GeoInformatica (2024) 28:245–269
https://doi.org/10.1007/s10707-023-00503-7

RESEARCH

MSCNet: Dense vehicle counting method based on multi-scale dilated convolution channel-aware deep network

Qiyan Fu[1] · Weidong Min[1,2,3] · Chunbo Li[1] · Haoyu Zhao[1] · Ye Cao[4] · Meng Zhu[1]

Received: 30 March 2023 / Revised: 27 May 2023 / Accepted: 20 June 2023 /
Published online: 8 July 2023

Abstract
Accurately counting the number of dense objects, such as crowds or vehicles, in an image is a challenging and meaningful task widely used in public safety management and traffic flow prediction. The existing CNN-based density map estimation methods are ineffective for extracting the counting features of long-distance queuing vehicles in traffic jams; In addition, these methods do not focus on counting in complex scenes, such as vehicle counting in the human-vehicle mixed scenes. To tackle this issue, we propose MSCNet, a novel multi-scale dilated convolution channel-aware deep network for vehicle counting. The proposed network solves the problem of scale variation for long-distance queuing vehicles and improves the ability to extract vehicle features in human-vehicle mixed scenes. The MSCNet consists of a front-end module and three functional modules: the front-end module is used to extract the initial features of the counting image; the direction-based perspective coding module (DPCM) encodes the perspective information of the image from four directions to extract continuous long-distance features; the multi-scale dilated residual module (MDRM) can densely extract the large-scale variation features; the channel-aware attention module (CAM) effectively enhances the channel features that are important for vehicle counting in mixed human-vehicle scenes. The MSCNet has conducted extensive comparative experiments on the TRANCOS dataset, the VisDrone2021 Vehicle&Crowd dataset, and the ShanghaiTech dataset. The experimental results show that the MSCNet outperforms the state-of-the-art counting networks for dense vehicle counting, especially in mixed human-vehicle scenes.

Keywords Dense vehicle counting · Data mining · Traffic flow prediction · Density map estimation · MSCNet

✉ Weidong Min
minweidong@ncu.edu.cn

[1] School of Mathematics and Computer Sciences, Nanchang University, Nanchang 330031, China

[2] Institute of Metaverse, Nanchang University, Nanchang 330031, China

[3] Jiangxi Key Laboratory of Smart City, Nanchang 330031, China

[4] School of Information Engineering, Nanchang University, Nanchang 330031, China

1 Introduction

Object counting is used to count the number of objects in some scenes of interest for images or videos, such as vehicle counting in traffic jam scenes [1], crowd counting in outdoor public scenes [2], and specific bacteria or cell counting in microscopic observation scenes [3]. Object counting has many applications in traffic flow prediction, public safety management, and biological and medical research. Vehicle counting is vital in urban traffic planning and Intelligent Transportation Systems (ITS) [4] management in smart cities. Studying vehicle counting involves research from multiple disciplines involving computer vision, machine learning, deep learning, and pattern recognition. Therefore, the research on vehicle counting has attracted much attention in recent years and has the potential for sustainable development [5].

Object counting approaches in images or videos have been classified into four categories: detection-based approaches [6], clustering-based approaches [7], regression-based approaches [8], and density map-based approaches [9]. The detection-based counting approaches usually rely on the accurate segmentation and detection of a single counting target, while dense objects are often heavily occluded and challenging to segment accurately. Therefore, this approach is unsuitable for dense vehicle counting in traffic jam scenarios. The clustering-based counting approaches are generally used for object counting in the video. For example, Antonini et al. [10] obtained the object trajectory through a tracking algorithm and then performed a clustering analysis on the trajectory. However, the target tracking algorithm is unsuitable for calculating dense vehicles in traffic jams in one image. The regression-based approaches usually estimate the number of objects in an image by building a regression model between regional image features and the number of counted objects in that region. For example, Liang et al. [8] used image features such as regional size, edge gradient, and image texture to derive a region regression model for counting vehicles on highways. The regression-based approaches can be used to calculate the number of vehicles in traffic jams to obtain high counting accuracy for dense objects in a relatively fixed background. However, the approaches essentially lose the spatial information of counting vehicles in the image. In 2010, Lempitsky et al. [11] innovatively proposed an object counting method based on density map regression. This method differs from the previous regression-based approaches, which focus on mining spatial information data and satisfying the dense object counting in crowded scenes. The density map-based approaches are well suited for massive dense object counting, especially for dense vehicle counting in traffic congestion scenarios. With the advent of deep learning, Convolution Neural Networks (CNN) have achieved great success in various computer vision tasks, which prompted researchers to use CNN for data mining to replace traditional feature extraction methods and learn complex functions between input images and density maps. These methods [9, 12–14], based on CNN can deal with severe occlusions between dense objects in images and generate estimated density maps more accurately and faster than traditional methods, thus obtaining an accurate number of objects.

The density map regression method using CNN can effectively deal with severe occlusions in dense vehicle counting. However, many other problems, such as perspective changes, large-scale variation, and complex backgrounds, still need to be researched and solved. Among them, the large-scale variation is the main issue because the scales of the object in the image will vary with their distance from the camera lens, and the range of scale variation is very large for long-distances queuing dense vehicles in an image. The perspective changes will also powerfully lead to the scale variation of the vehicle counting

in traffic congestion scenarios. Many multi-column networks based on CNN [14] have been proposed to handle scale variation for better counting accuracy. Although these methods have improved the counting performance of the networks, the scale diversity they captured is limited by the number of network columns, and continuous scale variation is not taken into account, so the features of continuous long-distance queuing vehicles in traffic jams cannot be well extracted. Complex background regions in counting images mainly include mixed objects with similar appearance or color to the foreground, and the problem can be generally suppressed by semantic segmentation or visual attention mechanism, such as [15]. However, most of these methods are networks with multi-task architectures designed for crowd counting, which are very complex and do not target the extraction of vehicle features in the human-vehicle mixed scenes. In summary, the problem of long-distance continuous scale variation of queuing vehicles in mixed scenes is the main problem to be solved, and this paper mainly deals with these problems through deep data mining of spatial information and channel correlation information.

We propose a novel multi-scale dilated convolution channel-aware deep network called MSCNet for vehicle counting to alleviate such problems. MSCNet is a CNN-based single-column but multi-branch deep network. The network adopts an end-to-end cascaded framework, which can effectively mine the spatial data and channel correlation information in the images, generate high-quality density maps and calculate the number of vehicles. Firstly, the counting image is sent into the first ten layers of VGG-16 [16] for the initial image feature extraction. In order to handle the problem of continuous perspective changes, the direction-based perspective coding module is employed to encode global features in four directions. This method also proposes a multi-scale dilated residual module that extracts feature information of continuously changing scales. We use an efficient and flexible cascading method to choose different dilated rates for layers within the module and use residual connection way to handle the vanishing gradient caused by the deep network [17]. The multi-scale dilated residual module can sample an extensive scale range in a much denser manner, thus handling the large-scale variation problem in dense vehicle counting. In addition, the channel-aware attention module is used to learn the weights of different channel features and enhance features of the vehicle to be extracted in mixed scenes. The module introduces global features in the channel dimension to extract the essential counting features hidden in the different channels. The experimental results based on benchmark datasets show that the proposed MSCNet for dense vehicle counting outperforms the existing methods.

In total, MSCNet mainly consists of a front-end module and three functional modules: a front-end module based on VGG-16 (VGG16), a direction-based perspective coding module (DPCM), a multi-scale dilated residual module (MDRM), and a channel-aware attention module (CAM). The main contributions of this study can be summarized as follows:

1) The DPCM encodes the perspective information from four directions, which is suitable for extracting the features of long-distance continuous queuing vehicles.
2) The MDRM is proposed to densely extract the object features at different scale variations, which obtains six sizes of receptive fields by combining three dilation rates. The DPCM and MDRM are used to deeply mine the spatial information and solve the problem of long-distance continuous scale variation for dense vehicle counting.
3) The CAM is a module with a channel-aware function. This module can deeply mine the channel correlations to enhance useful features for vehicle counting in the human-vehicle mixed scenes.

The subsequent sections of this paper are as follows: Section 2 discusses related works. Section 3 introduces our method along with the implementation of technical details. Section 4 describes the various comparative experiments and their results, and Section 5 draws conclusions and presents directions for our future research.

