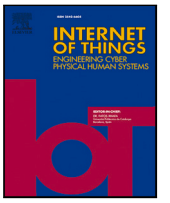
Internet of Things 26 (2024) 101167

Contents lists available at ScienceDirect

Internet of Things

journal homepage: www.elsevier.com/locate/iot



# Privacy-aware crowd counting by decentralized learning with parallel transformers

Jinyong Chen [a], Mingliang Gao [a,*], Qilei Li [b], Xiangyu Guo [a], Jianyong Wang [a], Jing'an Cheng [a], Xuening Xing [a]

[a] School of Electrical and Electronic Engineering, Shandong University of Technology, Zibo, 255000, China
[b] School of Electronic Engineering and Computer Science, Queen Mary University of London, London, E1 4NS, United Kingdom

ARTICLE INFO

*Keywords:*
Federated learning
Decentralized learning
Crowd counting
Attention mechanism
Deep learning

ABSTRACT

With the rapid advancement of deep learning, the performance of crowd counting has improved significantly. Nonetheless, existing crowd counting models primarily depend on a broad dataset gathered from a variety of individuals for model training. However, this diverse dataset comes at the cost of compromising people's privacy. Hence, the need to address privacy concerns when counting crowds in dense scenes is becoming increasingly apparent. To tackle this issue, we propose a novel framework called the Decentralized Learning with Parallel Transformer network (DLPTNet). Based on the federated learning mechanism, the DLPTNet adopts a decentralized learning framework that implements parameter sharing instead of data sharing. The DLPTNet consists of two pivotal modules, namely Halo Attention (HA) module and the Density-aware Transformer (DAT) module. The HA module has a large perception radius, which enhances its ability to perceive the context around the objects and extract more extensive information from local regions to address the occlusion issue in dense scenes. Meanwhile, the DAT module leverages the parallel mechanism of Density-aware Attention (DDA) to further capture long-range dependencies between different positions and thus gains learning of the correlations and density distributions of various regions within dense crowds globally.

## 1. Introduction

In recent years, privacy-aware crowd counting has emerged as a crucial task with far-reaching implications ranging from urban planning to public safety [1,2]. Despite remarkable advancements in performance over the past decade driven by the evolution of deep learning, existing models predominantly depend on extensive datasets gathered from numerous individuals across various scenarios and locations. Unfortunately, this approach often neglects the privacy concerns of individuals and fails to consider that data in certain environments cannot be shared. Federated Learning, as a decentralized approach, has gained attention for training models across distributed data sources while ensuring data privacy [3,4]. Combining federated learning with crowd counting not only enhances the accuracy and efficiency of crowd counting tasks but also reinforces considerations regarding data privacy and security.

Crowd counting methods can be classified into three categories, detection-based methods [5–7], regression-based methods [8–10], and CNN-based methods [11–13]. Detection-based [5–7] methods involve detecting each individual and then accumulating to obtain the total count. However, there is room for improvement when it comes to handling occlusion issues within dense crowds. To

* Corresponding author.
*E-mail address:* mlgao@sdut.edu.cn (M. Gao).

https://doi.org/10.1016/j.iot.2024.101167
Received 14 September 2023; Received in revised form 29 December 2023; Accepted 19 March 2024
Available online 1 April 2024

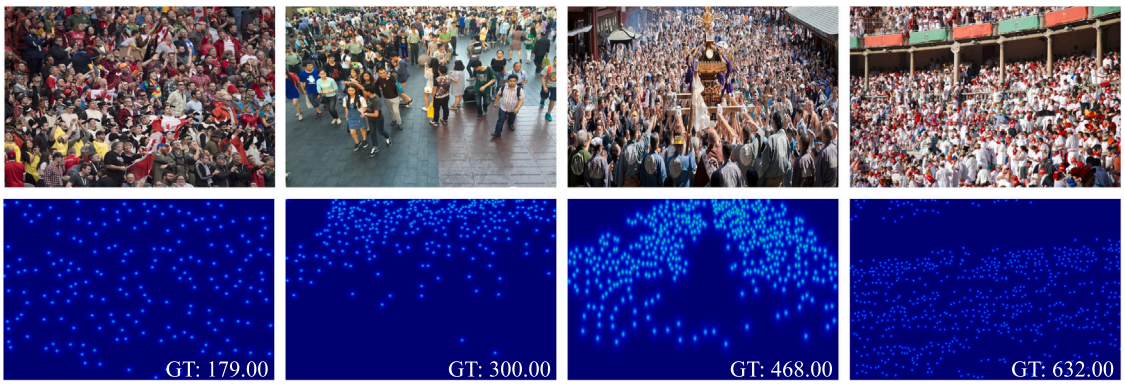



**Fig. 1.** Examples of congestion scenes in dense crowds. The first row is the input image, and the second row corresponds to the ground truth and the number of people.

better address the aforementioned issues, some regression-based approaches [8–10] have been proposed. These approaches directly learn the mapping from image patches to the number of people. However, these methods often overlook spatial information and focus excessively on low-level details, and thus fail to generate high-quality crowd density maps. With the rapid advancements in deep learning, convolutional neural networks (CNNs) have become the primary network architecture for density estimation and crowd counting [14,15].

Although the aforementioned methods have made significant advancements, they still encounter challenges. From Fig. 1, it is evident that the crowd is densely packed, with objects in proximity and instances of occlusion or overlap. Furthermore, there is a significant variation in scene density, and thus poses a challenge for the design of crowd counting algorithms. Such scenarios typically occur in congested urban areas, public events, transportation hubs, and other locations where large crowds gather, which present unique challenges for the advancement of crowd counting. Furthermore, crowd counting tasks may sometimes require collaboration with other organizations or individuals, such as government departments, research institutions, or businesses. In such cases, ensuring data privacy becomes another challenge for crowd counting tasks.

