Computational Visual Media
https://doi.org/10.1007/s41095-022-0313-5

Research Article

DTCC: Multi-level dilated convolution with transformer for weakly-supervised crowd counting

Zhuangzhuang Miao[1], Yong Zhang[1] (✉), Yuan Peng[2], Haocheng Peng[1], and Baocai Yin[1]

Abstract Crowd counting provides an important foundation for public security and urban management. Due to the existence of small targets and large density variations in crowd images, crowd counting is a challenging task. Mainstream methods usually apply convolution neural networks (CNNs) to regress a density map, which requires annotations of individual persons and counts. Weakly-supervised methods can avoid detailed labeling and only require counts as annotations of images, but existing methods fail to achieve satisfactory performance because a global perspective field and multi-level information are usually ignored. We propose a weakly-supervised method, DTCC, which effectively combines multi-level dilated convolution and transformer methods to realize end-to-end crowd counting. Its main components include a recursive swin transformer and a multi-level dilated convolution regression head. The recursive swin transformer combines a pyramid visual transformer with a fine-tuned recursive pyramid structure to capture deep multi-level crowd features, including global features. The multi-level dilated convolution regression head includes multi-level dilated convolution and a linear regression head for the feature extraction module. This module can capture both low- and high-level features simultaneously to enhance the receptive field. In addition, two regression head fusion mechanisms realize dynamic and mean fusion counting. Experiments on four well-known benchmark crowd counting datasets (UCF_CC_50, ShanghaiTech, UCF_QNRF, and JHU-Crowd++) show that DTCC achieves results superior to other weakly-supervised methods and comparable to fully-supervised methods.

Keywords crowd counting; transformer; dilated convolution; global perspective field; pyramid

1 Introduction

Crowd counting is an important topic in the field of crowd analysis: the aim is to estimate the number of people in an image. With increasing population and urbanization, there are more and more crowd-containing localities: e.g., subway platforms, bus stations, airports, tourist attractions, and shopping malls. Crowd congestion can occur during peak hours, with a serious negative impact on public safety. Accurate crowd counting can help to avoid crowd congestion, and plays an essential role in public security, abnormal situation warning, and pedestrian control.

Significant progress has been made in crowd counting via computer vision through years of relevant research. As Fig. 1 shows, existing crowd counting methods can be classified as depending on object detection, density estimation, point-supervision, and weak-supervision. Deep learning-based methods can also be divided into CNN-based and transformer-based methods. In an earlier study, researchers used object detection to solve the crowd counting problem [1, 2]. However, such methods do not work for dense scenes: severe occlusion and complex backgrounds typically occur in such cases, leading to unsatisfactory results. To solve these problems, some regression-based approaches have appeared. They usually learn low-level features (e.g., texture features, edge feature, etc.) using traditional algorithms and map features to

1 Beijing Key Laboratory of Multimedia and Intelligent Software Technology, Beijing Institute of Artificial Intelligence, Faculty of Information Technology, Beijing University of Technology, Beijing 100124, China. E-mail: Z. Miao, mzhuangzhuang2023@163.com; Y. Zhang, zhangyong2010@bjut.edu.cn (✉); H. Peng, haocheng.peng@ucdconnect.ie; B. Yin, ybc@bjut.edu.cn.

2 Taiji Computer Corporation Ltd., China. E-mail: yuan.peng@outlook.com.

Manuscript received: 2022-03-22; accepted: 2022-09-12

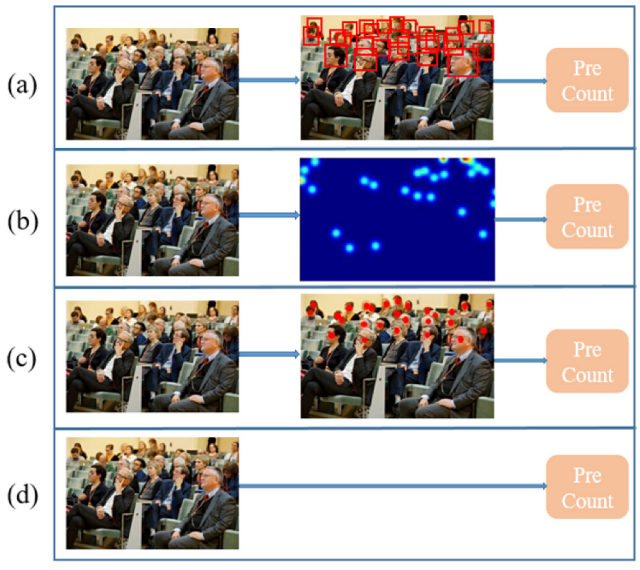


Fig. 1 Mainstream crowd counting methods can be classified as depending on (a) object detection, (b) density estimation, (c) point-supervision, and (d) weak-supervision.

the number of persons in the crowd through regression models. However, these methods ignore crowd distribution information in the image. To make use of it, Lempitsky and Zisserman [3] proposed a method based on density estimation which learns a linear or nonlinear mapping between image features and density maps. Nonetheless, the features extracted by traditional methods cannot capture deep-level feature representations. Therefore, Walach and Wolf [4] and others [5, 6] have used CNN-based approaches to regress density maps. The powerful image feature extraction capability of CNNs enables these methods to achieve better results. Nowadays, CNN-based methods have become mainstream for dense scenes.

Recent CNN-based fully-supervised methods [7–9] achieve excellent results; they require both a count and annotation of individual people as supervision. These methods generate the true density map from individual annotation and regress the predicted density map. Nevertheless, detailed individual labeling is tedious, limiting its application. Therefore, it is fundamental to find a method that can obtain precise results simply using crowd counts as annotations. Corresponding deep learning-based weakly-supervised methods have thus emerged [10, 11]. However, these existing weakly-supervised methods usually ignore the extraction of global receptive fields and multi-level information; they predict the total count directly from the entire image, so the global receptive field is important for these methods. A CNN is limited to extracting a global receptive field without using a density map due to the characteristics of local feature extraction. In 2021, a transformer was introduced to the weakly-supervised crowd counting task [12]. The global attention of the corresponding network can effectively overcome the limited receptive field of CNN-based methods. However, this work cannot effectively extract multi-level information about the target. Figure 2 shows an image with targets of different sizes in the two regions marked in red. Thus, for weakly-supervised crowd counting, multi-level information is very important. Sufficient features cannot be learned to regress counting if multi-level information is not properly utilized.

This paper proposes DTCC, a pyramid vision transformer network for weakly-supervised crowd counting. It comprises a transformer feature extraction module and a multi-level dilated convolution regression module. The main contributions of this paper are:

(1) DTCC, a multi-level transformer dilated convolution weakly-supervised framework, which is capable of accurate end-to-end crowd counting.
(2) A multi-level crowd information feature extraction module for dense prediction. The final feature representation can distinguish between dense crowd heads and larger scale crowd heads. The overall framework is a recursive pyramid structure, which combines a pyramid vision transformer backbone network and a fine-tuned recursive pyramid structure (recursive fine-FPN) to obtain multi-level contextual crowd information.
(3) A multi-level dilated convolution regression module, which can enhance the receptive field for features and capture stronger global features. It is combined with two networks, DTCC-Dynamic

Fig. 2 An image containing a crowd with people at different scales.

and DTCC-Mean, as multi-level regression heads adapted to different crowd scenes.

(4) Experiments on four well-known benchmark datasets demonstrating better accuracy than other weakly-supervised methods, with competitive results to mainstream fully-supervised methods.

2 Related work

2.1 Background

Crowd counting approaches can be divided into two categories: fully-supervised and weakly-supervised methods. Fully-supervised methods use a density map as supervisory information to train the model, which requires point-level annotations of the crowd. Weakly-supervised methods only need a count of the crowd. Mainstream crowd counting methods usually utilize CNN to regress a density map [23, 26, 44]. The success of transformer-based methods in computer vision tasks such as image classification [13–15], object detection [13, 16, 17], and image segmentation suggests use of a transformer framework as the backbone network for crowd counting.