2 Related work

The goal of the counting task is to make the computer accurately estimate the number of related objects in the image. As discussed in Section 1, summarizing the current mainstream approaches, the object counting approaches based on computer vision can be divided into four categories: the detection-based approaches, the clustering-based approaches, the regression-based approaches, and the density map-based approaches. This section reviews related studies on vehicle counting using the density map-based approaches from traditional and CNN-based methods.

2.1 Traditional methods

In 2010, Lempitsky et al. [11] first proposed a dense object counting model framework based on density map estimation. The main problem solved by this model is to count the objects of interest in the image when the single object is crowded or occluded. The main steps are as follows: Firstly, the ground truth image set of object density distribution (also known as GT density maps later) is generated by manually annotated object center point map, and the generation of the GT density map uses the Gaussian kernel function. Then, taking the GT density maps as the training set, the image features are extracted at each pixel position of the original image, and the regression model is trained to directly learn the mapping relationship from pixel features to the GT density map. Finally, the predicted density distribution map reflects the distribution of objects in the scene, and the number of objects in any region can be obtained by integrating the regional density map. The traditional methods use manual design features, such as the gray value of the image, the color value of the image, and SIFT features [11]. The regression model of traditional methods usually adopts ridge regression and random forest regression [18]. In 2015, Pham et al. [19] proposed learning the nonlinear mapping between local features of image blocks and density maps. In 2016, Wang et al. [20] proposed a fast density estimation method based on subspace learning to solve the problem of the low computational efficiency of traditional methods. These methods only use traditional manual features to extract effective information, so they are only suitable for specific counting scenarios and cannot efficiently and flexibly guide the generation of high-quality predictive density maps to obtain more accurate object counts.

2.2 CNN-based methods

In recent years, with the rapid development of deep learning, CNN has been widely used in various tasks of computer vision. Fu et al. [12] first introduced deep learning into object counting based on the density map in 2015 and used a CNN-based model to count crowds. The model improves the speed and accuracy by deleting some parallel network connections and cascading two ConvNet classifiers. Cross-scene model [13] belongs to the basic network model, which is one of the earliest counting models using deep learning technology.

Zhang et al. completed the optimization of the Cross-scene model by alternately training the model with two tasks: density map estimation and regional crowd estimation. Inspired by the successful application of multi-column deep neural networks [21] in the field of image classification, Zhang et al. proposed a density map estimation model named Multi-Column Convolutional Neural Network (MCNN) [14] based on multi-column networks. The principle idea of the MCNN model is to use different sizes of receptive fields in each column of the network to extract the features of the counting objects at different scales in the image. Another contribution of the model is that it improves the traditional density map generation algorithm based on Gaussian filtering and proposes an adaptive Gaussian kernel method for the first time. Li et al. [9] proposed a single-column network CSRNet for crowded scenarios, which abandoned the use of multi-column network architecture. The core idea is to design a deeper model to capture higher-level features with larger receiving fields and generate high-quality density maps without using more complex multi-column networks. This network uses dilated convolutions to capture the higher-level receptive fields and replace pooling operations. In contrast, CSCNet [22] proposed a shallow single-column network, and the critical element was designed to obtain complementary scale context and achieve high counting accuracy with limited network depth. In addition to studying the improvement of various network structures, generating a high-quality GT density map and designing appropriate loss functions are also important factors that directly affect the performance of the counting model. To solve the problem of large-scale variation, Zhang et al. [13] used different Gaussian kernels for the human head and body when generating the density map. This method used the weighting of perspective normalization parameters to calculate the size of the Gaussian kernel in each position of the scene. Zhang et al. [14] found that the size of the head was related to the distance between two adjacent people and proposed an adaptive Gaussian kernel method, which inspired much subsequent research work. The method had a good effect in dense areas of the scene but was easy to fail in sparse areas. The model training of CNN-based methods usually uses Euclidean distance as a loss function, which leads to some disadvantages, such as sensitivity to outliers (isolated counting objects). In addition, using the Euclidean distance will lose a part of the local spatial correlation between the GT density map and the predicted density map. Therefore, In Reference [23], a joint loss function of spatial correlation loss (SCL) and spatial abstract loss (SAL) have been proposed to improve the quality of the predicted density map. Some of the above methods have also been used for dense vehicle counting with good results. However, these methods are unsuitable for some complex traffic congestion scenarios where long-distance queuing vehicle features contain high spatial correlations, and the correlation information also hides continuous scale variations of vehicle counting. In addition, these methods do not consider analyzing channel correlations in complex human-vehicle mixed scenes to enhance the vital counting features of vehicles and make the predictions more accurate.

3 Our method

This work aims to estimate the density map and count the number of vehicles in traffic jam images using a novel multi-scale dilated convolution channel-aware deep network (MSCNet). In this section, we first define the dense vehicle counting in this paper, then introduce the architecture of MSCNet, and finally present the function of the corresponding modules.

3.1 Dense vehicle counting

The dense vehicle counting discussed in this paper refers to the count of dense vehicles in traffic jams or human-vehicle mixed scenes. Our method can generate a predicted density map for dense vehicles. Firstly, we use our method to predict an input image of dense vehicles to get a high-quality estimated density map. Then the integral of the estimated density map will obtain the accurate number of dense vehicles in the image.

3.2 Overview of the method

MSCNet is a single-column multi-branch CNN model that deals with continuous perspective changes, has large-scale variations, and fuses multiple information channels to enhance counting features. The proposed model mainly contains the front-end module (VGG16) as well as three other enhancement modules: the direction-based perspective coding module (DPCM), the multi-scale dilated residual module (MDRM), and the channel-aware attention module (CAM), as shown in Fig. 1.

Referring to the CSRNet [9] network structure, we use the first ten layers of VGG-16 [24] and reserve only three pooling layers as our front-end module. Firstly, the input image A is sent to the pre-trained VGG16 front-end module to obtain the feature map f_b, as shown in Eq. (1).

$$f_b = F_{VGG}(\mathrm{A}) \tag{1}$$

We use the transfer learning capability of VGG16 to extract the initial features of the input image A. However, the features extracted by VGG16 have a limitation in the dense vehicle analysis; VGG16 only encodes the limited receptive field over the entire input image. We deploy three modules, DPCM, MDRM, and CAM, to enhance the initial features to extract deeper spatial information and channel information of saliency. Our method employs an end-to-end training process to encode and decode the input image into a high-quality estimated density map, then obtain the number of dense vehicles.

3.3 DPCM

In the previous section, we mentioned the problem of continuous perspective changes encountered in dense vehicle counting studies. The work uses DPCM to deal with this problem, as shown in Fig. 2.

The DPCM uses SCNN [25] to encode the perspective information in the feature map from four directions. The module adopts the feature map f_b encoded from the front-end module as the input and mainly includes four convolution layers: the left layer, the right layer, the down layer, and the up layer. In Fig. 2, C, H, and W denote the feature map's channel, length, and width. The tensor is first divided into W slices in the left (and right) layer. Then the first slice begins the convolutional operation. Each convolution layer consists of a convolutional operation with a $C \times \omega$ size kernel and a ReLU activation function, where ω is a hyperparameter. Traditional CNN passes the output of this layer to the next layer to apply convolutional layer operations; however, this module adds a slice of convolutional output to the next slice as a new output and then continues to