To overcome the challenges posed by occlusion in dense scenes and data privacy in crowd counting, we propose a decentralized learning framework built upon the foundation of parallel Transformer, termed Decentralized Learning with Parallel Transformer network (DLPTNet). By adopting the federated learning mechanism, we achieved parameter sharing instead of data sharing, and thus protect privacy data. Specifically, to address the occlusion issue in dense scenes, we integrate the halo attention (HA) module. The HA module enhances the ability to perceive the context around the targets and thus enables it to extract extensive information from the local region in crowded scenes. Furthermore, to capture long-range dependencies between different regions and leverage the parallel computing mechanism of the multi-head self-attention in the Transformer, we proposed the density-aware Transformer (DAT) module. This enables the model to gain a learning of the correlations and density distributions among various regions within dense crowds globally. The combination of the HA and DAT modules effectively enhances the performance of crowd counting tasks. To sum up, the contributions of this paper are as follows.

1. A decentralized learning framework rooted in parallel Transformer is proposed to overcome the obstacles posed by occlusion in dense crowd scenes and data privacy.
2. An HA module is introduced to enhance contextual awareness and extract local information from occluded regions. Meanwhile, a DAT module that leverages the parallel computing mechanism of the Transformer is built to further capture long-distance dependencies between different regions, and thus enable the model to gain a learning of the correlation and density distribution among various areas within dense crowd scenes globally.
3. We substantiate the effectiveness of the proposed model through a comprehensive set of experiments, and showcase its accuracy and robustness in privacy-aware crowd counting.

The remaining sections of the paper are structured as follows. Section 2 presents an overview of research efforts closely related to the content of this paper. Detailed insights into the proposed method are shown in Section 3. Experimental analyses and discussion are presented in Section 4. The conclusion is drawn in Section 5.

## 2. Related work

### *2.1. Transformer models in crowd counting*

The Transformer models [16,17] gain much traction in crowd counting. The self-attention mechanism introduced by the Transformer captures global relationships within the entire crowd, which improves the accuracy of crowd counting. Liang et al. [18] pioneered to leverage the self-attention mechanism of the Transformer to extract semantic crowd information. In their work, the

multi-head attention mechanism of the Transformer empowers the model to attend to features of different regions simultaneously, which enhances its capacity to effectively capture the distribution of crowd density. Liu et al. [19] introduced an innovative method that employs a multiscale token Transformer for count-guided fusion and a multiscale deformable Transformer decoder for modal-guided enhancement. This approach facilitates effective interaction and enhancement between modalities and crowd information. Tran et al. [20] introduced an approach that leverages the attention mechanism of vision Transformers to integrate local features and spatial information, which contributes to a considerable decrease in crowd counting errors.

### 2.2. Federated learning in crowd counting

Federated learning can be regarded as a form of distributed machine learning, and it offers the capability to train models on multiple local data sources while preserving data privacy. McMahan et al. [21] introduced a federated learning strategy to reduce communication overhead during decentralized deep network training with distributed data without centralizing data. This minimizes data transmission, improves model convergence efficiency, and is particularly suitable for scenarios prioritizing data privacy and communication bandwidth. Senthilkumar et al. [22] employed a federated learning-based approach with the federated averaging algorithm to decentralize training, expedite the process, and ensure data privacy. This approach strives to minimize training time while safeguarding data privacy and enhancing crowd counting accuracy. Pang et al. [23] utilized a horizontal federated learning framework to train crowd counting models while ensuring privacy preservation. The method empowers the smart surveillance system to leverage model aggregation for learning without the need to access sensitive data on local devices. As a result, it circumvents the necessity of transmitting video data, which results in reduced communication costs and safeguards against potential leaks of raw data. Tan et al. [24] utilized federated learning algorithms to achieve distributed model training while enabling model aggregation without sharing raw data. This approach effectively addresses privacy and data sharing concerns in indoor unmanned aerial vehicle crowd investigation. Jiang et al. [25] applied federated learning algorithms to address the issues of data reliability and privacy protection in mobile crowd sensing. In this approach, users of mobile devices can participate in model training while only sharing the updated parameters of the model. This enhances the performance and generalization capability of the model.

### 2.3. Attention mechanisms in crowd counting

Attention mechanisms excel in capturing local features for crowd counting, which enables effective individual localization within a crowd. Zhai et al. [13] proposed a dual attention-aware network that emphasizes spatial dependencies across the feature map for accurate head localization and manages channel relations to highlight discriminative information. This approach enhances crowd counting by addressing spatial and channel-related challenges. Guo et al. [26] introduced a triple attention and scale-aware network to mitigate background clutter, which employs three-dimensional attention operations on the input tensor to capture interaction dependencies across dimensions and distinguish object regions. Zhai et al. [27] presented an innovative attentive hierarchy network, which incorporates a re-calibrated attention module at different levels to mitigate background interferences, and a feature enhancement module to identify head regions at different scales. Guo et al. [28] introduced the dense attention fusion network, which incorporates an iterative attention fusion module, primarily utilizing the multiscale channel attention unit to mitigate background clutter's impact. Zhai et al. [29] introduced a feature pyramid attention network, which incorporates an attention module that focuses on the crowd region and mitigates the influence of misleading information. This enhancement improves the accuracy of localizing dense areas in input data without relying on prior knowledge. Traditional mechanisms exhibit limitations by potentially confining their focus to local regions, which can make it challenging to capture the overall relationships within a crowd.

## 3. Methodology

### 3.1. Overview

The architecture of the proposed DLPTNet is illustrated in Fig. 2. It consists of four components, *i.e.*, feature extractor, HA module, DAT module, and decoder. First, the feature extractor adopts VGG-19 to extract intricate feature representations from input images, which encompass spatial and semantic information pertaining to crowd distribution. The HA module captures extensive local information around the objects and enhances counting performance in dense scenes. The DAT module consists of two cascaded encoders, which incorporate density-aware attention (DAA) and feedforward unit (FFU). The position embedding generator (PEG) is only present in the backend of the first encoder. The purpose is to capture long-range dependencies between different positions and globally gain learning of the correlations and density distributions in various regions within dense crowds. The final decoder is employed to upsample the enhanced feature map and predict the density map.

