arXiv:2006.10270*.

[36] Y. Tay, D. Bahri, D. Metzler, D.-C. Juan, Z. Zhao, and C. Zheng, "Synthesizer: Rethinking self-attention for transformer models," in *Proc. Int. Conf. Mach. Learn.* 2021, pp. 10183–10192.

[37] N. Parmar *et al.*, "Image transformer," in *Proc. Int. Conf. Mach. Learn.*, 2018, pp. 4055–4064.

[38] Y. Tay, D. Bahri, L. Yang, D. Metzler, and D.-C. Juan, "Sparse sinkhorn attention," in *Proc. Int. Conf. Mach. Learn.*, 2020, pp. 9438–9447.

[39] A. Dosovitskiy *et al.*, "An image is worth 16×16 words: Transformers for image recognition at scale," 2020, *arXiv:2010.11929*.

[40] Z. Liu *et al.*, "Swin transformer: Hierarchical vision transformer using shifted Windows," 2021, *arXiv:2103.14030*.

[41] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 770–778.

[42] X. He, Y. Chen, and Z. Lin, "Spatial-spectral transformer for hyperspectral image classification," *Remote Sens.*, vol. 13, no. 3, p. 498, Jan. 2021.

[43] D. Zhou *et al.*, "DeepViT: Towards deeper vision transformer," 2021, *arXiv:2103.11886*.

[44] R. Strudel, R. Garcia, I. Laptev, and C. Schmid, "Segmenter: Transformer for semantic segmentation," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2021, pp. 7262–7272.

[45] Y. Tian, X. Chu, and H. Wang, "CCTrans: Simplifying and improving crowd counting with transformer," 2021, *arXiv:2109.14483*.

[46] U. Sajid, X. Chen, H. Sajid, T. Kim, and G. Wang, "Audio-visual transformer based crowd counting," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. Workshops (ICCVW)*, Oct. 2021, pp. 2249–2259.

[47] T.-Y. Lin, P. Goyal, R. Girshick, K. He, and P. Dollar, "Focal loss for dense object detection," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Oct. 2017, pp. 2980–2988.

[48] D. Lian, J. Li, J. Zheng, W. Luo, and S. Gao, "Density map regression guided detection network for RGB-D crowd counting and localization," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2019,, pp. 1821–1830.

[49] V. Sindagi, R. Yasarla, and V. M. M. Patel, "JHU-CROWD++: Large-scale crowd counting dataset and a benchmark method," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 44, no. 5, pp. 2594–2609, May 2022.

[50] L. Zhu, Z. Zhao, C. Lu, Y. Lin, Y. Peng, and T. Yao, "Dual path multi-scale fusion networks with attention for crowd counting," 2019, *arXiv:1902.01115*.

[51] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," 2014, *arXiv:1412.6980*.

[52] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," 2014, *arXiv:1409.1556*.

[53] X. Liu, J. Yang, W. Ding, T. Wang, Z. Wang, and J. Xiong, "Adaptive mixture regression network with local counting map for crowd counting," in *Proc. Eur. Conf. Comput. Vis.* Glasgow, U.K.: Springer, 2020, pp. 241–257.

[54] B. Zhang, N. Wang, Z. Zhao, A. Abraham, and H. Liu, "Crowd counting based on attention-guided multi-scale fusion networks," *Neurocomputing*, vol. 451, pp. 12–24, Sep. 2021.

[55] Y. Hu *et al.*, "Nas-count: Counting-by-density with neural architecture search," in *Proc. Eur. Conf. Comput. Vis.* Glasgow, U.K.: Springer, 2020, pp. 747–766.

[56] J. Wan and A. Chan, "Modeling noisy annotations for crowd counting," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 33, 2020, pp. 3386–3396.

[57] J. Wan, Q. Wang, and A. B. Chan, "Kernel-based density map generation for dense object counting," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 44, no. 3, pp. 1357–1370, Mar. 2022.

[58] S. Bai, Z. He, Y. Qiao, H. Hu, W. Wu, and J. Yan, "Adaptive dilated network with self-correction supervision for counting," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2020, pp. 4594–4603.

[59] X. Jiang *et al.*, "Attention scaling for crowd counting," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2020, pp. 4706–4715.

[60] Y. Yang, G. Li, Z. Wu, L. Su, Q. Huang, and N. Sebe, "Reverse perspective network for perspective-aware object counting," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2020, pp. 4374–4383.