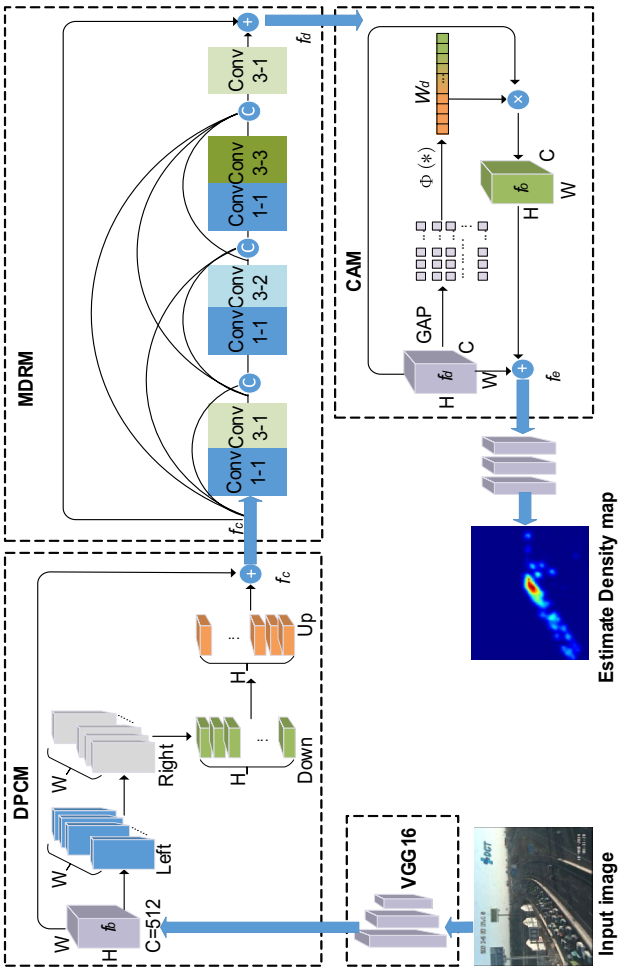

Fig. 1 The architecture of the proposed MSCNet for vehicles counting

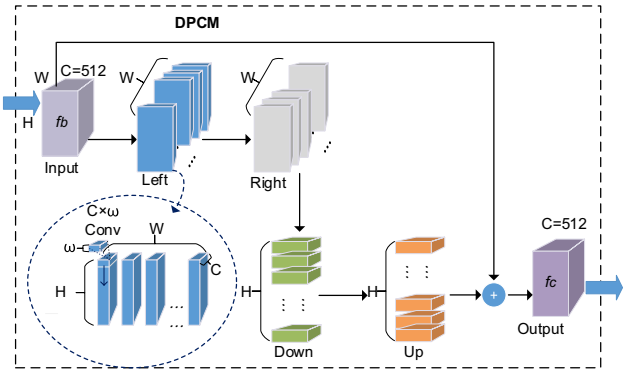

Fig. 2 The direction-based perspective coding module (DPCM)

apply convolutional layer operations to the new output until all slices have been superimposed. To visually show the convolution operation of the left layer, the formula for the feedforward computation of the left layer is defined as shown in Eq. (2),

$$L_W^i = L_W\left(L_W^{i-1} + f_W^i\right),\ i = 1, 2, 3 \ldots \ldots \mathrm{W} \tag{2}$$

where L_W (*) represents the left layer operation (Conv and ReLU), $f_W{}^i$ is the ith feature slice in the W direction. Similarly, the other three layers (right, down, and up layer) have the same operation except for the sliding direction of the convolution kernel and the slicing direction. Specifically, the slicing direction of the left (right) layers is from right (left) to left (right) along the W direction, and the sliding direction of the convolution kernel on each slice is from top to bottom; the slicing direction of the down (up) layers is from top (bottom) to bottom (top) along the H direction, and the sliding direction of the convolution kernel on each slice is from left to right. This structure is mainly used to sequentially encode the perspective information of the feature map from four directions and output the feature map f_c.

This module performs the convolutional operation in four directions to extract as much information as possible about vehicle queuing features hidden due to perspective effects, while the density feature of the vehicle distribution is also hidden in the information. DPCM transmits perspective information between pixel rows and pixel columns of the image: the left and right layers transmit feature information from right to left and from left to right along the W direction, and the down and up layers transmit the information from top to bottom and from bottom to top along the H direction. Since the perspective information of the whole image needs to be passed successively between multiple rows and columns, the network structure becomes deeper. According to the effectiveness of residual connection [26] on the deep network, this module also uses the residual connection method for learning. In addition, such a design has other advantages. The module does not change the channel dimension of the input feature and introduces global spatial information into the feature map. In summary, these operations effectively encode the spatial perspective information of the entire image. They are particularly suitable for extracting the object features of continuous long-distance shapes, such as queuing vehicles with a robust spatial relationship but poor appearance clues in traffic jam scenes.

3.4 MDRM

As discussed in Section 1, the large-scale variation is a major research issue in dense vehicle counting. The scale variation of densely arranged vehicles is almost continuous across the whole image and has an extensive range, so it is necessary to extract the scale variation features in the image more densely. As shown in Fig. 3, the MDRM mainly extracts the image features at continuous variable scales.

The module mainly contains three dilated convolution layers with different dilation rates of 1, 2, and 3. A 1×1 filter size convolution layer precedes each dilated convolution layer to control the number of feature channels. Moreover, a standard convolution kernel with a 3×3 filter size (dilation rate = 1) is adopted to fuse all concatenated features from the front dilated convolution layers and reduce the number of channels output at the end of the module. And a ReLU activation function is applied after every dilated convolution layer. Each dilated convolution layer within the module is densely connected with other layers so that each layer can access all the subsequent layers and transfer the feature information that needs to be extracted. After the dense connection, the module converges multiple receptive fields, increasing the captured scale diversity, as shown in Table 1. These three dilation rates are combined to form receptive fields of six sizes. For example, when the dilation rate is 1, the size of the receptive field is 3×3; when the dilation rate is 2, the size of the receptive field is 5×5; when the dilation rates combination is 1 and 3, the receptive field is 9×9.

The dilated convolution has been demonstrated in object segmentation tasks and can significantly improve the model accuracy. Compared with the pooling layer, the dilated convolution can retain the spatial information of the feature map. Although the deconvolution layer also preserves spatial information, it increases the additional complexity and execution delay of the model. The dilated convolution is a better choice, which uses sparse kernels instead of the pooling layer and convolution layer, and it can extract the spatial information of queuing vehicles at multi-scale. The dilated convolution extends the receptive field without increasing the number of parameters. In the dilated convolution, a small size convolution kernel with a $k\times k$ filter is amplified to $k+(k-1)\times(r-1)$ by different dilation rates r. This module uses 3×3 convolution kernels with three dilation rates (1,2,3) to form six receptive fields of different sizes, which can densely extract large-scale features and retain spatial information. In addition, the module also designs a residual connection to deal with the gradient vanishing problem in deep networks.

In this module, a meaningful way to efficiently use the multi-scale information is to concatenate all captured multi-scale features with the original input feature f_c. This operation adds the input feature of the module to the output feature, which becomes f_d. The function C (*) represents the concatenation of four feature maps: f_c, f_1, f_2, f_3, and execution of the 3×3 convolution operation is then performed to complete the channel fusion. The operation shown in Eq. (3) will also retain more initial feature information after dense multi-scale dilated convolution processing, which can integrate the image feature fc and multiple multi-scale feature information.

$$f_d = f_c + C\left(\left[f_c, f_1, f_2, f_3\right]\right) \tag{3}$$

MDRM achieves receptive fields of six sizes by combining three dilation rates, thus more densely extracting image features at different scales and preserving spatial information. The cooperative use of DPCM and MDRM can better extract the features of long-distance continuous large-scale variations for dense vehicle counting.

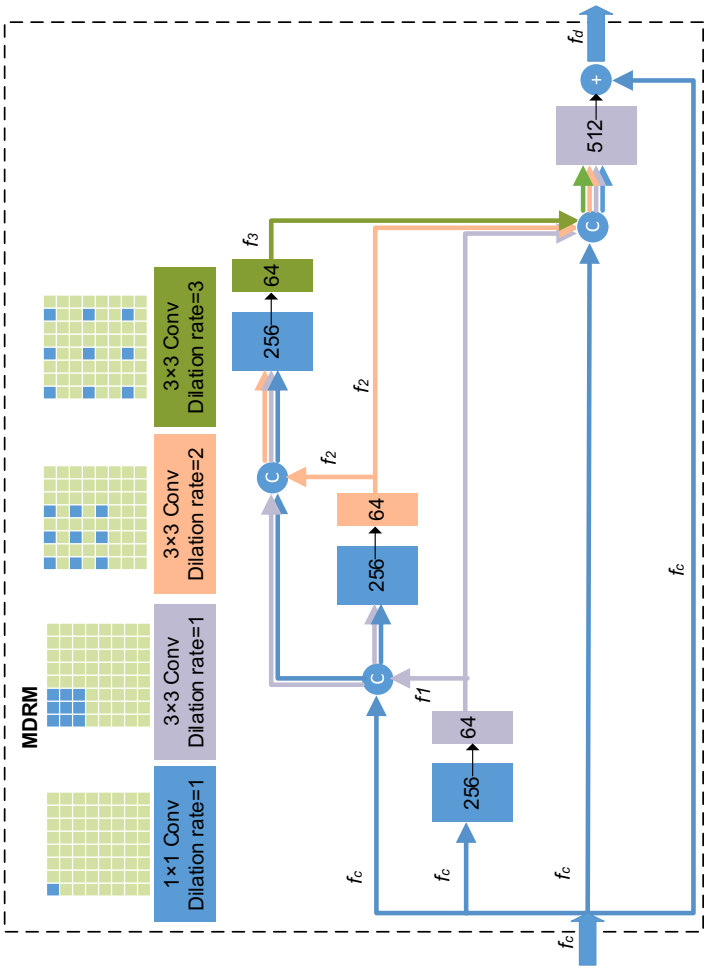

Fig. 3 The multi-scale dilated residual module (MDRM)

Table 1 The multiple receptive fields of dilated convolution layers

Dilation Rates Combination	Six Receptive Fields	
	Size	Diagram
1	3×3	
2	5×5	
1,2	7×7	
3		
1,3	9×9	
2,3	11×11	
1,2,3	13×13	

3.4.1 CAM

After encoding by DPCM and MDRM, the feature f_d is sent into the CAM, as shown in Fig. 4. DPCM and MDRM extract continuous multi-scale spatial information and complete features fusion in the spatial dimension, while CAM performs features fusion in the channel dimension. Some image classification tasks [27] also use such a module.