### 3.2. Decentralized learning framework

To address concerns related to data privacy and security, we propose a decentralized learning framework aimed at facilitating collaborative model training, as illustrated in Fig. 3. It enables the merging of model parameters from different locations or devices without sharing raw data, which leads to more accurate and better-performing predictions. Specifically, each local device retains its local data and conducts model training on-site. Each local device only shares the updated parameters of its model, which avoids the sharing of raw data. This approach effectively safeguards data privacy and reduces data transmission and communication costs.

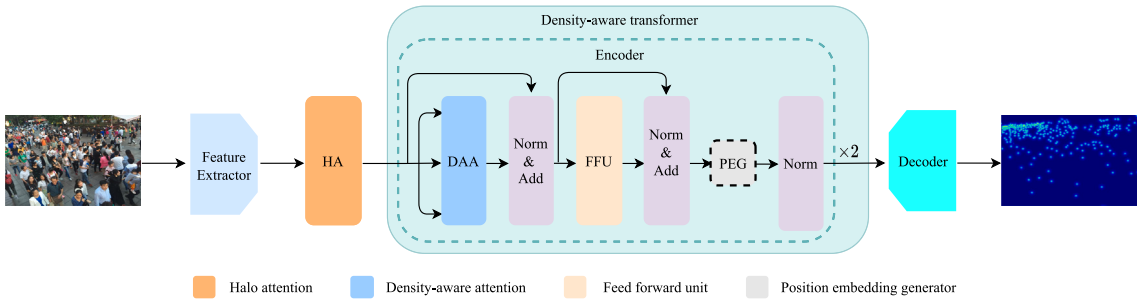



**Fig. 2.** The architecture of the DLPTNet for crowd counting. The FFU consists of two convolutional layers performing linear transformations and a non-linear activation function. This constitutes a perception layer within the Transformer, which is designed to learn relationships among different input features. The green dashed box indicates that the DAT module consists of two encoders, while the black dashed box indicates that PEG exists only in the first encoder. The decoder consists of four deconvolution layers, which are employed for upsampling to restore the original resolution size and regress to generate a density map.

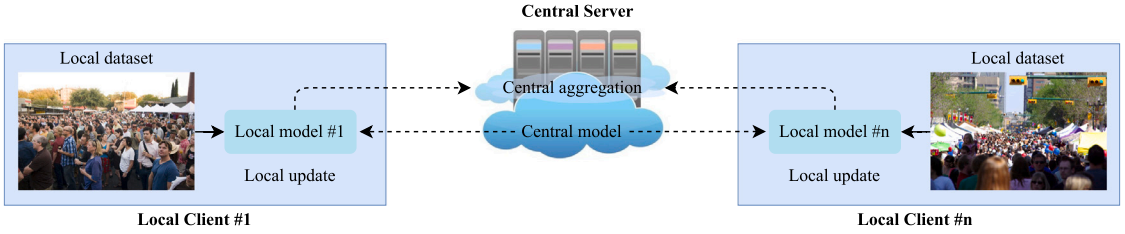



**Fig. 3.** The framework of the proposed decentralized learning. Local update refers to downloading the weight parameters of the global model from the central server to update the weight parameters of the local model. The weight parameters of the global model are aggregated from the weight parameters of all local models by the central server.

The objective of central aggregation is to aggregate the weight parameters from multiple local models using a weighted approach. We calculate the reciprocal of the metrics received from each local model to determine the proportional weight of that model in the global model. It is formulated as,

$$W_g = \sum_{i=1}^{n} W_i \times \frac{1}{e_i} \times \frac{1}{\sum_{i=1}^{n} \frac{1}{e_i}}, \tag{1}$$

where $W_g$ indicates the weight of the global model, $W_i$ represents the weights of each local model, and $e_i$ corresponds to the metrics of each individual local model. Decentralized learning refers to the independent training of the same model on each local client with its local dataset. The sharing of weight parameters among different clients occurs only during weight updates or information aggregation of the global model. If all local clients use the same test set, the weight of the best-performing local model is set to 1 during aggregation, while the weights of the other local models are set to 0. Thus, the global model is equivalent to selecting the best-performing local model. In essence, our decentralized learning framework offers a novel solution for crowd counting tasks by striking a balance between data privacy and performance enhancement.

### 3.3. Halo attention module

Halo attention [30] introduces overlapping local windows to enhance interactions between these windows. It assists the model in capturing local relationships within occluded regions of the input feature maps and thus improves the counting performance. By allowing for overlap between windows, HA can comprehensively consider dependencies between different local regions, which enhances the model's perception of local information. Therefore, halo attention proves to be an effective self-attention mechanism that enhances the counting performance in crowd counting tasks, especially in dense scenes. The primary workflow of the HA module is depicted in Fig. 4. First, the input feature map $F$ is subjected to a patch-wise operation, which divides it into $H \times W$ patches, followed by a linear transformation to generate a query matrix $Q$. The haloing operation involves adding a border of zero values around the feature map and then performing sliding window operations. In other words, a sliding window operation is applied to the input feature map using a $3 \times 3$ convolutional kernel, and thus generates the key matrix $K$ and value matrix $V$. This operation increases the perception radius, which enhances its ability to perceive the local region and extract more extensive information. $K$ and $V$ use a $3 \times 3$ window, which focuses on local occluded regions. For $Q$, dividing the input feature map into $1 \times 1$ windows helps to learn relationships between different occluded areas, not just limited to the current occluded region. To enhance the parallel computing capability of the model, a multi-head attention mechanism is employed. Next, the correlations between different local patches are computed using the query matrix $Q$ and key matrix $K$. Before assigning weights to different local patches, relative