2.2 Fully-supervised crowd counting

CNN-based methods regress a density map and obtain the total number of people in the crowd by integrating the density map. Zhang et al. [18] proposed a network with three differently-sized receptive fields, which was capable of learning multi-level crowd features. This method replaces the fully connected layer by a convolution layer and can modify the size of the input images. Sam et al. [19] proposed a selective CNN with several convolution kernels of different sizes as the density map regression head; a selection classifier selects the optimal regression head for the input to predict the result. Li et al. [20] presented a deeper framework with convolution layers as the backbone, based on a combination of VGG16 and dilated convolution layers to expand the receptive field. It extracted deeper features without losing resolution. Later advances considered new density map loss functions with better results: Ma et al. [21] illustrated a point-supervised loss function for crowd count estimation, converting a sparse point labeling into a ground truth density map using a Gaussian kernel. This was used as a learning target to train the density map estimator. Liu et al. [22] used a swin transformer as the backbone network and a top–down fusion mechanism to fully utilize the various spatial information extracted from different stages of the model. Abousamrad et al. [24] reported a method that uses topological constraints instead of binary region maps to compute L2 loss functions for the head and background region. Only a few methods are based on transformer networks to realize fully-supervised crowd counting. Among them, Liang et al. [23] proposed an elegant, end-to-end crowd localization transformer that solves the task using a regression-based paradigm. Sun et al. [25] investigated the role of global contextual information in crowd counting. This method extracts global information from overlapping image blocks using a transformer, and adds contextual tags to the input sequence. In addition, a token attention module and regression token module are proposed to predict the total number of people in images. Gao et al. [26] showed a dilated convolutional transformer method, introducing a window-based vision transformer for crowd counting.

In summary, fully-supervised crowd counting methods have been extensively studied and have achieved good results. However, the application of fully-supervised methods to specific scenes is very limited, because they require individual annotation to generate density maps, and it is tedious and difficult to perform accurate individual annotation for dense scenes.

2.3 Weakly-supervised crowd counting

Weakly-supervised counting methods just rely on crowd counts for training. Shang et al. [27] proposed an end-to-end CNN architecture that exploits shared computation over overlapping regions. Wang et al. [28] presented a novel and efficient counter, which explores embedded global dependency modeling and total count regression by designing a multi-granularity regressor. Lei et al. [29] suggested a new multi-assisted task training strategy, MATT, which learns from a few images with individual annotations and many simply with counts to obtain more accurate predictions. Transformers have an inherent advantage in weakly-supervised crowd counting, since they can enhance global information about features and capture contextual knowledge. TransCrowd [12] was the first transformer-based crowd counting framework, which reformulates the counting problem from a

sequential perspective to a counting perspective. CCTrans [31] is applicable to both fully-supervised and weakly-supervised data, and uses Twins [32] as a feature extraction framework. It combines the features of multiple stages of the Twins network through multi-level dilated convolutions for feature fusion, finally predicting the number of people through a regression head. Savner and Kanhangad [52] proposed an architecture based on a pyramid vision transformer network to extract multi-scale features with global context. Wang et al. [53] proposed a joint CNN and transformer network based on weakly-supervised learning to reduce the number of parameters and overcome the problem of target segmentation.

Without annotations of individuals, weakly-supervised crowd counting is challenging. Existing weakly-supervised methods cannot extract sufficient global features and multi-level information, leading to the loss of collective semantic information and a failure to provide rich global features for the final regression. Using a global attention mechanism provides a new way to design an effective weakly-supervised crowd counting model.

3 Method

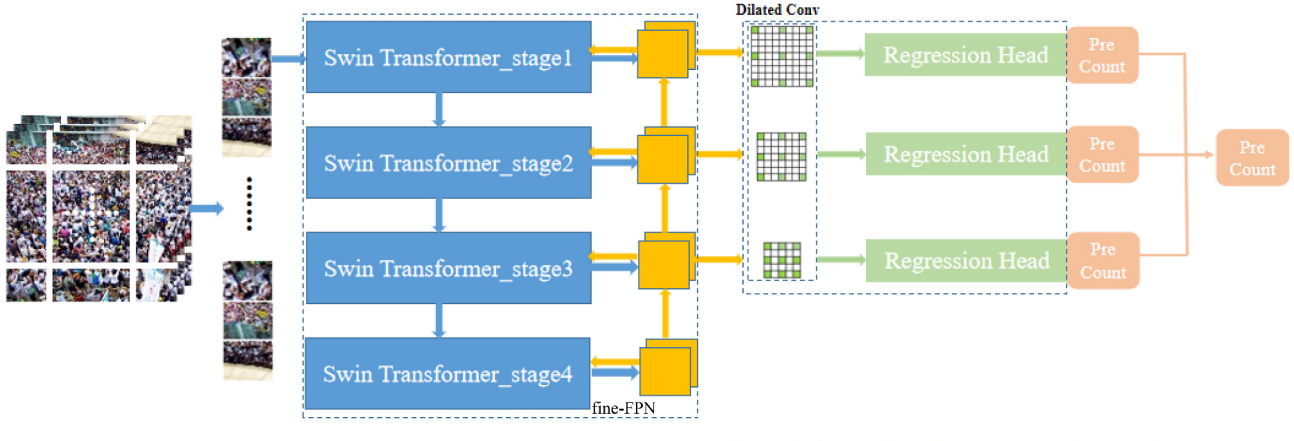
3.1 Approach

Existing weakly-supervised crowd counting methods have two problems to be solved: extraction of a global receptive field and utilization of multi-level information. Therefore, this paper introduces a swin transformer to capture global features. A feature pyramid structure is also introduced to enrich the multi-level feature representation, so that single-level features contain rich multi-level information. In addition, since the window attention mechanism of the swin transformer processes image patches, this alleviates the problem of uneven distribution of the crowd to a certain extent. To enhance the receptive field of features, a multi-level dilated convolution module is designed for the swin transformer, to solve the problem of local domain loss by dilated convolutions. Based on the above ideas, we propose DTCC, an end-to-end weakly-supervised method for crowd counting, which can provide accurate crowd counts based only on crowd count annotations.

3.2 Network architecture of DTCC

The framework of DTCC is shown in Fig. 3. The input image is divided into blocks of the same size and converted into a 1D sequence for the swin transformer. DTCC is composed of two main modules. The recursive swin transformer feature extraction module consists of a swin transformer [13] and the recursive fine-FPN. The multi-level dilated convolution regression head module consists of a multi-level dilated convolution and a linear regression head. The counting results from multi-level feature regression are given different weights to obtain the final count.

For feature extraction, the swin transformer is composed of a transformer-encoder. Therefore, the 2D image structure must be converted to a 1D sequence required as input to a transformer-encoder. This network is commonly used in natural language processing, but can also get good results



(a) Recursive swin transformer (b) Multi-level dilated conv regression head

Fig. 3 Network architecture of DTCC.

in computer vision by using ViT [30] to solve the input problem.

The input image to the swin transformer is defined as $X \in \mathbb{R}^{H\times W\times C}$, where H, W, C represent the height and width of the image and the number of channels, respectively. Firstly, the image is divided into image patches of size $P \times P$. Thus the image of size $H \times W \times C$ is divided into patches $X' \in \mathbb{R}^{N\times P\times P\times C}$, where $N = (H/P)\times(W/P)$. Then, each image patch is linearly transformed into a sequence of length $P^2 \times C$ for input to the model. Therefore, the input image is transformed into $Z \in \mathbb{R}^{N\times T}$ by preprocessing, where $T = P^2 \times C$. The feature extraction backbone network of swin transformer calculates local window attention, and performs image embedding operations in the window for each image patch. It uses two operations for dividing windows. The image is first divided into image patches, and then the image patches are divided by moving windows. This method moves non-overlapping local windows, which reduces computational complexity to linear in image size.