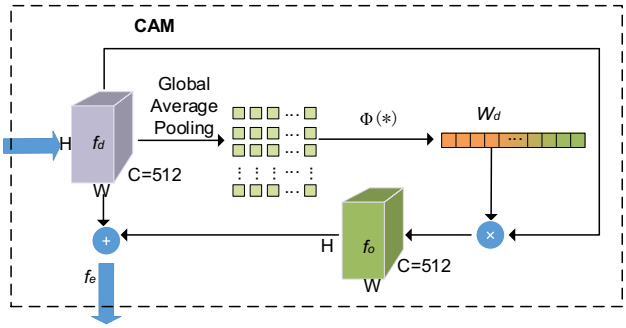

Fig. 4 The channel-aware attention module (CAM)

These tasks use channel attention mechanisms to extract important learned features hidden in the different channels of the feature map.

As discussed in Section 1, the module can learn the importance of different feature channels. Specifically, the model learns to obtain weights for each feature channel, then uses these weights to enhance useful counting features and suppress unimportant channels for vehicle counting tasks. The whole procedure can be described as Eq. (4).

$$f_e = f_d + \Phi\left(GAP\left(f_d, C\right)\right) \odot f_d \tag{4}$$

The feature map f_d is represented as a tensor $W \times H \times C$ (width, height, channel), and the tensor is first averaged to a $1 \times 1 \times C$ tensor by performing a global average pooling (GAP) operation on the W and H dimensions. The GAP function averages the information contained by all pixels in the image as a single value, which masks the spatial distribution information of the feature map and introduces attention to the channel dimension. This operation encodes spatial features as global features in each channel. Then, the function Φ (*) is used to process the $1 \times 1 \times C$ tensor to obtain the channel correlation. The function consists of a bottleneck network structure with two fully connected layers, a ReLU function, and a Sigmoid function. In this way, more attention can be paid to the correlation between channels while ignoring the correlation in spatial distribution, and the weight coefficient W_d (value: 0–1) of each channel can be obtained by Sigmoid function activation. Then, the weight coefficient of each channel is multiplied by the original input feature f_d to enhance the input feature f_d on the channel dimension. Finally, the enhanced feature f_o is concatenated with the input feature f_d to further enhance the feature map's representation. After CAM processing, the enhanced feature f_e is fed into a decoder consisting of several convolution layers to generate a predicted density map to obtain the target number. In essence, the CAM implements the attention mechanism on the channel dimension. This attention mechanism enables the model to pay more attention to the channels with more informative features and suppress those channels that are not important for counting features. Different feature channels introduce global information through the GAP function, and some channels show high activation states, which are related to specific vehicle counting features that need to be learned. Therefore, the CAM module can effectively extract important channel features for vehicle counting to generate an accurately predicted density map in human-vehicle mixed scenes.

4 Experiments

4.1 Experimental setup

The MSCNet is built on Ubuntu 18.04 and Pytorch 1.11 experimental environments. The important supporting hard-wares are Intel Core™ i7-12,700 4.7 GHz and GeForceRTX3090. The model is trained using the L2 loss function and Adam optimizer, and the learning rate is 0.00001. The hyperparameter ω in the DPCM module is set as 9. Data augmentation is used for training, where the original input image was randomly cropped to 1/2 of its original size and flipped horizontally. Accordingly, density maps used as training labels are cropped and flipped correspondingly. The affine transformation is used to make the size of labels consistent with the training output image size.

4.2 Evaluation Metrics

In this paper, we test the model using standard evaluation metrics for density map regression methods: mean absolute error (MAE) and the root mean squared error (RMSE) [28], which are defined as Eqs. (5) and (6),

$$MAE = \frac{1}{N}\sum_{i=1}^{N}\left|C_i - C_i^{GT}\right| \tag{5}$$

$$\mathrm{RMSE} = \sqrt{\frac{1}{\mathrm{N}}\sum_{\mathrm{i}=1}^{\mathrm{N}}\left(\mathrm{C_i} - \mathrm{C_i^{GT}}\right)^2} \tag{6}$$

where N is the total number of the test images, C_i^{GT} is the ground truth of the object count, and C_i is the predicted object count.

The Grid Average Mean Absolute Error (GAME) [29] is also used for model evaluation on a specific dataset during the experiment. The GAME error is defined as Eq. (7),

$$\mathrm{GAME(L)} = \frac{1}{\mathrm{N}}\sum_{\mathrm{i}=1}^{\mathrm{N}}\sum_{\mathrm{l}=1}^{4^{\mathrm{L}}}\left(\left|\mathrm{C_i^l} - \mathrm{C_{iGT}^l}\right|\right) \tag{7}$$

where N is the total number of test images, and C_i^l is the predicted object count of the input images within region l. C_{iGT}^l represents the ground truth object counting within the corresponding region. For each level L in Equation, the *GAME(L)* uses 4^L non-overlapping grids covering the entire image to segment the image. For example, when the value of L is 0, the GAME is equivalent to the MAE.

In addition, standard deviation (SD) is used to evaluate the predicted density maps, which are defined as Eqs. (8),

$$SD = \sqrt{\frac{1}{N}\sum_{i=1}^{N}\left(P_i - \mu\right)^2} \tag{8}$$

where N is the total number of pixels in the predicted images, and P_i is the value of the pixel. μ is the pixel average.

4.3 Comparisons with state-of-the-art

This study conducted comparative experiments with state-of-the-art methods on five different public datasets: TRANCOS [29], VisDrone2021 crowd & vehicle [30], and ShanghaiTech Part_A & Part_B [14]. The experiment results show that the proposed MSCNet model achieves excellent counting results on these five datasets. The statistics of these five datasets are shown in Table 2.

The TRANCOS dataset is a vehicle counting dataset for traffic jam scenes. The public transport dataset captured by road monitoring equipment also provides the region of interest (ROI) for evaluation. The dataset contains 1,244 images of crowded traffic scenes from different perspectives and 46,796 annotated vehicles.

The VisDrone2021 challenge dataset is derived from an object detection dataset with annotated bounding boxes for the targets. We modify the original dataset into two datasets of object counting: the vehicle dataset and the crowd dataset. The original dataset includes 11 categories of detected objects, and the categories of pedestrians and people

Table 2 Statistics of different datasets

Dataset	Images	Count Statistics			Average Resolution
		Max	Min	Total	
TRANCOS [29]	1,244	107	9	46,796	640×480
VisDrone2021 Vehicle [30]	5,581	390	9	199,986	991×1511
VisDrone2021 People [30]	3,440	298	9	108,987	969×1482
ShanghaiTech Part_A [14]	482	313	33	241, 677	589×868
ShanghaiTech Part_B [14]	716	578	9	88, 488	768×1024

are screened out to form a crowd dataset (VisDrone2021 crowd dataset). The annotation object of the new crowd dataset is the central coordinate of the head, and the annotation operation is redefined as Eq. (9).

$$\text{People}[X,Y] = \left[\text{bbox}_{\text{left}} + \frac{\text{bbox}_{\text{width}}}{2}, \text{bbox}_{\text{top}}\right] \tag{9}$$

In the same way as the method used to modify the VisDrone2021 crowd dataset, we select category car, category van, category truck, and category bus to form a vehicle dataset (VisDrone2021 vehicle dataset). The object annotation operation of the new vehicle dataset is defined as Eq. (10).

$$\text{Vehicle}[X,Y] = \left[\text{bbox}_{\text{left}} + \frac{\text{bbox}_{\text{width}}}{2}, \text{bbox}_{\text{top}} + \frac{\text{bbox}_{\text{height}}}{2}\right] \tag{10}$$

The ShanghaiTech dataset is a crowd dataset containing 1,198 images with 330,165 annotated persons. The dataset consists of two sub-datasets: the ShanghaiTech Part_A dataset with 482 images and the ShanghaiTech Part_B dataset with 716 images. The ShanghaiTech Part_A is a dataset with extremely crowded crowd scenes, with a total of 241,677 annotated persons; the ShanghaiTech Part_B is a relatively sparse dataset of crowd scenes, with a total of 88,488 annotated persons.