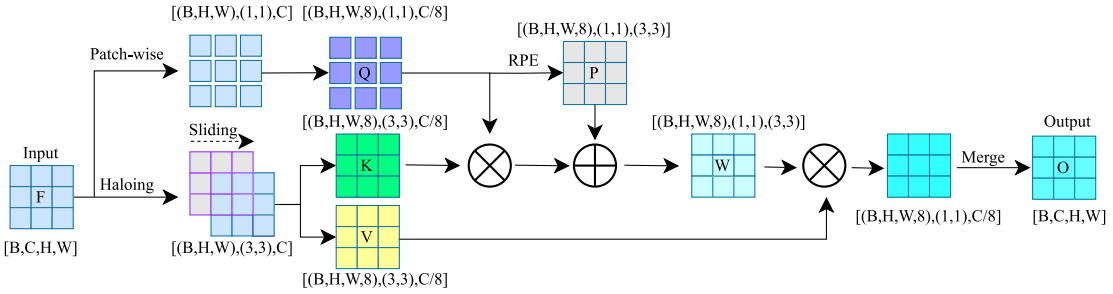



**Fig. 4.** The framework of HA module. The notation $[B, C, H, W]$ represents the batch size, number of channels, height, and width of the input feature map $F$, respectively. $(1, 1)$ and $(3, 3)$ indicate the size of the sliding window, which corresponds to the dimensions of each image patch. The number 8 signifies that there are eight attention heads. For instance, $[(B, H, W, 8), (3, 3), C/8]$ indicates the presence of $B \times H \times W \times 8$ image patches, each with a size of $3 \times 3$, and a channel dimension (*i.e.*, the length of feature vectors) of $C/8$.

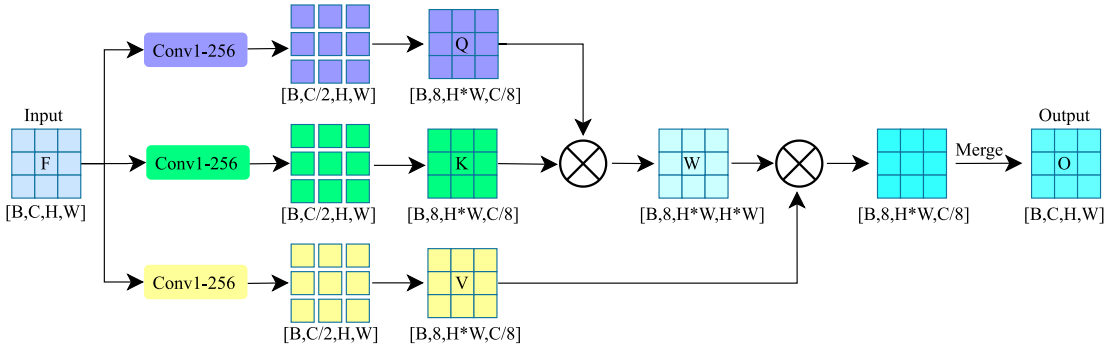



**Fig. 5.** The framework of DAA module. The notion $[B, C, H, W]$ respectively represents the batch size, number of channels, height, and width of the input feature map $F$. $Conv1 - 256$ indicates a convolutional layer with a $1 \times 1$ kernel and an output channel size of 256. Its role is to partition the input feature map into blocks, each of size $1 \times 1$. The number 8 signifies the presence of 8 attention heads. For instance, the form of the query matrix $Q$ is denoted as $[B, 8, H * W, C/8]$, which means there are $B \times 8 \times H \times W$ image patches (*i.e.*, the number of feature vectors), each of size $1 \times 1$, and each image patch has $C/8$ channels (*i.e.*, the length of feature vectors).

positional embedding (RPE) is introduced to learn the density levels in different regions and thus enables more accurate weight calculations. Finally, the value matrix $V$, which has been assigned weights, is merged and reorganized to restore the original input shape for further processing. This process assists the model in capturing spatial relationships and local information when dealing with crowded areas, ultimately improving counting performance.

### 3.4. Density-aware transformer module

The DAT module is introduced as an extension of the HA module, with the goal of further addressing occlusion in dense scenes. Unlike the HA module, which primarily enhances local object perception, the DAT module not only focuses on local regions but also captures features from the global context. The DAT module is built to learn the correlations and density distribution in different regions of dense crowds by leveraging the self-attention mechanism of the Transformer. To introduce positional information between different features, we introduce a PEG between two cascaded encoders. By performing a $3 \times 3$ convolution operation on the feature maps, we can obtain positional features. It is worth noting that deep separable convolution is utilized in the PEG, where convolution operations are applied separately to each channel. This approach aids the model in gaining an understanding of the interrelationships and local details among different positions. In each encoder, the residual connection mechanism is employed to ensure that the original feature information is not lost. The second encoder, built upon the first encoder, further enhances its ability to perceive dense crowds by capturing a broader context of information. The "add & norm" operation indicates residual connections and layer normalization in the DAT module, and thus helps improve the model's training stability and feature representation capacity. The FFU performs a non-linear mapping in the channel dimension using a $1 \times 1$ convolutional layer and thus enhances the feature representation capability. The purpose is to capture the complex relationships within the input feature map.