3.3 Recursive swin transformer

3.3.1 Approach

The recursive swin transformer (RST) effectively combines the pyramid structure transformer backbone with the recursive fine-FPN. For crowd counting in dense prediction tasks, the swin transformer has a pyramid structure similar to a CNN which can extract multi-level feature representations of images. The self-attention mechanism of transformer solves the disadvantages of local feature extraction in CNNs and can capture stronger global features. In addition, the window attention mechanism of swin transformer is executed on image patches, which alleviates the problem of uneven distribution of the crowd to a certain extent. Recursive fine-FPN iteratively fuses multi-level features to observe multiple views of the image, and produces richer feature representations for the regression head.

3.3.2 Transformer backbone network

The transformer backbone network inputs $Z \in \mathbb{R}^{N\times T}$ to the transformer encoder and uses a multi-headed attention mechanism to extract features (since the visual task does not require the encoding part, only the decoding part is utilized). The transformer encoder consists of multi-head self-attention (MSA) and MLP layers, while each layer uses residual connections and layer normalization (LN). The overall process is given by

$$Z'_{l-1} = \mathrm{MSA}\left(\mathrm{LN}\left(Z_{l-1}\right)\right) + Z_{l-1} \tag{1}$$

$$Z_l = \mathrm{MLP}\left(\mathrm{LN}\left(Z'_l\right)\right) + Z'_{l-1} \tag{2}$$

where Z'_{l-1} is the output of MSA.

Self-attention is the most important contribution of the transformer. The attention mechanism can assign different weights to input information when aggregating information. Briefly, the mechanism can learn the attention between a sequence and other sequences, which is a weight matrix from an operational point of view. There are three concepts in attention: the query (Q), the key (K), and the value (V). Each sequence outputs Q, K, and V by multiplying by the W^Q, W^K, and W^V matrices where K and V exist in pairs. The attention between different sequence pairs for each subsequence Q is

$$\mathrm{Attention}\ (Q,K,V) = \mathrm{SoftMax}\left(QK^{\mathrm{T}}/\sqrt{d}\right)V \tag{3}$$

where d is the size of the query and key.

The original transformer structure also adds a positional encoding to provide location information. ViT [30] does not use the default fixed positional encoding and instead sets the positional encoding to a set of learnable 1D sequences. The position encoding used by swin transformer has two differences: the position encoding is different, and it is added to the attention matrix. Relative position information is used instead of absolute position information. The attention can be written as

$$\mathrm{Attention}\ (Q,K,V) = \mathrm{SoftMax}(QK^{\mathrm{T}}/\sqrt{d} + B)V \tag{4}$$

where B is the relative position bias matrix.

The attention mechanism in the transformer is multi-head attention, using h heads to compute attention. This allows the model to focus on different aspects of information. The input sequence $Z \in \mathbb{R}^{N\times T}$ is divided into h sequential inputs of size $Z \in \mathbb{R}^{N\times d}$, where $T = h \times d$. Finally, this module concatenates the output from the h heads and obtains the final output using a linear transformer.

The swin transformer is a pyramid structure that can handle multiple levels as well as reducing complexity. This network consists of four transformer network layers to calculate local attention, with step sizes for the four stages given by $P = [4, 8, 16, 32]$. The swin transformer combines two types of window

division method which effectively captures both local and global attention.

3.3.3 Recursive fine-FPN

As Fig. 4 shows, we add a recursive feature pyramid structure [46] after the transformer; it is inspired by the idea of looking and thinking twice before acting. This network can deliver better semantic information through feedback connections in the structure of the fine-FPN. Multi-level features are extracted and fed back to the transformer backbone layer to realize bottom–up connections for the corresponding network layer. It is important to note that our method uses a recursive feature pyramid network to fuse the features from different stages, which differs from the winner of ICCV-VisDrone [22]. This architecture can look at images twice or more, so can better observe detailed information in a dense crowd image.

The fine-FPN solves the multi-level problem in crowd counting tasks through simple network connections. The overall display is a bottom–up structure that integrates features at different scales. Each layer of fine-FPN first adjusts the number of channels using 1×1 Conv, and up-sampling features from the previous stage. Next, it performs fusion by a simple add operation, and finally the 3×3 Conv is used to eliminate blending effects. fine-FPN improves upon the original FPN: we up-sample the fused features after 3×3 Conv to give the higher-level fused features, improving robustness. A two-level recursive feature pyramid is used in this paper, defined as

$$M' = \text{fine\_FPN}(\text{SwinT}(M')) \tag{5}$$

$$M = \text{fine\_FPN}(\text{SwinT}(M' + X)) \tag{6}$$

where M' is the output of the first stage fine-FPN and M is the final output. We use the same method to combine them as for combining the fine-FPN output and the output of the swin transformer.

3.4 Multi-level dilated convolution regression head

The density of people in images varies greatly in the crowd counting task, and images contain objects at different scales. Therefore, extraction of global features is an important foundation for weakly-supervised crowd counting. We use a multi-level dilated convolution regression head to enhance the receptive field of features. As Fig. 3(b) shows, the multi-level dilated convolution regression head (M-DRH) module consists of multi-level dilated convolution and multi-headed linear regression layers. Dilated convolution is commonly used in computer vision to collect contextual information without adding extra parameters, while widening the receptive field. The dilation rate represents the interval used in the convolution kernel. When the rate is equal to 1, the result is the same as ordinary convolution.

Using the output of multi-level features from RST, the M-DRH module performs mutli-level dilated convolution for various features. The dilation rate is inversely proportional to feature level: [2, 3, 4]. At the same time, the M-DRH module avoids the problem of local information loss resulting from dilated convolution; the swin transformer has four stages which can realize down-sampling to extract multi-level features as in CNN-based crowd counting methods. The down-sampling rate of each stage is 2, so the elements are selected at a row- and

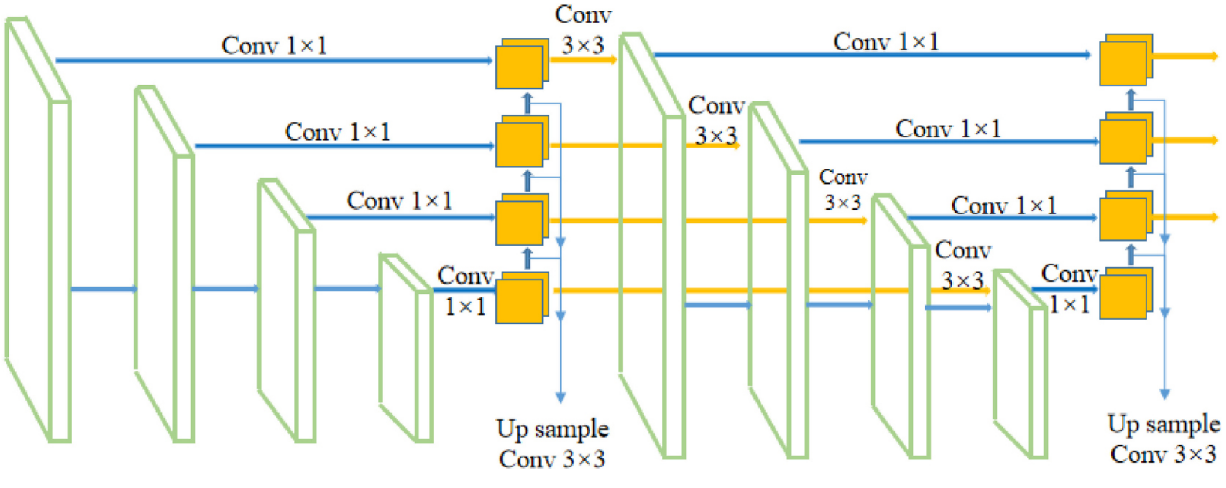


Fig. 4 Recursive fine-FPN network architecture.

column-wise interval of 2. See Fig. 5: K' is the next stage of output to K, and green shows where convolutional operations are performed while white means that no convolution is performed. Down-sampling for the swin transformer merges features near four points. So, K'_{12} is composed of $K_{(1-2)(3-4)}$ and K'_{13} is composed of $K_{(1-2)(5-6)}$. We perform dilated convolution with dilated rate of 2 and 3 for K' and K, respectively. K'_{12} does not pass the operation of convolution and $K_{(1-2)(3-4)}$ has convolution operation. Although $K_{(1-2)(5-6)}$ pixel blocks are not the operation of convolution, but K'_{13} has convolution operation, and so on for the other parts. This mechanism ensures that our proposed M-DRH is able to avoid the problem of local feature loss by dilated convolution.