Figure 5 shows the comparison results of MSCNet with other methods on the TRANCOS dataset. The TRANCOS is a challenging dataset with various images from different perspectives and severe vehicle occlusion. The existing methods, such as [9, 29, 31, 32], have achieved good counting performance. Among them, IbPRIA 2015 is the first method to use the TRANCOS dataset for training and testing, while Hydra-3 s and CSRNet methods present leading test results on the TRANCOS dataset for vehicle counting. Although FCN-HA does not give the results from GAME (1) to GAME (3) in the corresponding paper, its experimental result of GAME (0) is excellent. CSRNet is a classical dense object counting network structure that also uses dilated convolution for feature extraction and currently has the best GAME values on the TRANCOS dataset. Our method gets a significant improvement on four different GAME levels: 3.49 GAME (0), 5.11 GAME (1), 7.81 GAME (2), and 14.01 GAME (3). The counting performance is better than these other methods. MSCNet and CSRNet are very close in the GAME (0) values. It can be inferred that both networks use dilated convolution to extract more detailed spatial information. MDRM in this paper uses dilated convolution lays that can pay more attention to extracting features of vehicles with large-scale variations, while

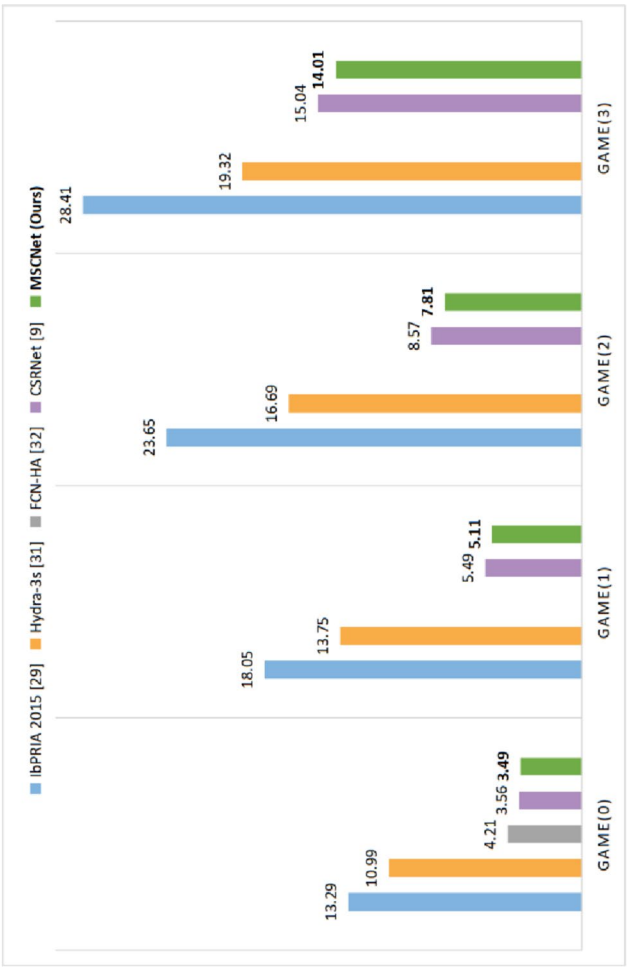

Fig. 5 Comparison of MSCNet with other methods on TRANCOS dataset

DPCM and MDRM are used in conjunction to solve the problem of long-distance continuous scale variations through the image.

Figure 6 shows the comparison results of MSCNet with other excellent counting methods [9, 14, 33, 34], on the VisDrone2021 vehicle dataset. Among them, CSRNet is a classical single-column crowd counting network structure using dilated convolution. MCNN is a typical network for multi-scale feature extraction using a multi-column network structure. PCC Net fully considers the impact of perspective changes. DSNet has excellent performance on many crowd datasets. The vehicle dataset is a challenging dataset with various classes of vehicles and complex backgrounds in each image, and the experiment results show that the MSCNet can handle this more complex dataset. Our method consistently shows a better counting performance than other methods, demonstrating the importance of the channel attention module CAM for handling more complex datasets. CAM can learn channel correlation through attention mechanisms to enhance important features for vehicle counting in complex human-vehicle mixed scenes.

Figure 7 shows the comparison results of MSCNet with other methods on the VisDrone2021 crowd dataset. Beyond vehicle counting, we also conduct comparative experiments on a crowd dataset to demonstrate the robustness and generality of our method. This crowd dataset is collected in the outdoor scenes with a complex background, and the counting objects in the images have a wide range of density distributions. In this paper, the experimental results of MSCNet are compared with those of other methods, such as [9, 14, 33, 34]. The results of this experiment show that the MSCNet can also deal with the crowd dataset with such a large-scale variation and complex background while achieving satisfactory counting results.

Figure 8 compares MSCNet with other methods [9, 14, 33–36] on two challenging crowd counting datasets: ShanghaiTech Part_A and ShanghaiTech Part_B. These two datasets are classical crowd counting datasets, and the perspective changes and scale variations presented in their images provide research challenges for many counting models based on CNN. The density of the object distribution in the ShanghaiTech Part_A dataset is larger than that in ShanghaiTech Part_B. It can be found that the MSCNet can obtain the best MAE and RMSE on ShanghaiTech Part _A. On the Part_A dataset, MSCNet can get 58.9 MAE and 94. 1 RMSE. On the Part_B dataset, MSCNet can get 7.5 MAE and 11.2 RMSE. Compared with the other classical methods, the average performance of this network model is better. In particular, our method outperforms the advanced methods CSRNet [9] and PCC Net [33] on the dense Part_A dataset and the sparse Part_B dataset. These results further demonstrate that the proposed MSCNet is suitable for dense vehicle counting and also shows excellent robustness and effectiveness for crowd counting.

MSCNet has shown excellent performance on TRANCOS, VisDrone2021 crowd and vehicle, ShanghaiTech Part_A and Part_B datasets; in particular, the vehicle datasets TRANCOS and VisDrone2021 vehicle exhibit the best performance. The crowd datasets VisDrone2021 crowd and ShanghaiTech are mainly used to prove the robustness and generality of our method, and the optimal performance is also achieved on ShanghaiTech Part_A, where the object is denser.

Our method performs best on the TRANCOS dataset, a classical vehicle dataset for dense object counting, and outperforms other excellent counting methods on the VisDrone2021 vehicle dataset. These excellent counting methods use dilated convolution, multi-column network structure, and perspective enhancements to improve the counting performance. However, our method overall considers the influence of the above factors on feature extraction of long-distance queuing vehicles and always shows better counting