To capture long-range dependencies among different regions in dense crowds and aid the model in globally learning the correlations and density distribution between different areas, we propose density-aware attention, as shown in Fig. 5. For the input feature map, we employ $1 \times 1$ convolution operations to generate corresponding query, key, and value matrices. Simultaneously, we reduce the number of channels to half, which means halving the length of feature vectors for each local patch. This helps reduce the computational burden on the model, and thus enhances computational efficiency. Furthermore, we utilize a multi-head attention

**Table 1**
Information of the datasets adopted for comparison.

| Dataset | # Images | Train | Val | Test | Average resolution | Min | Max | Avg | Total |
|---|---|---|---|---|---|---|---|---|---|
| ShanghaiTech Part A [32] | 482 | 300 | – | 182 | 589 * 868 | 33 | 3,139 | 501 | 241,677 |
| ShanghaiTech Part B [32] | 716 | 400 | – | 316 | 768 * 1024 | 9 | 578 | 123 | 88,488 |
| UCF-QNRF [33] | 1,535 | 1,201 | – | 334 | 2013 * 2902 | 49 | 12,865 | 815 | 1,251,642 |
| JHU++ [34] | 4,372 | 2,272 | 500 | 1,600 | 910 * 1430 | 0 | 25,791 | 346 | 1,515,005 |
| NWPU-Crowd [35] | 5,109 | 3,190 | 500 | 1,500 | 2191 * 3209 | 0 | 20,033 | 418 | 2,133,375 |
| CARPK [36] | 1,448 | 989 | – | 459 | 720 * 1280 | 1 | 188 | 62 | 89,777 |
| PUCPR+ [36] | 125 | 100 | – | 25 | 720 * 1280 | 0 | 331 | 135 | 16,456 |

mechanism with 8 attention heads to further boost the model's capability to model the correlations and density distribution between different positions accurately. This aids in capturing features and relationships within dense crowd scenarios more precisely. Next, we introduce weights for the value matrix $V$ to highlight the correlations and density distribution in different areas. The formula is as follows,

$$O = W(F)V(F) = Softmax(\frac{Q(F)K(F)^T}{\sqrt{d_k}})V(F), \tag{2}$$

where $Q(F)$, $K(F)$, and $V(F)$ represent the distributions of their respective query, key, and value matrices. $d_k$ denotes the dimension of each attention head, which is 32. Finally, dimension rearrangement is applied to the obtained $O(F)$ to restore the original input's dimension for subsequent operations.

### 3.5. Ground truth generation

We employ the Focal Inverse Distance Transform (FIDT) map [31] to generate the ground truth. Compared with ground truth generated through the Gaussian kernel, FIDT ensures no overlap among nearby heads even in extremely dense crowds. It provides an accurate representation of crowd density and includes precise head annotations. The formula is as follows,

$$I = \frac{1}{P(x,y)^{\alpha \times P(x,y)+\beta} + C}, \tag{3}$$

where $I$ represents the FIDT map, with $\alpha$ and $\beta$ set as 0.02 and 0.75, respectively. $C$ is a non-zero constant used to prevent division by zero, and it is set to 1. $P(x, y)$ denotes the distance between any pixel $(x, y)$ and the nearest annotated head position $(x', y')$. It is formulated as,

$$P(x,y) = \min_{(x',y')\in D} \sqrt{(x-x')^2 - (y-y')^2}, \tag{4}$$

where $D$ indicates the set of all head annotations $(x', y')$.

### 3.6. Loss function

The widely used Mean Squared Error (MSE) loss function is adopted to minimize the difference between predicted values and ground truth. Let $Y_j$ denote the ground truth crowd count and $\hat{Y}_j$ represent the predicted crowd count for the $i$th image. The MSE loss $L_{MSE}$ for the $j$th image can be formulated as,

$$L_{MSE} = \frac{1}{N}\sum_{j=1}^{N} \|Y_j - \hat{Y}_j\|_2^2, \tag{5}$$

where $N$ is the total number of crowd counts in an image and $j$ iterates through each individual count. The optimization process seeks to minimize the $L_{MSE}$ loss across all training images, and thus quantify the disparity between predicted and actual crowd counts and aid the model's convergence to accuracy.

## 4. Experiments

### 4.1. Datasets

Seven datasets are applied to evaluate the performance of the proposed DLPTNet. The essential information of these datasets is shown in Table 1. ShanghaiTech Part A [32] consists of images collected from the internet, while ShanghaiTech Part B [32] is composed of images gathered from the downtown area of Shanghai. UCF-QNRF [33] features a large-scale crowd with diverse scenes, multiple perspectives, and variations in lighting. JHU++ [34] includes numerous images featuring weather-based degradations and illumination variations. NWPU-Crowd [35] is a large-scale crowd counting dataset obtained from the internet. CARPK [36] comprises drone-view images from four different parking lots. PUCPR+ [36] is a versatile vehicle counting dataset that includes various weather conditions.

**Table 2**
Comparison results on the ShanghaiTech dataset. The best results are highlighted in **bold**.

| Method | Part A | | Part B | |
|---|---|---|---|---|
| | MAE | RMSE | MAE | RMSE |
| MCNN [32] | 110.2 | 173.2 | 26.4 | 41.3 |
| CMTL [37] | 101.3 | 152.4 | 20.0 | 31.1 |
| NLT [38] | 93.8 | 157.2 | 11.8 | 19.2 |
| Switch-CNN [39] | 90.4 | 135.0 | 21.1 | 30.1 |
| C-CNN [40] | 88.1 | 141.7 | 14.9 | 22.1 |
| A-CCNN [41] | 85.4 | 124.6 | 19.2 | 31.5 |
| SaCNN [42] | 83.8 | 139.2 | 16.2 | 25.8 |
| MATT [43] | 80.1 | 129.4 | 11.7 | 17.5 |
| AMCNN [44] | 76.1 | 110.7 | 15.3 | 27.4 |
| PCCNet [45] | 73.5 | 124.0 | 19.2 | 31.5 |
| DNCL [46] | 73.5 | 112.3 | 18.7 | 26.0 |
| IG-CNN [47] | 72.5 | 118.2 | 13.6 | 21.1 |
| ACM-CNN [48] | 72.2 | 103.5 | 17.5 | 22.7 |
| CSRNet [49] | 68.2 | 115.0 | 10.6 | 16.0 |
| SCAR [50] | 66.3 | 114.1 | 9.5 | **15.2** |
| DENet [51] | 65.5 | 101.2 | 9.6 | 15.4 |
| DLPTNet (ours) | **58.4** | **95.0** | **9.3** | 15.6 |

### 4.2. Implementation details

In this study, all experiments are carried out within the PyTorch framework. The training and testing procedures are executed on an NVIDIA RTX3080Ti GPU. To optimize the weight parameters of the trained model, we employ the Adam optimizer, initializing the learning rate to 1e−4 and applying a weight decay of 5e−4. Due to the high resolution of images, we perform random cropping of the training set images into $256 \times 256$ dimensions and apply random horizontal flips for data augmentation to conserve memory during training. The batch size for training is set to 16, and a total of 3000 training iterations are conducted. For testing, a batch size of 1 is used. Both training and testing in the experiments are conducted on a single client. In ablation experiments, considering scenarios with $n$ different local clients, the random selection of $1/n$ of the training set by each client for training leads to a reduction in the training data. To prevent overfitting, we apply $colorJitter$ for image augmentation every $n$ epoch.