Since crowd images contain head information at multiple scales, different levels of crowd head feature maps have different advantages. Therefore, we use multi-level feature maps to regress results and a dynamic layer to learn optimal fusion parameters. Specifically, an activation function and linear layer overlay component are designed to perform regression on multi-level features simultaneously. We use two kinds of fusion mechanism. In Fig. 6(left), we add a dynamic layer of parameters to learn fusion weights for the three regression results; this layer contains three learnable parameters. In Fig. 6(right), we directly average the three regression results to get the final result.

3.5 Loss function

The number of people in dense scenes can be relatively large. However, the L1 loss function commonly used in related studies has fold points which can lead to

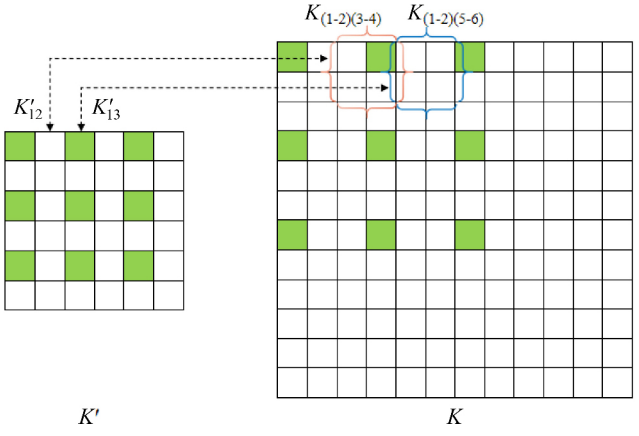


Fig. 5 Dilated convolution for different levels of image patches.

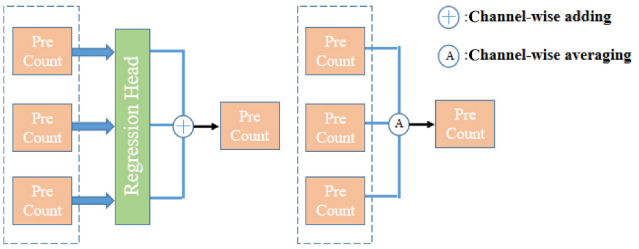


Fig. 6 Two types of regression head for DTCC.

instability in the case of large counts. In this paper, the SmoothL1 [33] loss function is used to ensure smooth output and enhance robustness; it is less likely to cause gradient explosion. It is given by

$$\text{SmoothL1}(p, D) = \begin{cases} 0.5x^2, & |x| < 1 \\ |x| - 0.5, & \text{otherwise} \end{cases} \tag{7}$$

where $x = p - D$ represents the difference between the predicted result p and the ground truth D.

The feature extraction backbone network outputs multi-level feature representations which solves the problem of target scale change in images. Therefore, two regression head fusion mechanisms are used in this paper. DTCC-Dynamic uses a dynamic network layer to automatically learn different fusion weights, with loss function:

$$\text{L}_\text{D} = \text{SmoothL1}((ae_1 + be_2 + ce_3), D) \tag{8}$$

where the multi-level regression values are e_1, e_2, e_3, and the output of the dynamic layer is defined as a, b, c. DTCC-Mean takes the mean of the predicted values, so the loss function of DTCC-Mean is

$$\text{L}_\text{M} = \text{SmoothL1}\left(\frac{e_1 + e_2 + e_3}{3}, D\right) \tag{9}$$

4 Experiments

4.1 Overview

In this section, we evaluate DTCC using several public crowd counting datasets: ShanghaiTech, UCF_CC_50, UCF_QNRF, and JHU-Crowd++. We compare our results to those of both weakly-supervised and fully-supervised methods in Tables 1–3. In addition, results of ablation experiments conducted to evaluate each component of the proposed framework are shown in Tables 4–6.

4.2 Experimental setting

4.2.1 Datasets

We used the following datasets:

Table 1 Comparison, in terms of MAE and MSE, of the proposed method to other popular methods on UCF_CC_50, ShangHaiA, ShangHaiB, UCF_QNRF

Method	Venue	Label		UCF_CC_50		ShanghaiA		ShanghaiB		UCF_QNRF	
		Location	Number	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE
MCNN [35]	CVPR16	✓	✓	277.0	426.0	110.2	173.2	26.4	41.3	277.0	426.0
CMTL [43]	AVSS17	✓	✓	322.8	397.9	101.3	152.4	20.0	31.1	252.0	514.0
Switching CNN [17]	CVPR17	✓	✓	318.1	439.2	90.4	135.4	21.6	33.4	228.0	445.0
CP-CNN [44]	ICCV17	✓	✓	298.8	320.9	73.6	106.4	20.1	30.1	—	—
ACSCP [45]	CVPR18	✓	✓	291.0	404.6	75.7	102.7	17.2	27.4	—	—
CSRNet [20]	CVPR18	✓	✓	266.1	397.5	68.2	115.0	10.6	16.0	—	—
CAN [37]	CVPR19	✓	✓	212.2	243.7	62.3	100.0	7.8	12.2	107	183
PACNN [39]	CVPR19	✓	✓	267.9	357.8	66.3	106.4	8.9	13.5	—	—
S-DCNet [40]	ICCV19	✓	✓	204.2	301.3	58.3	95.0	6.7	10.7	104.4	176.1
BL [21]	ICCV19	✓	✓	229.3	308.2	62.8	101.8	7.7	12.7	88.7	154.8
RPNet [47]	CVPR20	✓	✓	—	—	61.2	96.9	8.1	11.6	—	—
ADSCNet [38]	CVPR20	✓	✓	—	—	55.4	97.7	6.4	11.3	71.3	132.5
GL [48]	CVPR21	✓	✓	—	—	61.3	95.4	7.3	11.7	—	—
P2PNet [41]	ICCV21	✓	✓	172.7	256.2	52.7	85.1	6.3	9.9	85.3	154.5
SASNet [54]	AAAI21	✓	✓	161.4	234.5	53.5	88.3	6.3	9.9	85.2	147.3
CCTrans [31]	arXiv21	✓	✓	168.7	234.5	52.3	84.9	6.2	9.9	82.8	142.3
MATT [29]	PR21	✗	✓	355.0	550.2	80.1	129.4	11.7	17.5	—	—
TransCrowd [12]	SCIS22	✗	✓	—	—	66.1	105.1	9.3	16.1	97.2	168.5
CCTrans [31]	arXiv21	✗	✓	**245.0**	**343.6**	**64.4**	**95.4**	**7.0**	**11.5**	**92.1**	**158.9**
DTCC-Dynamic (ours*)	—	✗	✓	211.1	319.9	60.8	97.0	7.2	10.8	88.7	162.4
DTCC-Mean (ours*)	—	✗	✓	182.9	312.6	64.8	100.0	8.3	12.2	93.2	168.9

Table 2 Results on the JHU-Crowd++ validation set. Low, Medium, and High refer to images with up to 50, 50–500, and over 500 people, respectively