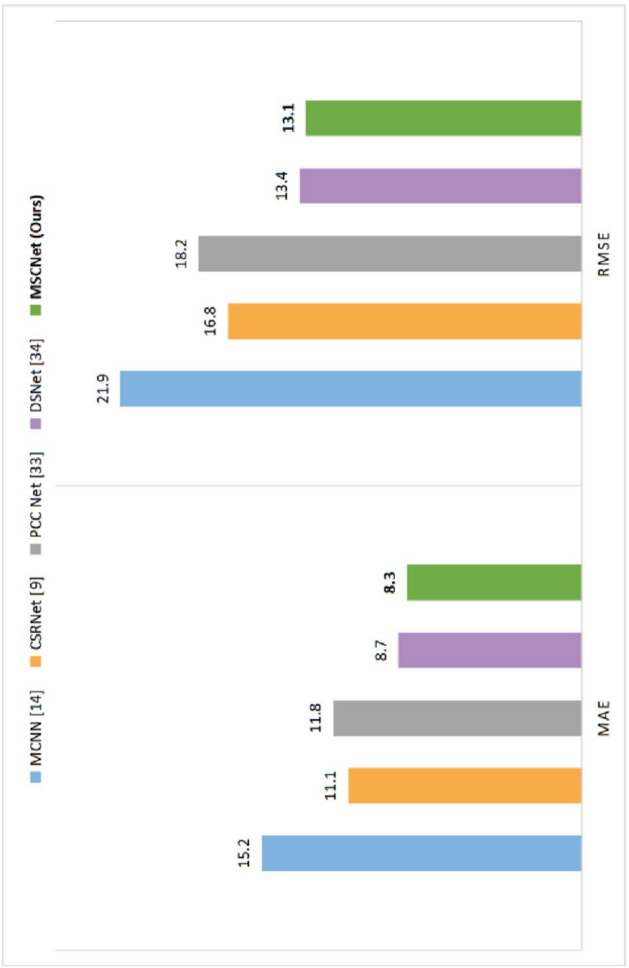

Fig. 6 Comparison of MSCNet with other methods on VisDrone2021 vehicle dataset

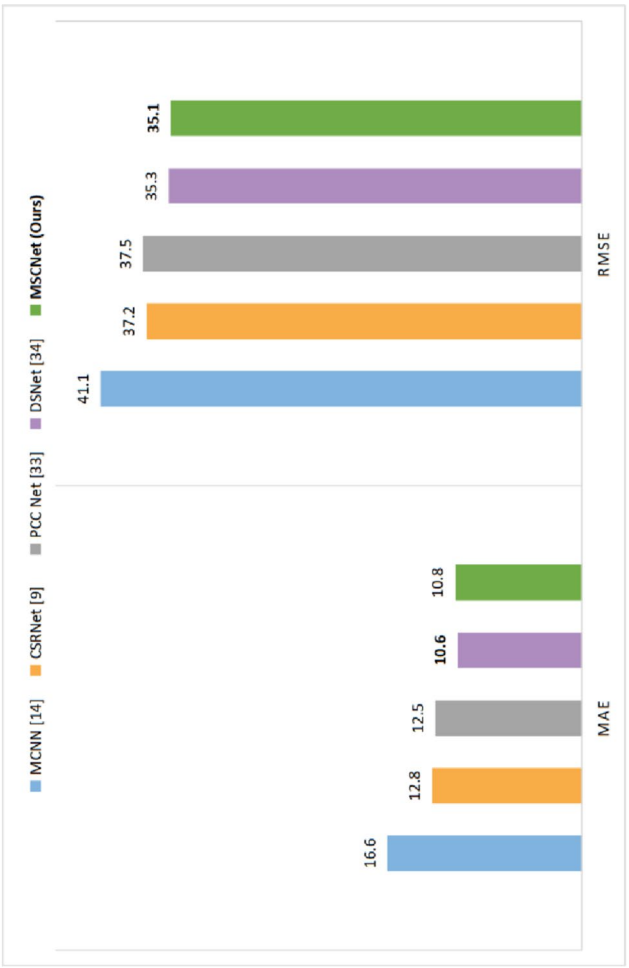

Fig. 7 Comparison of MSCNet with other methods on VisDrone2021 crowd dataset

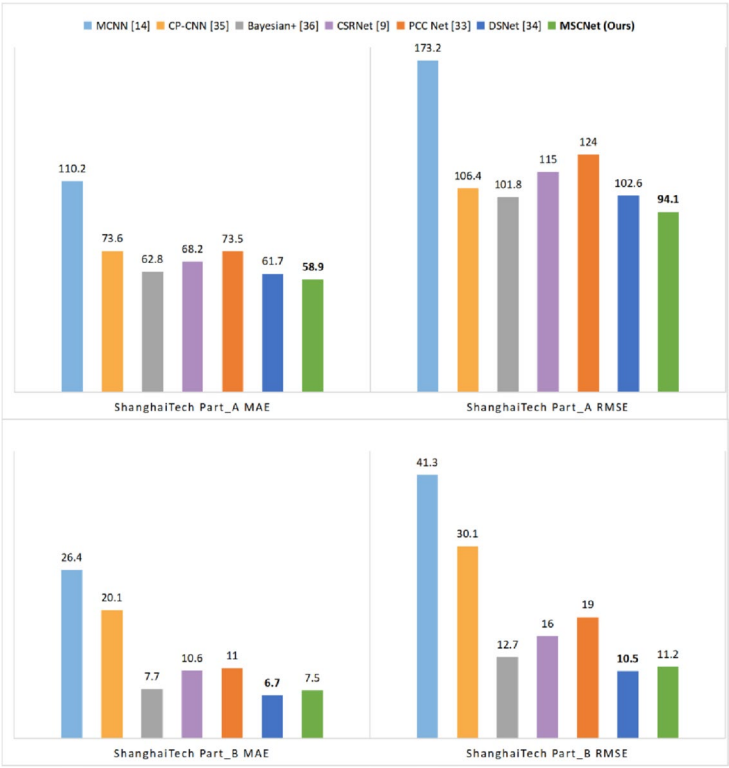

Fig. 8 Comparison of MSCNet with other methods on ShanghaiTech dataset

performance on vehicle datasets than other methods. The counting performance of our method is better than the state-of-the-art method CSRNet on the TRANCOS dataset.

This paper also presents the experimental results of the visualization on the TRANCOS and the VisDrone2021 vehicle dataset, as shown in Figs. 9 and 10. Each blue or red dot represents one image randomly selected from the datasets. The blue dot values indicate the actual number of counted objects, and the red dot values indicate the estimated number of objects predicted by the model. The visualized experimental results show that the change trends of the two curves are close, indicating that the model's prediction error amplitude is small and the minimum error is zero. The closer the red and blue curves fit, the better the model's performance.

This paper also evaluates the predicted density maps of the test images on TRANCOS, VisDrone2021 vehicle, VisDrone2021 crowd, ShanghaiTech Part_A, and ShanghaiTech Part_B datasets using standard deviation, and the values are shown in Fig. 11.

4.4 Ablation study

In this section, we conduct ablation research experiments on the VisDrone2021 vehicle dataset to investigate the effectiveness of the front-end module VGG16, as well as the three functional modules, DPCM, MDRM, and CAM. MSCNet with different front-end modules and MSCNet without these three functional modules are used as experimental subjects.

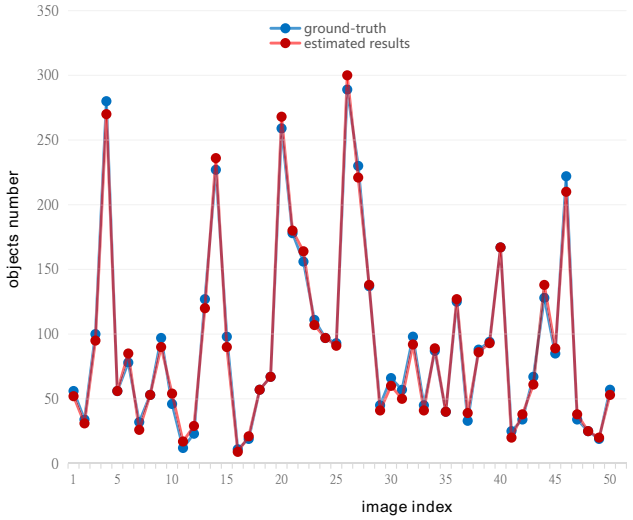

Fig. 9 The comparison between ground truth and estimated results on TRANCOS

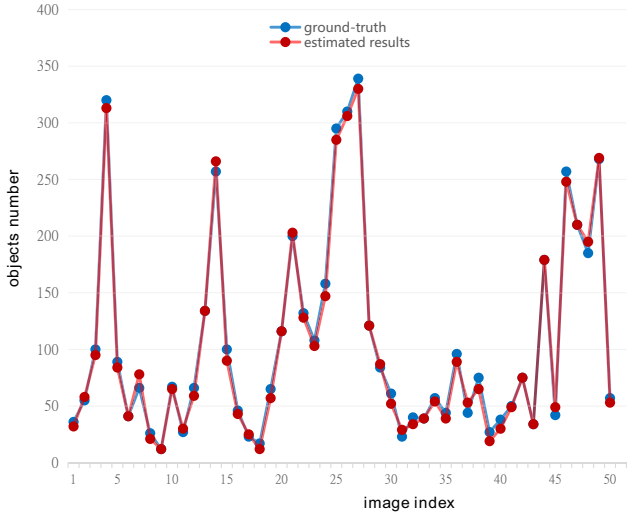

Fig. 10 The comparison between ground truth and estimated results on VisDrone2021 vehicle dataset

The comparative experiment uses the VGG and ResNet as the front-end modules. The detailed configuration information is as follows:

1) The MSCNet with different front-end modules: is denoted as the VGG-MSCNet (Ours) and the ResNet-MSCNet.
2) The MSCNet without DPCM: is denoted as the MSCNet-no-DPCM.
3) The MSCNet without MDRM: is denoted as the MSCNet-no-MDRM.
4) The MSCNet without CAM: is denoted as the MSCNet-no-CAM.