### 4.3. Evaluation protocols

We employ the Mean Absolute Error (MAE) and Root Mean Square Error (RMSE) as evaluation metrics [12,13]. The value of MAE is computed by the average absolute difference between predicted values and corresponding ground truths across all test samples,

$$MAE = \frac{1}{N}\sum_{i=1}^{N}\left|y_i - \hat{y}_i\right|, \tag{6}$$

The value of RMSE is calculated as the square root of the average squared difference between predicted values and ground truths

$$RMSE = \sqrt{\frac{1}{N}\sum_{i=1}^{N}|y_i - \hat{y}_i|^2}, \tag{7}$$

where $N$ represents the number of test images, and $y_i$ and $\hat{y}_i$ denote the predicted and ground truth values for the $i$th image, respectively.

### 4.4. Experiments on ShanghaiTech dataset

Table 2 presents a comparative evaluation of the DFPT against state-of-the-art (SOTA) methods. On Part A [32], the proposed method achieves impressive scores of 58.4 for MAE and 95.0 for RMSE. Compared with DENet [51], which employs a multi-scale pyramid structure, the proposed method achieves a 10.9% reduction in MAE and 6.1% decrease in RMSE on the Part A dataset. When compared with the attention-based SCAR [50], it reports 12.0% decrease in MAE and 16.7% reduction in RMSE on the Part A dataset. Regarding Part B [32], the DLPTNet still ranks first in terms of MSE when compared with SCAR and DENet. It achieves 2.1% reduction in MAE compared with SCAR and 3.1% reduction compared with DENet. While the score of RMSE is slightly inferior to SCAR and DENet, it only increases by 2.6% and 1.3%, respectively. Nevertheless, it maintains a strong performance compared with other methods, which secures the third position in the rankings. The reason for the higher RMSE on the Part B dataset is that in some samples, the distribution of the crowd is relatively sparse, with large gaps between individuals. This makes it challenging to accurately estimate the positions of each person within these samples and thus leads to a larger RMSE. Subjective comparisons on the ShanghaiTech dataset are depicted in Figs. 6 and 7. The first row displays the input images, the second row shows the corresponding ground truth, and the bottom row exhibits the respective prediction density map. It can be observed that the predicted density distribution approximates the ground truth, especially in images with significant congestion, where the density map exhibits a sense of hierarchy.