Method	Venue	Label		JHU-Low		JHU-Medium		JHU-High		JHU-Total	
		Location	Number	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE
MCNN [35]	CVPR16	✓	✓	90.6	202.9	125.3	259.5	494.9	856.0	160.6	377.7
CMTL [43]	AVSS17	✓	✓	50.2	129.2	88.1	170.7	583.1	986.5	138.1	379.5
DSSI-Net [49]	ICCV19	✓	✓	50.3	85.9	82.4	164.5	436.6	814.0	116.6	317.4
CAN [37]	CVPR19	✓	✓	34.2	69.5	65.6	115.3	336.4	619.7	89.5	239.3
SANet [50]	ECCV18	✓	✓	13.6	26.8	50.4	78.0	397.8	749.2	82.1	272.6
CSRNet [46]	CVPR18	✓	✓	22.2	40.0	49.0	99.5	302.5	669.5	72.2	249.9
CG-DRCN [36]	PAMI20	✓	✓	17.1	44.7	40.8	71.2	317.4	719.8	67.9	262.1
MBTTBF [55]	ICCV19	✓	✓	23.3	48.5	53.2	119.9	294.5	674.5	73.8	256.8
SFCN [11]	CVPR19	✓	✓	11.8	19.8	39.3	73.4	297.3	679.4	62.9	247.5
BL [21]	ICCV19	✓	✓	6.9	10.3	39.7	85.2	279.8	620.4	59.3	229.2
TransCrowd-Token [12]	SCIS22	✗	✓	7.1	10.7	**33.3**	**54.6**	302.5	557.4	58.4	201.1
TransCrowd-GAP [12]	SCIS22	✗	✓	**6.7**	**9.5**	34.5	55.8	**285.9**	**532.8**	**56.8**	**193.6**
DTCC-Dynamic (our*)	—	✗	✓	4.8	7.0	28.6	44.9	261.2	546.4	51.6	204.1
DTCC-Mean (our*)	—	✗	✓	4.6	6.8	29.3	44.7	266.5	566.1	54.0	187.8

(1) UCF_CC_50 [34] consists of 50 images in total, divided into training and validation sets in a ratio of 4:1. The dataset contains a small number of images and high density variation, with a maximum of 4633 people and a minimum of 96 people, with an average count of 1297.

(2) ShanghaiTechA [35] consists of 482 images in total, with 300 training images and 182 validation images. The images were randomly crawled from the Internet, so the images have a very wide range of sources. The images contain an average of 501 and a range of 33–3139 people.

Table 3 Results on the JHU-Crowd++ test set. Low, Medium, and High refer to images with up to 50, 50–500, and over 500 people, respectively

Method	Venue	Label		JHU-Low		JHU-Medium		JHU-High		JHU-Total	
		Location	Number	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE
MCNN [35]	CVPR16	✓	✓	97.1	192.3	121.4	191.3	618.6	1166.7	188.9	483.4
CMTL [43]	AVSS17	✓	✓	58.5	136.4	81.7	144.7	635.3	1225.3	157.8	490.4
DSSI-Net [49]	ICCV19	✓	✓	53.6	112.8	70.3	108.6	525.5	1047.4	133.5	416.5
CAN [37]	CVPR19	✓	✓	37.6	78.8	56.4	86.2	384.2	789.0	100.1	314.0
SANet [50]	ECCV18	✓	✓	17.3	37.9	46.8	69.1	397.9	817.7	91.1	320.4
CSRNet [46]	CVPR18	✓	✓	22.2	40.0	49.0	99.5	302.5	669.5	72.2	249.9
CG-DRCN [36]	PAMI20	✓	✓	19.5	58.7	38.4	62.7	367.3	837.5	82.3	328.0
MBTTBF [55]	ICCV19	✓	✓	19.2	58.8	41.6	66.0	352.2	760.4	81.8	299.1
SFCN [11]	CVPR19	✓	✓	16.5	55.7	38.1	59.8	341.8	758.8	77.5	297.6
BL [21]	ICCV19	✓	✓	10.1	32.7	34.2	54.5	352.0	768.7	75.0	299.9
TransCrowd-Token [12]	SCIS22	✗	✓	8.5	23.2	**33.3**	**71.5**	368.3	816.4	76.4	319.8
TransCrowd-GAP [12]	SCIS22	✗	✓	**7.6**	**16.7**	34.8	73.6	**354.8**	**752.8**	**74.9**	**295.6**
DTCC-Dynamic (our*)	—	✗	✓	7.7	19.1	30.9	50.8	296.0	652.3	64.1	254.7
DTCC-Mean (our*)	—	✗	✓	8.4	17.3	33.7	55.9	302.1	651.8	66.9	255.1

Table 4 Numbers of parameters and GFlops used by DTCC and various mainstream crowd counting methods

	DTCC	DCST	SASNet	TransCrowd
Venue	—	CVPR21	AAAI21	SCIS22
GFlops	218.2	154.8	130.9	49.3
Params	205.1	252.3	38.9	89.1

Table 5 Multi-level dilated convolution regression head ablation experiment

Method	ShanghaiA		ShanghaiB	
	MAE	MSE	MAE	MSE
DTCC (w/o M-DRH)	63.8	103.5	8.9	13.3
DTCC	**60.8**	**97.0**	**7.2**	**10.8**

Table 6 Experiment on choice of dilation rates

Dilation rates	ShanghaiA		ShanghaiB	
	MAE	MSE	MAE	MSE
1, 2, 3	64.9	101.0	7.7	11.5
2, 3, 4	**60.8**	**97.0**	**7.2**	**10.8**
3, 4, 5	62.5	97.2	7.7	11.7

(3) ShanghaiTechB [35] consists of 716 images in total, with 400 training images and 316 validation images. These are real images from the streets of Shanghai, captured by road cameras. The images contain an average of 123 and a range of 9–578 people.

(4) JHU-Crowd++ [36] consists of 4822 images in total, with 2722 training images, a validation set of 500 images, and a test set of 1600 images. This dataset has rich image information including count, person center coordinates, head frame coordinates, weather information, and lighting conditions. It can also be divided into three datasets according to the number of people contained: JHU-Low, JHU-Medium, and JHU-High. The images contain an average of 437 and a range of 2–7286 people.

(5) UCF_QNRF [51] consists of 1535 images in total, with 1201 training images and a validation set of 334 images. It contains real scenes from around the world, including buildings, vegetation, sky, and roads, which are important for counting crowds in different situations. The images contain an average of 815 and a range of 49–12,865 people.

4.2.2 Baselines and compared methods

In order to verify the effectiveness of our method, we choose a large number of comparator methods including mainstream fully-supervised and state-of-the-art weakly-supervised methods. Fully-supervised methods need both person location and count annotations, and include CAN [37], ADSCNet [38], PACNN [39], S-DCNet [40], and P2PNet [41]. Weakly-supervised methods only need count annotations, and include that of MATT, TransCrowd, and CCTrans. In particular, TransCrowd and CCTrans also use a transformer as the backbone network for feature extraction.

4.2.3 Implementation details

We used the Swin-L model pre-trained on ImageNet-22K to speed up convergence of the model. For the backbone network of the swin transformer, the number of heads used was [6, 12, 24, 48], the position embedding was a position bias matrix, the window size was 12, the number of layers was [6, 12, 24, 48], and the number of channels in the hidden layer of the first stage was 192. In the training section, we strictly followed the input image size requirement of 384×384 for Swin-L. We used the same approach as TransCrowd [12]: we resize all original images to 1152×768 (landscape) or 768×1152 (portrait), then cropped each image into 6 blocks of size 384×384, and calculated the number of people in each image block by location annotation of the people in the image. We also utilized data augmentation strategies, such as random flipping and gray scaling. For compatibility with the operation of dividing each image into 6 image blocks, the training batch size was set to 24. All experiments were executed on a Linux system with an Intel E5-2620v4 Xeon CPU at 2.10 GHz and an NVidia P100 16 GB Tesla. The learning rate was set to 10^{-5} initially and decreased to 10^{-6} in the final epoch.

In the evaluation phase, we choose the widely accepted MSE and MAE as metrics:

$$\mathrm{MAE} = \frac{1}{N}\sum_{i=1}^{N}|P_i - G_i| \tag{10}$$

$$\mathrm{MSE} = \sqrt{\frac{1}{N}\sum_{i=1}^{N}|P_i - G_i|^2} \tag{11}$$

where N is the number of images, and P_i and G_i represent the i-th predicted count and ground truth, respectively. The MAE represents the mean absolute error, and is a very intuitive evaluation metric representing the distance between predicted value and ground truth. MSE better represents the stability of the model.