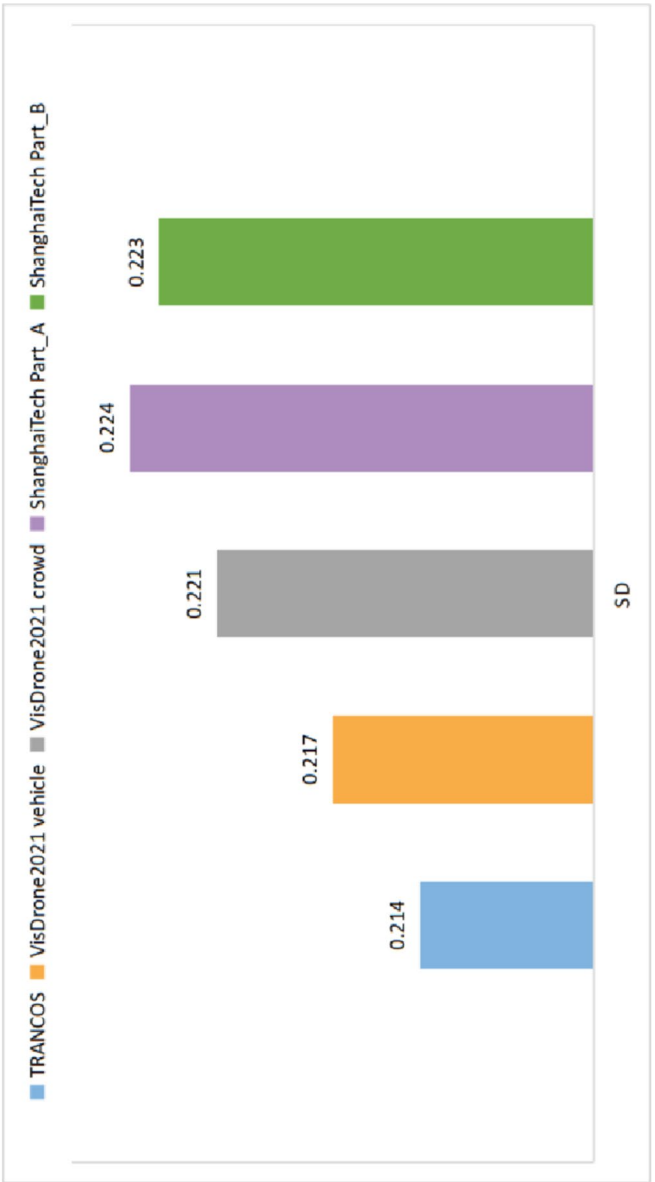

Fig. 11 SD of the predicted density maps

Figure 12 shows that replacing VGG16 with ResNet does not achieve better accuracy and results in lower MAE and RMSE, while experiments with configurations 2), 3), and 4) confirm the effectiveness of DPCM, MDRM, and CAM. These functional modules can extract valuable features hidden in spatial and channel dimensions to complete the feature enhancement.

5 Conclusions

This paper proposes a single-column and multi-branch deep vehicle counting network, which can effectively mine spatial data in images, generate high-quality density maps, and complete the task of dense vehicle counting in complex mixed scenes. In order to solve the problem of continuous long-distance scale variation for queuing vehicles, this work proposes DPCM and MDRM to get the long-distance and large-scale queuing vehicle feature information. Besides, the CAM is also employed to learn the important information of the channel feature to improve the feature extraction ability in complex mixed scenes. Experiment results based on the benchmark datasets show that the proposed method outperforms the existing classical methods and is robust. In the future, we will continue improving the feature extraction capabilities in mixed scenes and focus on studying the new loss functions to generate higher-quality GT density maps to improve counting performance.

Acknowledgements This work was supported by the National Natural Science Foundation of China (Grant No.62076117) and Jiangxi Key Laboratory of Smart City, China (Grant No.20192BCD40002).

Authors' contributions Qiyan Fu: Conceptualization, Methodology, Software, Validation, Investigation, Formal Analysis, Writing—Original Draft; Weidong Min (Corresponding Author): Conceptualization, Funding Acquisition, Resources, Supervision, Writing—Review & Editing; Chunbo Li: Data Curation, Visualization, Writing—Review & Editing; Haoyu Zhao: Resources, Investigation; Ye Cao: Writing—Review & Editing; Meng Zhu: Validation; All authors reviewed the manuscript.

Funding This work was supported by the National Natural Science Foundation of China (Grant No.62076117) and Jiangxi Key Laboratory of Smart City, China (Grant No.20192BCD40002).

Data availability All data created or used during this study are publicly available at the following websites: https://gram.web.uah.es/data/datasets/trancos/index.html, https://opendatalab.com/VisDrone, and https://github.com/desenzhou/ShanghaiTechDataset.

Declarations

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

References

1. Min W, Liu R, He D et al (2022) Traffic Sign Recognition Based on Semantic Scene Understanding and Structural Traffic Sign Location. IEEE Trans Intell Transp Syst 23(9):15794–15807
2. Zhao H, Min W, Wei X et al (2021) MSR-FAN: Multi-Scale Residual Feature-Aware Network for Crowd Counting. IET Image Process 15(14):3512–3521
3. Fan Z, Zhang H, Zhang Z et al (2022) A Survey of Crowd Counting and Density Estimation Based on Convolutional Neural Network. Neurocomputing 472:224–251
4. Dirir A, Ignatious H, Elsayed H et al (2021) An Advanced Deep Learning Approach for Multi-Object Counting in Urban Vehicular Environments. Future Internet 13(12):306