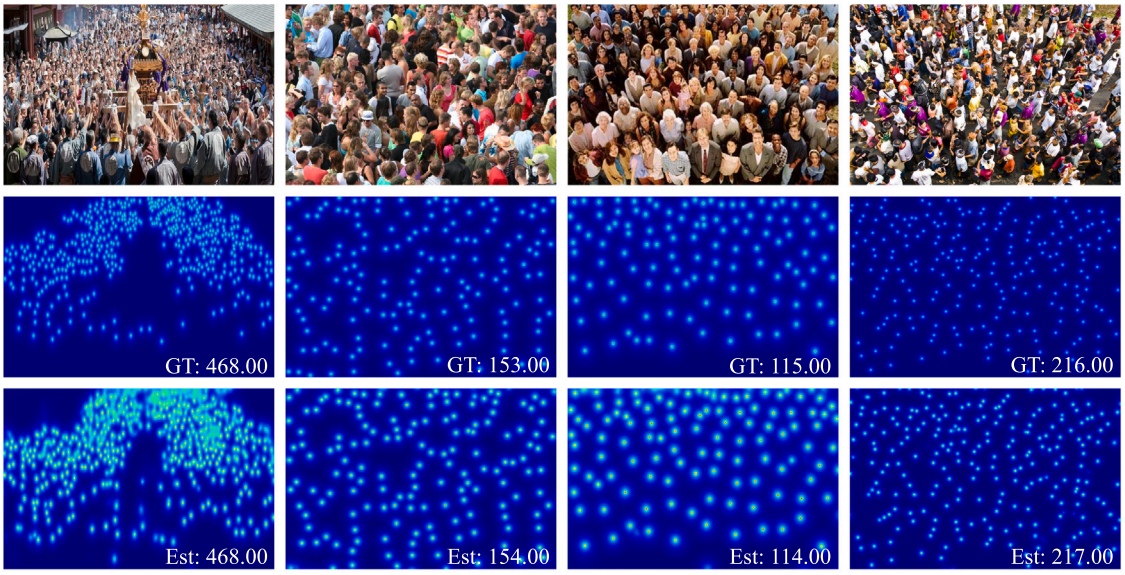



**Fig. 6.** Subjective results of ShanghaiTech Part A.

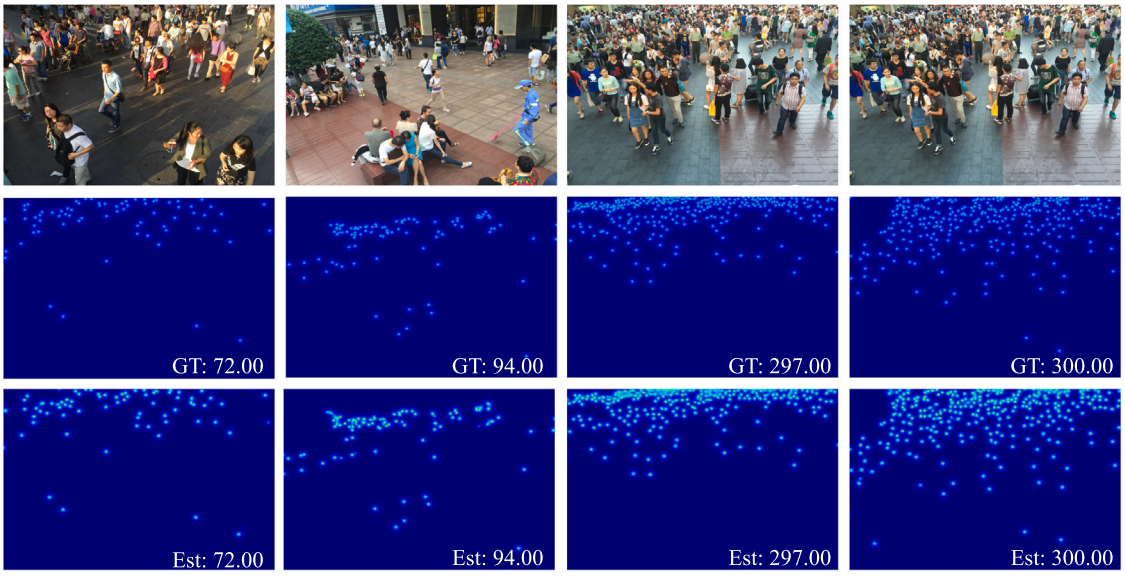



**Fig. 7.** Subjective results of ShanghaiTech Part B.

### *4.5. Experiments on UCF-QNRF dataset*

The objective comparison results on the UCF-QNRF [33] are reported in Table 3. Compared with PCCNet [45], which addresses crowd counting from different viewpoints, the proposed method exhibits an MAE reduction of 18.6% and an RMSE reduction of 8.7%. When compared with SCAR [50], although the RMSE is slightly higher, the MAE is reduced by 6.2%, which ranks us in the first position. The predicted density maps generated can be seen in Fig. 8. It can be observed that the locations of high-density areas in the density map are consistent with the actual densely crowded areas.

### *4.6. Experiments on JHU++ dataset*

The objective comparison results on the JHU++ [34] dataset are shown in Table 4. On the JHU++ dataset, although DLPTNet exhibits a slightly higher RMSE compared to CSRNet [49], with an increase of 10%, it still maintains a certain advantage by reducing

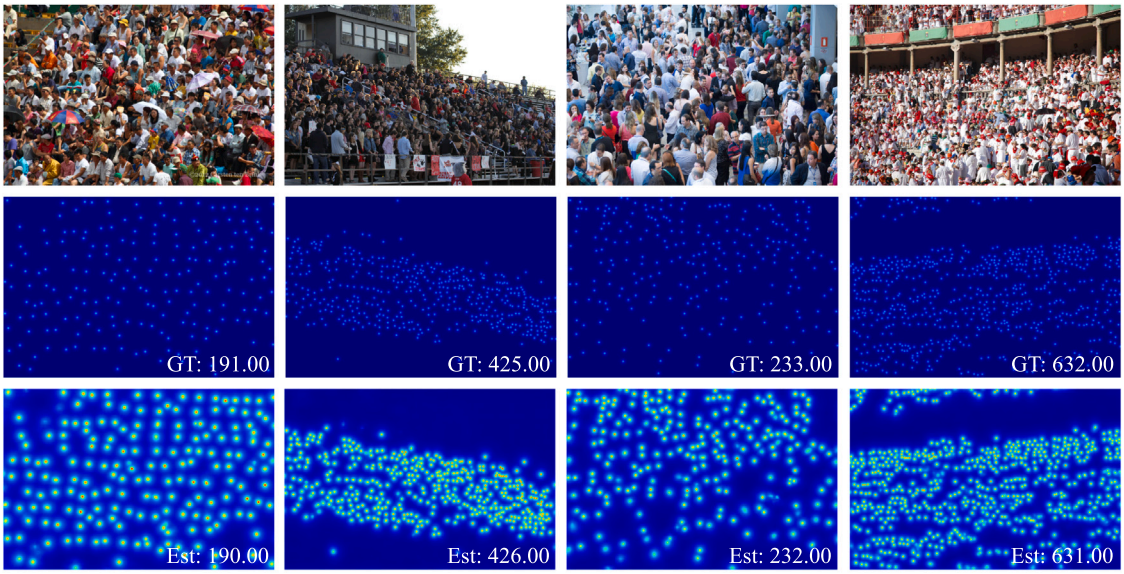



**Fig. 8.** Subjective results of UCF-QNRF dataset.

**Table 3**
Comparison results on the UCF-QNRF dataset. The best results are highlighted in **bold**.

| Method | MAE | RMSE |
|---|---|---|
| Zhang et al. [52] | 467.0 | 498.5 |
| Idress et al. [53] | 315.0 | 508.0 |
| MCNN [32] | 277.0 | 509.1 |
| SCAR [50] | 264.8 | 418.3 |
| CMTL [37] | 252.0 | 514.0 |
| Switch-CNN [39] | 228.0 | 445.0 |
| NLT [38] | 172.3 | 263.1 |
| PCCNet [45] | 148.7 | 247.3 |
| CSRNet [49] | 129.0 | **209.0** |
| DLPTNet (ours) | **121.0** | 225.8 |

**Table 4**
Comparison results on the JHU++ dataset. The best results are highlighted in **bold**.

| Method | MAE | RMSE |
|---|---|---|
| MCNN [32] | 188.9 | 483.4 |
| A-CCNN [41] | 171.2 | 453.1 |
| LSC-CNN [54] | 112.7 | 454.4 |
| SANet [55] | 91.1 | 320.4 |
| CSRNet [49] | 85.9 | **309.2** |
| DLPTNet (ours) | **77.7** | 340.1 |

MSE by 9.5%. Fig. 9 illustrates a comparison of subjective results on the JHU++ dataset. It is evident from the predicted density maps that they accurately portray the spatial distribution of the crowd, which allows for easy recognition of areas with differing levels of crowd density.