4.3 Comparison to existing methods

Our method, with alternatives DTCC-Dynamic and DTCC-Mean, shows good accuracy compared to other weakly-supervised methods. Table 1 gives errors for the UCF_CC_50, ShanghaiTech, and UCF_QNRF datasets. UCF_CC_50 has only a few images, and they are all of dense crowds. Without increasing the amount of data, our method has made significant progress compared to other weakly-supervised methods. DTCC-Mean achieves better results than DTCC-Dynamic, showing that using an average fusion mechanism for the regression head can provide good accuracy and robustness given a small amount of data. The ShanghaiTech partA dataset comes from a wide range of scenes with large variations in crowd density, so accurately estimating the number of people is very challenging. Our proposed method DTCC-Dynamic achieves the best MAE. This indicates that our backbone swin transformer network can better adapt to different densities, while the dynamic fusion regression head can learn optimal ratios from a large number of datasets. However, the MSE metrics is less satisfactory. So, in the presence of anomalies, our method still needs to be improved. On ShanghaiTech partB, DTCC again achieves significant improvements over other weakly-supervised methods.

For the JHU-Crowd++ dataset, we conducted experiments on the validation set (Table 2) and test set (Table 3) separately. We divided the dataset into three count levels of low (0–50), medium (51–500), high (500+), and also aggregated total results. For the validation set, DTCC achieves the better results than other weakly-supervised methods, and shows competitive results when compared to mainstream fully-supervised methods. To further demonstrate the effectiveness of the proposed DTCC, we conducted further experiments on the test set using the pre-trained model parameters of JHU-Total. In this case, for the JHU-Low dataset, our proposed method achieves competitive results, differing slightly from the state-of-the-art weakly-supervised method. On JHU-Medium, our method achieves the best estimates compared to other weakly-supervised methods; it shows a strong advantage for this higher density dataset. On JHU-High further improvements are seen. The proposed dynamic fusion mechanism achieves better results: dynamic learning parameters provide good fault tolerance for ultra-dense scenes. JHU-Total contains all horizontal images, and the density range of the dataset is large, which requires a robust model. The good improvements to MAE and MSE, show that our method has not only high accuracy but also good stability.

4.4 Visualization of feature maps

To verify the effectiveness of our method, we visualized feature maps on ShanghaiTech PartA using

heat maps. As noted, each image is split into six sub-images to input into the model, which can be seen in the visualization. Figure 7 shows that our method pays more attention to dense image regions and adapts to different scenes. It can also be seen that there are a few areas incorrectly given attention due to the lack of individual person annotations.

4.5 Computational cost

To evaluate the all-round performance of DTCC, we calculated the number of parameters and GFlops consumed by the model. Table 4 compared the results with other mainstream full-supervised and weakly-supervised crowd counting methods. DTCC consumes more computing resources than other methods, because it uses the swin transformer as its backbone network. Using a recursive swin transformer further increases the cost. However, the cost of our work is still of the same order of magnitude as that of other works, with no major impact on practical applications.

4.6 Ablation experiments and tuning

We conducted various ablation experiments on the DTCC-Dynamic version using the ShanghaiTech dataset to verify the contribution of each module and to justify the reasoning behind it, and to tune operation.

4.6.1 Multi-level dilated convolution regression head

Table 5 shows results of an ablation experiment on use of the multi-level dilated convolution regression module. By comparing the results with and without M-DRH, we can see that introducing the multi-level dilated convolutional regression head improves counting accuracy. This justifies our assumption that multi-level feature relationship modeling can capture different scales of crowd information in images with dense crowd scenes. In addition, the presence of the dilated convolution can enhance the global receptive field, which is important for weakly-supervised methods in crowd counting.

We conducted further experiments to assess different choices of dilation rate. As Table 6 shows, optimal results were obtained by setting the dilation rates to [2, 3, 4]. Using too small dilation rates does not enhance the receptive field of features enough, while using too large dilation rates may lead to loss of local features. As a compromise, we set the dilation rates to [2, 3, 4].

We also performed experiments with different multi-level features for weakly-supervised methods, as reported in Table 7. Dilation rates 1, 2, 3 represent feature maps with resolutions of 12×12, 24×24, and 48×48, respectively. Adding successive feature maps of different resolutions improves the model's results significantly, demonstrating that multi-level features are important to our method.

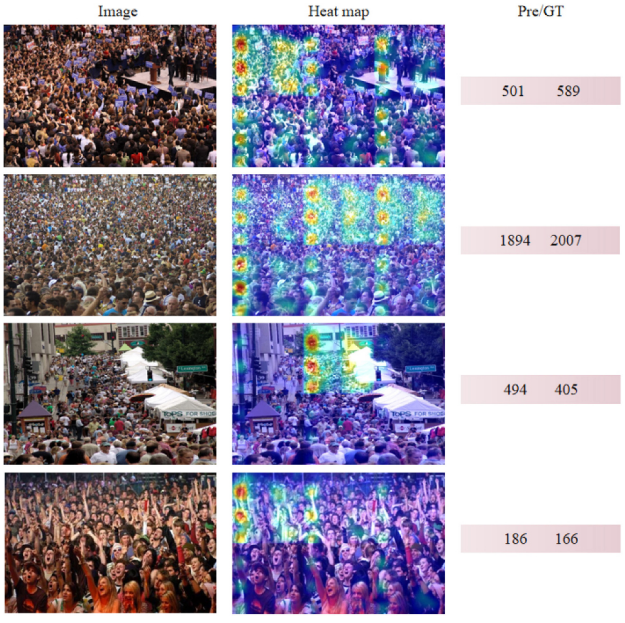


Fig. 7 Visualization of feature maps.

4.6.2 Recursive fine-FPN

In Table 8, a baseline of DTCC-Dynamic was used; it compares results of using the baseline with recursive FPN, and using the baseline with recursive fine-FPN. We can see that using fine-FPN achieves better results than the original FPN. This indicates that for crowd counting, fusion of deep features upsampled by 3×3 Conv can provide better performance.

Table 7 Experiment on use of multi-level features

Dilation rates	ShanghaiA		ShanghaiB	
	MAE	MSE	MAE	MSE
1	63.8	97.2	8.1	13.0
1, 2	62.7	96.4	7.9	12.1
1, 2, 3	**60.8**	**97.0**	**7.2**	**10.8**

Table 8 Ablation experiment on recursive fine-FPN

Module	ShanghaiA		ShanghaiB	
	MAE	MSE	MAE	MSE
Recursive FPN	62.1	99.0	8.0	12.5
Recursive fine-FPN	**60.8**	**97.0**	**7.2**	**10.8**

We performed further experiments on the pyramid structure. The baseline was DTCC-Dynamic method without any pyramid structures. Table 9 compares results of three sets of experiments, using baseline, baseline with fine-FPN, and baseline with recursive fine-FPN. Addition of the pyramid structure effectively improves the accuracy of the model; the recursive pyramid structure achieves the best accuracy. Due to the features at the last level of the transformer output, pixel information is easily lost in the process of increasing the step size for patches. Using the recursive fine-FPN causes extra feedback connections from fine-FPN to be incorporated into the bottom–up backbone layers, allowing all levels of feature maps to have strong contextual information.

4.6.3 Loss function

We separately evaluated the commonly used L1 and SmoothL1 [33] loss functions. From the results in Table 10, it can be concluded that SmoothL1 gives better results. SmoothL1 is more stable and can adapt well to both large and small errors.

Table 9 Experiments on feature pyramid

Module	Shanghai-A		Shanghai-B	
	MAE	MSE	MAE	MSE
DTCC w/o fine-FPN	68.1	118.6	9.6	16.9
fine-FPN	64.5	103.1	8.3	13.1
Recursive fine-FPN	**60.8**	**97.0**	**7.2**	**10.8**

Table 10 Choice of loss function

Method	ShanghaiA		ShanghaiB	
	MAE	MSE	MAE	MSE
L1	62.1	99.0	8.0	12.5
SmoothL1	**60.8**	**97.0**	**7.2**	**10.8**

5 Conclusions

This work proposes a pyramidal vision transformer network for weakly-supervised crowd counting; it can achieve end-to-end crowd counting. A multi-level feature extraction module and a multi-level dilated convolutional regression module are designed for dense prediction tasks; they can better capture global features and generate more reasonable features for weakly-supervised crowd counting. Extensive experiments on four well-known benchmark datasets demonstrate that DTCC achieves superior counting performance compared to other mainstream weakly-supervised methods and is competitive with some fully-supervised methods.