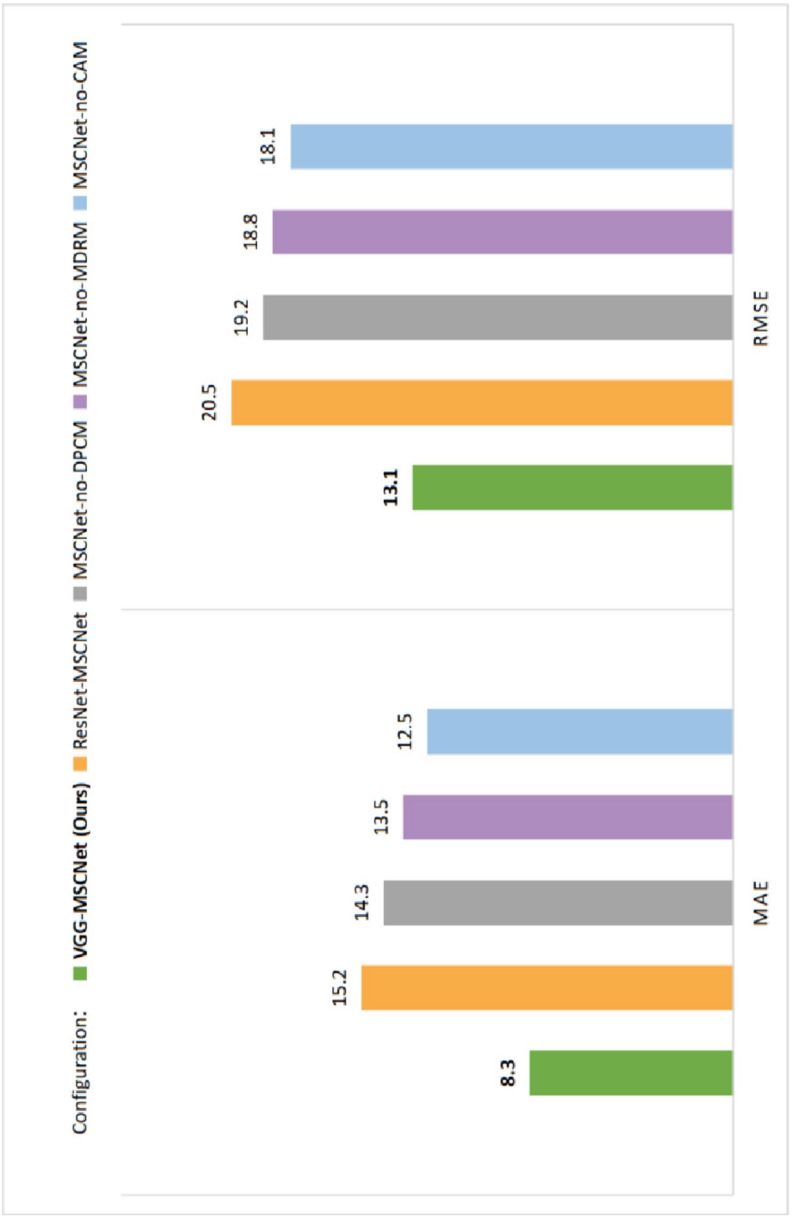

Fig. 12 The results of ablation research experiments

5. Dai Z, Song H, Wang X et al (2019) Video-Based Vehicle Counting Framework. IEEE Access 7:64460–64470
6. Girshick R, Donahue J, Darrell T et al (2014) Rich Feature Hierarchies for Accurate Object Detection and Semantic Segmentation. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 580–587
7. Liu Z, Zhang W, Gao X et al (2020) Robust Movement-Specific Vehicle Counting at Crowded Intersections. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops, pp. 614–615
8. Liang M, Huang X, Chen C et al (2015) Counting and Classification of Highway Vehicles by Regression Analysis. IEEE Trans Intell Transp Syst 16(5):2878–2888
9. Li Y, Zhang X, Chen D (2018) CSRNet: Dilated Convolutional Neural Networks for Understanding the Highly Congested Scenes. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 1091–1100
10. Antonini G, Thiran JP (2006) Counting Pedestrians in Video Sequences Using Trajectory Clustering. IEEE Trans Circuits Syst Video Technol 16(8):1008–1020
11. Lempitsky V, Zisserman A (2010) Learning to Count Objects in Images. In: Proceedings of the Advances in Neural Information Processing Systems, pp. 1324– 1332
12. Fu M, Xu P, Li X et al (2015) Fast Crowd Density Estimation with Convolutional Neural Networks. Eng Applic Artif Intell 43(auga):81–88
13. Zhang C, Li H, Wang X et al (2015) Cross-scene Crowd Counting via Deep Convolutional Neural Networks. In: Proceedings of the 2015 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), pp. 833–841
14. Zhang Y, Zhou D, Chen S et al (2016) Single-Image Crowd Counting via Multi-Column Convolutional Neural Network. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 589–597
15. Liu L, Wang H, Li G et al (2018) Crowd Counting using Deep Recurrent Spatial-Aware Network. In: Proceedings of the 2018 International Joint Conference on Artificial Intelligence (IJCAI), pp. 849–855
16. Chen J, Su W, Wang Z (2020) Crowd Counting with Crowd Attention Convolutional Neural Network. Neurocomputing 382:210–220
17. Szegedy C, Ioffe S, Vanhoucke V et al (2017) Inception–v4, Inception-ResNet and the Impact of Residual Connections on Learning. In: Proceedings of the 2017 AAAI Conference on Artificial Intelligence, pp. 4278–4284
18. Fiaschi L, Kthe U, Nair R et al (2012) Learning to Count with Regression Forest and Structured Labels. In: Proceedings of the 2012 International Conference on Pattern Recognition (ICPR), pp. 2685–2688
19. PhamVQ, Kozakaya T, Yamaguchi O et al (2015) COUNT Forest: CO-Voting Uncertain Number of Targets Using Random Forest for Crowd Density Estimation. In: Proceedings of the 2015 IEEE International Conference on Computer Vision (ICCV), pp. 3253–3261
20. WangY, Zou Y (2016) Fast Visual Object Counting via Example-Based Density Estimation. In: Proceedings of the 2016 IEEE International Conference on Image Processing (ICIP), pp. 3653–3657
21. Ciregan D, Meier U, Schmidhuber J (2012) Multi-Column Deep Neural Networks for Image Classification. In: Proceedings of the 2012 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), pp. 3642–3649
22. ZhouZ, Su L, Li G et al (2020) CSCNet: A Shallow Single Column Network for Crowd Counting. In: Proceedings of the 2020 IEEE International Conference on Visual Communications and Image Processing (VCIP), pp. 535–538
23. JiangX, Xiao Z, Zhang B et al (2019) Crowd Counting and Density Estimation by Trellis Encoder-Decoder Networks. In: Proceedings of the 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), pp. 6133–6142
24. Simonyan K, Zisserman A (2014) Very Deep Convolutional Networks for Large-Scale Image Recognition. arXivpreprint, arXiv.1409.1556
25. Pan X, Shi J, Luo P et al (2018) Spatial as Deep: Spatial CNN for Traffic Scene Understanding. In: Proceedings of the 2018 AAAI Conference on Artificial Intelligence 32(1):7276–7283
26. He K, Zhang X, Ren S et al (2016) Deep Residual Learning for Image Recognition. In: Proceedings of the 2016 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), pp. 770–778
27. Hu J, Shen L, Sun G et al (2018) Squeeze-and-Excitation Networks. In: Proceedings of the 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), pp. 7132–7141

28. Siva P, Javad Shafiee M, Jamieson M (2016) Real-Time, Embedded Scene Invariant Crowd Counting Using Scale-Normalized Histogram of Moving Gradients (HoMG). In: Proceedings of the 2016 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops, pp. 67–74
29. Guerrero-Gmez-Olmedo R, Torre-Jimnez B, Lpez-Sastre R et al (2015) Extremely overlapping Vehicle Counting. In: Proceedings of the 2015 Iberian Conference on Pattern Recognition and Image Analysis (IbPRIA), pp. 423–431
30. Zhu P, Wen L, Bian X et al (2018) Vision meets drones:A challenge. arXivpreprint, arXiv:1804.07437
31. Onoro-Rubio D, Lpez-Sastre RJ (2016) Towards Perspective-Free Object Counting with Deep Learning. In: Proceedings of the 2016 European Conference on Computer Vision (ECCV), pp. 615–629
32. Zhang S, Wu G, Costeira JP (2017) FCN-rLSTM: Deep Spatio-Temporal Neural Networks for Vehicle Counting in City Cameras. In: Proceedings of the 2017 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), pp. 3667–3676
33. Gao J, Wang Q, Li X (2019) PCC Net: Perspective Crowd Counting via Spatial Convolutional Network. IEEE Trans Circuits Syst Video Technol 30(10):3486–3498
34. Dai F, Liu H, Ma Y et al (2021) Dense Scale Network for Crowd Counting. In: Proceedings of the 2021 International Conference on Multimedia Retrieval (ICMR), pp. 64–72
35. Sindagi VA, Patel VM (2017) Generating High-Quality Crowd Density Maps Using Contextual Pyramid CNNs. In: Proceedings of the 2017 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), pp. 1861–1870
36. Ma Z, Wei X, Hong X et al (2019) Bayesian Loss for Crowd Count Estimation With Point Supervision. In: Proceedings of the 2019 IEEE International Conference on Computer Vision (ICCV), pp. 6142–6151

Qiyan Fu received the M.E. degree in Electronic and Communication Engineering from Nanchang University, China in 2017. She is currently pursuing the Ph.D. degree at Nanchang University, China. Her current research focuses on artificial intelligence and computer vision.

Weidong Min received the B.E., M.E. and Ph.D. degrees in computer application from Tsinghua University, China in 1989, 1991 and 1995, respectively. He is currently a Professor and the Dean, School of Metaverse, Nanchang University, China. He is an Executive Director of China Society of Image and Graphics. His current research interests include image and video processing, artificial intelligence, big data, distributed system and smart city information technology.

Chunbo Li MS candidate, CCF member. He is currently pursuing the M.E degree at Nanchang University, China. His current research interests include computer vision, and computer architecture.

Haoyu Zhao obtained the B.E. and M.E. degrees of computer science and technology from Nanchang University, China in 2019, and 2022. He is currently pursuing the Ph.D. degree at Fudan University. His research interests include computer vision, deep learning, and video understanding.

Ye Cao born in June 1983, is a lab master at the School of Information Engineering of Nanchang University, China. Her research direction is electronic information engineering.

Meng Zhu received the B.E. and M.E. degrees in computer science and technology from Nanchang University, China in 2018 and 2021, respectively. He is currently pursuing the Ph.D. degree at Nanchang University, China. His current research interests include computer vision, and natural language processing.