[61] L. Liu, H. Lu, H. Zou, H. Xiong, Z. Cao, and C. Shen, "Weighing counts: Sequential crowd counting by reinforcement learning," in *Proc. Eur. Conf. Comput. Vis.* Cham, Switzerland: Springer, 2020, pp. 164–181.

[62] X. Liu *et al.*, "Exploiting sample correlation for crowd counting with multi-expert network," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2021, pp. 3215–3224.

[63] B. Chen *et al.*, "Variational attention: Propagating domain-specific knowledge for multi-domain learning in crowd counting," in *Proc. IEEE Int. Conf. Comput. Vis.*, Oct. 2021, pp. 16065–16075.

[64] C. Wang *et al.*, "Uniformity in heterogeneity: Diving deep into count interval partition for crowd counting," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2021, pp. 3234–3242.

[65] Q. Song *et al.*, "Rethinking counting and localization in crowds: A purely point-based framework," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2021, pp. 3365–3374.

[66] B. Wang, H. Liu, D. Samaras, and M. Hoai, "Distribution matching for crowd counting," 2020, *arXiv:2009.13077*.

[67] H. Mei, T. Wan, and J. Eisner, "Noise-contrastive estimation for multivariate point processes," 2020, *arXiv:2011.00717*.

[68] J. Wan, Z. Liu, and A. B. Chan, "A generalized loss function for crowd counting and localization," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2021, pp. 1974–1983.

Hong Mo received the M.S. degree in computer science from the Huazhong University of Science and Technology in 2016. She is currently pursuing the Ph.D. degree with the State Key Laboratory of Virtual Reality Technology and Systems, Beihang University, China. Her research focused on deep learning and computer vision.

Wenqi Ren (Member, IEEE) received the Ph.D. degree from Tianjin University, Tianjin, China, in 2017. From 2015 to 2016, he was supported by the China Scholarship Council and working with Prof. Ming-Husan Yang as a joint-training Ph.D. student at the Electrical Engineering and Computer Science Department, University of California Merced, Merced. He is currently an Associate Professor with the School of Cyber Science and Technology, Sun Yat-sen University, Shenzhen Campus, Shenzhen, China. His research interests include image processing and related high-level vision problems.

Xiong Zhang received the M.S. degree in computer science and technology from Beihang University in 2015. He is currently the Director of the Perception Team, Neolix Autonomous Vehicle. His research interests include computer vision and machine learning.

Feihu Yan received the Ph.D. degree in computer science from the State Key Laboratory of Virtual Reality Technology and Systems, Beihang University, Beijing, China, in 2021. He is currently a Lecturer with the School of Electrical and Information Engineering, Beijing University of Civil Engineering and Architecture, China. His current research interests include computer vision, 3D reconstruction, and SLAM.

Zhong Zhou (Member, IEEE) received the B.S. degree from Nanjing University in 1999 and the Ph.D. degree from Beihang University, Beijing, China, in 2005. He is currently a Professor and a Ph.D. Adviser with the State Key Laboratory of Virtual Reality Technology and Systems, Beihang University. His main research interests include virtual reality/augmented reality/mixed reality, computer vision, and artificial intelligence.

Xiaochun Cao (Senior Member, IEEE) received the B.E. and M.E. degrees in computer science from Beihang University, Beijing, China, and the Ph.D. degree in computer science from the University of Central Florida, Orlando, FL, USA. He spent about three years with ObjectVideo Inc., as a Research Scientist. From 2008 to 2012, he was a Professor with Tianjin University, Tianjin, China. He has been a Professor with the Institute of Information Engineering, Chinese Academy of Sciences, Beijing, since 2012. He is currently a Professor with the School of Cyber Science and Technology, Sun Yat-sen University, Shenzhen Campus, Shenzhen, China. He has authored or coauthored over 120 journal articles and conference papers. His dissertation was nominated for the University of Central Florida's University-Level Outstanding Dissertation Award. In 2004 and 2010, he was a Recipient of the Piero Zamperoni Best Student Article Award at the International Conference on Pattern Recognition. He is an Editorial Board of the IEEE TRANSACTIONS ON IMAGE PROCESSING, IEEE TRANSACTIONS ON MULTIMEDIA, and IEEE TRANSACTIONS ON CIRCUITS AND SYSTEMS FOR VIDEO TECHNOLOGY.

Wei Wu received the Ph.D. degree from the Harbin Institute of Technology, Harbin, China, in 1995. He is currently a Professor with the State Key Laboratory of Virtual Reality Technology and Systems, Beihang University. He is the Chair of the Technical Committee on Virtual Reality and Visualization, China Computer Federation. His current research interests include virtual reality, wireless networking, and distributed interactive systems.