In future, we plan to further investigate a more concise feature extraction backbone network for crowd counting, and design a better regression head for prediction. In addition, we also intend to further extend DTCC to other dense prediction scenarios, such as traffic counting for intelligent transportation.

Acknowledgements

This research project was partially supported by the National Natural Science Foundation of China (Grant Nos. 62072015, U19B2039, U1811463), and the National Key R&D Program of China (Grant No. 2018YFB1600903).

A portion of the work in this paper was carried out using the Taiji machine learning engine, and we thank Taiji for their support.

Declaration of competing interest

The authors have no competing interests to declare that are relevant to the content of this article.

References

[1] Li, M.; Zhang, Z. X.; Huang, K. Q.; Tan, T. N. Estimating the number of people in crowded scenes by MID based foreground segmentation and head-shoulder detection. In: Proceedings of the 19th International Conference on Pattern Recognition, 1–4, 2008.

[2] Wu, B.; Nevatia, R. Detection and tracking of multiple, partially occluded humans by Bayesian combination of edgelet based part detectors. *International Journal of Computer Vision* Vol. 75, No. 2, 247–266, 2007.

[3] Lempitsky, V. S.; Zisserman, A. Learning to count objects in images. In: Proceedings of the 23rd International Conference on Neural Information Processing Systems, Vol. 1, 1324–1332, 2010.

[4] Walach, E.; Wolf, L. Learning to count with CNN boosting. In: *Computer Vision – ECCV 2016. Lecture Notes in Computer Science, Vol. 9906.* Leibe, B.; Matas, J.; Sebe, N.; Welling, M. Eds. Springer Cham, 660–676, 2016.

[5] Wang, C.; Zhang, H.; Yang, L.; Liu, S.; Cao, X. C. Deep people counting in extremely dense crowds. In: Proceedings of the 23rd ACM International Conference on Multimedia, 1299–1302, 2015.

[6] Fu, M.; Xu, P.; Li, X. D.; Liu, Q. H.; Ye, M.; Zhu, C. Fast crowd density estimation with convolutional neural networks. *Engineering Applications of Artificial Intelligence* Vol. 43, 81–88, 2015.

[7] Song, Q. Y.; Wang, C. G.; Jiang, Z. K.; Wang, Y. B.; Tai, Y.; Wang, C. J.; Li, J. L.; Huang, F. Y.; Wu, Y. Rethinking counting and localization in crowds: A purely point-based framework. In: Proceedings of the IEEE/CVF International Conference on Computer Vision, 3345–3354, 2021.

[8] Meng, Y. D.; Zhang, H. R.; Zhao, Y. T.; Yang, X. Y.; Qian, X. S.; Huang, X. W.; Zheng, Y. Spatial uncertainty-aware semi-supervised crowd counting. In: Proceedings of the IEEE/CVF International Conference on Computer Vision, 15529–15539, 2021.

[9] Wan, J.; Liu, Z. Q.; Chan, A. B. A generalized loss function for crowd counting and localization. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 1974–1983, 2021.

[10] Liu, X. L.; van de Weijer, J.; Bagdanov, A. D. Exploiting unlabeled data in CNNs by self-supervised learning to rank. *IEEE Transactions on Pattern Analysis and Machine Intelligence* Vol. 41, No. 8, 1862–1878, 2019.

[11] Wang, Q.; Gao, J. Y.; Lin, W.; Yuan, Y. Learning from synthetic data for crowd counting in the wild. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 8190–8199, 2019.

[12] Liang, D. K.; Chen, X. W.; Xu, W.; Zhou, Y.; Bai, X. TransCrowd: Weakly-supervised crowd counting with transformers. *Science China Information Sciences* Vol. 65, No. 6, Article No. 160104, 2022.

[13] Liu, Z.; Lin, Y. T.; Cao, Y.; Hu, H.; Wei, Y. X.; Zhang, Z.; Lin, S.; Guo, B. Swin transformer: Hierarchical vision transformer using shifted windows. In: Proceedings of the IEEE/CVF International Conference on Computer Vision, 9992–10002, 2021.

[14] Chen, C. F R.; Fan, Q. F.; Panda, R. CrossViT: Cross-attention multi-scale vision transformer for image classification. In: Proceedings of the IEEE/CVF International Conference on Computer Vision, 347–356, 2021.

[15] Huang, Z.; Ben, Y.; Luo, G.; Cheng, P.; Yu, G.; Fu, B. Shuffle transformer: Rethinking spatial shuffle for vision transformer. *arXiv preprint* arXiv:2106.03650, 2021.

[16] Carion, N.; Massa, F.; Synnaeve, G.; Usunier, N.; Kirillov, A.; Zagoruyko, S. End-to-end object detection with transformers. In: *Computer Vision – ECCV 2020. Lecture Notes in Computer Science, Vol. 12346.* Vedaldi, A.; Bischof, H.; Brox, T.; Frahm, J. M. Eds. Springer Cham, 213–229, 2020.

[17] He, L.; Zhou, Q. Y.; Li, X. T.; Niu, L.; Cheng, G. L.; Li, X.; Liu, W.; Tong, Y.; Ma, L.; Zhang, L. End-to-end video object detection with spatial-temporal transformers. In: Proceedings of the 29th ACM International Conference on Multimedia, 1507–1516, 2021.

[18] Zhang, Y. Y.; Zhou, D. S.; Chen, S. Q.; Gao, S. H.; Ma, Y. Single-image crowd counting via multi-column convolutional neural network. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 589–597, 2016.

[19] Sam, D. B.; Surya, S.; Babu, R. V. Switching convolutional neural network for crowd counting. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 4031–4039, 2017.

[20] Li, Y. H.; Zhang, X. F.; Chen, D. M. CSRNet: Dilated convolutional neural networks for understanding the highly congested scenes. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 1091–1100, 2018.

[21] Ma, Z. H.; Wei, X.; Hong, X. P.; Gong, Y. H. Bayesian loss for crowd count estimation with point supervision. In: Proceedings of the IEEE/CVF International Conference on Computer Vision, 6141–6150, 2019.

[22] Liu, Z.; He, Z.; Wang, L.; Wang, W.; Yuan, Y.; Zhang, D.; Zhang, J.; Zhu, P.; Van Gool, L.; Han, J.; et al. VisDrone-CC2021: The vision meets drone crowd counting challenge results. In: Proceedings of the IEEE/CVF International Conference on Computer Vision Workshops, 2830–2838, 2021.

[23] Liang, D.; Xu, W.; Bai, X. An end-to-end transformer model for crowd localization. *arXiv preprint* arXiv:2202.13065, 2022.

[24] Abousamra, S.; Hoai, M.; Samaras, D.; Chen, C. Localization in the crowd with topological constraints. *Proceedings of the AAAI Conference on Artificial Intelligence* Vol. 35, No. 2, 872–881, 2021.

[25] Sun, G. L.; Liu, Y.; Probst, T.; Paudel, D. P.; Popovic, N.; Van Gool, L. Boosting crowd counting with transformers. *arXiv preprint* arXiv:2105.10926, 2021.

[26] Gao, J. Y.; Gong, M. G.; Li, X. L. Congested crowd instance localization with dilated convolutional swin transformer. *arXiv preprint* arXiv:2108.00584, 2021.

[27] Shang, C.; Ai, H. Z.; Bai, B. End-to-end crowd counting via joint learning local and global count. In: Proceedings of the IEEE International Conference on Image Processing, 1215–1219, 2016.

[28] Wang, M. J.; Zhou, J.; Cai, H.; Gong, M. L. CrowdMLP: Weakly-supervised crowd counting via multi-granularity MLP. *arXiv preprint* arXiv: 2203.08219, 2022.

[29] Lei, Y. J.; Liu, Y.; Zhang, P. P.; Liu, L. Q. Towards using count-level weak supervision for crowd counting. *Pattern Recognition* Vol. 109, 107616, 2021.

[30] Dosovitskiy, A.; Beyer, L.; Kolesnikov, A.; Weissenborn, D.; Zhai, X. H.; Unterthiner, T.; Dehghani, M.; Minderer, M.; Heigold, G.; Gelly, S.; et al. An image is worth 16x16 words: Transformers for image recognition at scale. In: Proceedings of the International Conference on Learning Representations, 2021.

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

[32] Chu, X.; Tian, Z.; Wang, Y.; Zhang, B.; Ren, H.; Wei, X.; Xia, H.; Shen, C. Twins: Revisiting the design of spatial attention in vision transformers. In: Proceedings of the Advances in Neural Information Processing Systems, Vol. 34, 9355–9366, 2021.

[33] Girshick, R. Fast R-CNN. In: Proceedings of the IEEE International Conference on Computer Vision, 1440–1448, 2015.

[34] Idrees, H.; Saleemi, I.; Seibert, C.; Shah, M. Multi-source multi-scale counting in extremely dense crowd images. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2547–2554, 2013.

[35] Zhang, Y. Y.; Zhou, D. S.; Chen, S. Q.; Gao, S. H.; Ma, Y. Single-image crowd counting via multi-column convolutional neural network. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 589–597, 2016.

[36] Sindagi, V. A.; Yasarla, R.; Patel, V. M. JHU-CROWD: Large-scale crowd counting dataset and a benchmark method. *IEEE Transactions on Pattern Analysis and Machine Intelligence* Vol. 44, No. 5, 2594–2609, 2022.

[37] Liu, W. Z.; Salzmann, M.; Fua, P. Context-aware crowd counting. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 5094–5103, 2020.

[38] Bai, S.; He, Z. Q.; Qiao, Y.; Hu, H. Z.; Wu, W.; Yan, J. J. Adaptive dilated network with self-correction supervision for counting. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 4593–4602, 2020.

[39] Shi, M. J.; Yang, Z. H.; Xu, C.; Chen, Q. J. Revisiting perspective information for efficient crowd counting. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 7271–7280, 2019.

[40] Xiong, H. P.; Lu, H.; Liu, C. X.; Liu, L.; Cao, Z. G.; Shen, C. H. From open set to closed set: Counting objects by spatial divide-and-conquer. In: Proceedings of the IEEE/CVF International Conference on Computer Vision, 8361–8370, 2019.

[41] Song, Q.; Wang, C.; Jiang, Z.; Wang, Y.; Tai, Y.; Wang, C.; Li, J.; Huang, F.; Wu, Y. Rethinking counting and localization in crowds: A purely point-based framework. In: Proceedings of the IEEE/CVF International Conference on Computer Vision, 3345–3354, 2021.

[42] Yang, Y.; Li, G.; Wu, Z.; Su, L.; Huang, Q.; Sebe, N. Weakly-supervised crowd counting learns from sorting rather than locations. In: *Computer Vision – ECCV 2020. Lecture Notes in Computer Science, Vol. 12353*. Vedaldi, A.; Bischof, H.; Brox, T.; Frahm, J. M. Eds. Springer Cham, 1–17, 2020.

[43] Sindagi, V. A.; Patel, V. M. CNN-based cascaded multi-task learning of high-level prior and density estimation for crowd counting. In: Proceedings of the 14th IEEE International Conference on Advanced Video and Signal Based Surveillance, 1–6, 2017.

[44] Sindagi, V. A.; Patel, V. M. Generating high-quality crowd density maps using contextual pyramid CNNs. In: Proceedings of the IEEE International Conference on Computer Vision, 1879–1888, 2017.

[45] Shen, Z.; Xu, Y.; Ni, B. B.; Wang, M. S.; Hu, J. G.; Yang, X. K. Crowd counting via adversarial cross-scale consistency pursuit. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 5245–5254, 2018.

[46] Qiao, S. Y.; Chen, L. C.; Yuille, A. DetectoRS: Detecting objects with recursive feature pyramid and switchable atrous convolution. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 10208–10219, 2021.

[47] Yang, Y. F.; Li, G. R.; Wu, Z.; Su, L.; Huang, Q. M.; Sebe, N. Reverse perspective network for perspective-aware object counting. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 4373–4382, 2020.

[48] Wan, J.; Liu, Z. Q.; Chan, A. B. A generalized loss function for crowd counting and localization. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 1974–1983, 2021.

[49] Liu, L. B.; Qiu, Z. L.; Li, G. B.; Liu, S. F.; Ouyang, W. L.; Lin, L. Crowd counting with deep structured scale integration network. In: Proceedings of the IEEE/CVF

International Conference on Computer Vision, 1774–1783, 2019.

[50] Cao, X.; Wang, Z.; Zhao, Y.; Su, F. Scale aggregation network for accurate and efficient crowd counting. In: *Computer Vision – ECCV 2018. Lecture Notes in Computer Science, Vol. 11209*. Ferrari, V.; Hebert, M.; Sminchisescu, C.; Weiss, Y. Eds. Springer Cham, 757–773, 2018.

[51] Idrees, H.; Tayyab, M.; Athrey, K.; Zhang, D.; Al-Maadeed, S.; Rajpoot, N.; Shah, M. Composition loss for counting, density map estimation and localization in dense crowds. In: *Computer Vision – ECCV 2018. Lecture Notes in Computer Science, Vol. 11206*. Ferrari, V.; Hebert, M.; Sminchisescu, C.; Weiss, Y. Eds. Springer Cham, 544–559, 2018.

[52] Savner, S. S.; Kanhangad, V. CrowdFormer: Weakly-supervised crowd counting with improved generalizability. *arXiv preprint* arXiv:2203.03768, 2022.

[53] Wang, F. S.; Liu, K.; Long, F.; Sang, N.; Xia, X. F.; Sang, J. Joint CNN and transformer network via weakly supervised learning for efficient crowd counting. *arXiv preprint* arXiv:2203.06388, 2022.

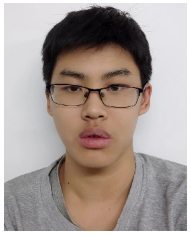
[54] Song, Q.; Wang, C.; Wang, Y.; Tai, Y.; Wang, C.; Li, J.; Wu, J.; Ma, J. To choose or to fuse? Scale selection for crowd counting. *Proceedings of the AAAI Conference on Artificial Intelligence* Vol. 35, No. 3, 2576–2583, 2021.

[55] Sindagi, V. A.; Patel, V. M. Multi-level bottom–top and top–bottom feature fusion for crowd counting. In: Proceedings of the IEEE/CVF International Conference on Computer Vision, 1002–1012, 2019.

Zhuangzhuang Miao is a master student in the Faculty of Information Technology of Beijing University of Technology (BJUT). He got his B.S. degree from Shijiazhuang University in 2020. His research interests include deep learning and computer graphics.

Yong Zhang received his Ph.D. degree in computer science from BJUT in 2010. He is currently an associate professor of computer science at BJUT. His research interests include intelligent transportation systems, big data analysis, visualization, and computer graphics.

Yuan Peng received his M.S. degree in software engineering and IT methods applied to business management from Jules Verne University of Picardy, France in 2011 and 2012, respectively. He is currently a senior engineer in China Electronics Technology Group. His current research interests include geographic information systems, air traffic control, computer graphics, atmospheric operation modes, and radar echos.



Haocheng Peng is currently studying for a bachelor degree in IoT in Beijing Dublin International College. His current research interests include deep learning and block chains.

Baocai Yin received his B.S., M.S., and Ph.D. degrees in computational mathematics from Dalian University of Technology, China, in 1985, 1988, and 1993, respectively. He is currently a professor in the Beijing Key Laboratory of Multimedia and Intelligent Software Technology, BJUT. His research interests include multimedia, image processing, computer vision, and pattern recognition.