### *4.7. Experiments on NWPU-crowd dataset*

Table 5 presents the objective comparison results on the NWPU-Crowd [35] datasets. One can see that the proposed DLPTNet exhibits a 2.0% reduction in MAE and an impressive 7.1% decrease in RMSE compared with BL [63] trained with point supervision. While DLPTNet ranks second in terms of RMSE compared to GSANet [59], it outperforms by achieving an 11% reduction in MSE, which secures the top ranking. The subjective results are shown in Fig. 10. It can be observed that our predicted density maps vividly depict the spatial distribution of the crowd, which effectively highlights areas with varying levels of congestion.

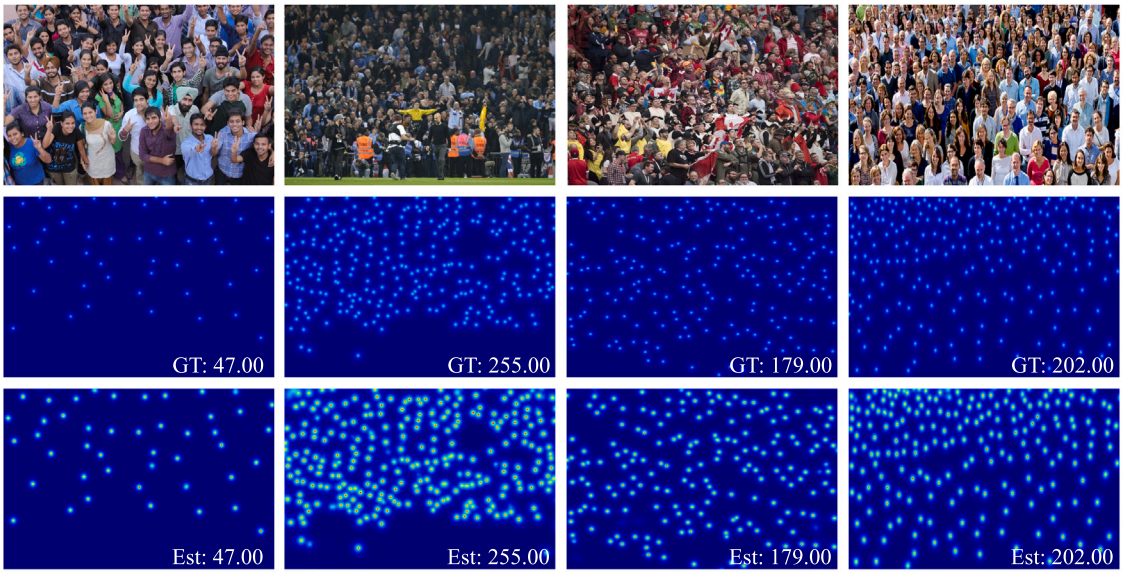



**Fig. 9.** Subjective results of JHU++ dataset.

**Table 5**
Comparison results on the NWPU-Crowd dataset.
The best results are highlighted in **bold**.

| Method | MAE | RMSE |
|---|---|---|
| MCNN [32] | 232.5 | 714.6 |
| SANet [55] | 190.6 | 491.4 |
| A-CCNN [41] | 176.5 | 520.6 |
| RAZNet [56] | 152.8 | 907.3 |
| ADMG [57] | 152.8 | 907.3 |
| STANet [58] | 122.6 | 468.3 |
| GSANet [59] | 116.1 | **415.3** |
| PCCNet [45] | 112.3 | 457.0 |
| SUA [60] | 111.7 | 443.2 |
| SCAR [50] | 110.0 | 495.3 |
| TopoCount [61] | 107.8 | 438.5 |
| SFCN [62] | 105.7 | 424.1 |
| BL [63] | 105.4 | 454.2 |
| DLPTNet (ours) | **103.3** | 421.9 |

### *4.8. Experiments on CARPK and PUCPR+ datasets*

The objective comparison results on the CARPK [36] and PUCPR+ [36] datasets are presented in Table 6. On the CARPK dataset, compared with TSANet [26], DLPTNet achieves a remarkable 31.0% reduction in MAE and a substantial 35.3% decrease in RMSE. Although there is a mere 0.1% increase in RMSE on the PUCPR+ dataset, it attains the top rank by reducing MSE by 11.4%. Furthermore, when compared with PSGCNet [64], DLPTNet showcases a substantial 9.2% decrease in RMSE. The subjective comparison results on the CARPK and PUCPR+ datasets are shown in Figs. 11 and 12. It can be observed that DLPTNet excels not only on crowd datasets but also demonstrates outstanding performance on vehicle datasets. This indicates that DLPTNet has a broad range of potential applications and thus extends beyond crowd counting tasks, to areas such as vehicle counting. Furthermore, it can be observed that on the PUCPR+ dataset, DLPTNet is capable of detecting vehicles that are partially submerged at the edges of the image background. Therefore, DLPTNet demonstrates effectiveness in addressing challenges in crowded environments and offers some relief from background interference.

Occlusion noise and background noise are the most prominent types of noise that significantly impact performance. Occlusion noise typically refers to interference caused by partial occlusion of individuals due to high crowd density. Background noise encompasses visual elements in the image other than the crowd, which may introduce errors. Specifically, occlusion noise may involve partial obstruction or concealment of certain areas within the crowd, while background noise may include non-crowd elements in the image, such as trees, buildings, etc. From Figs. 6 to 12, it can be noted that the proposed DLPTNet excels in suppressing occlusion noise and mitigates the impact of background noise to some extent. This indicates that the DLPTNet exhibits strong accuracy in handling occlusion and background interference.

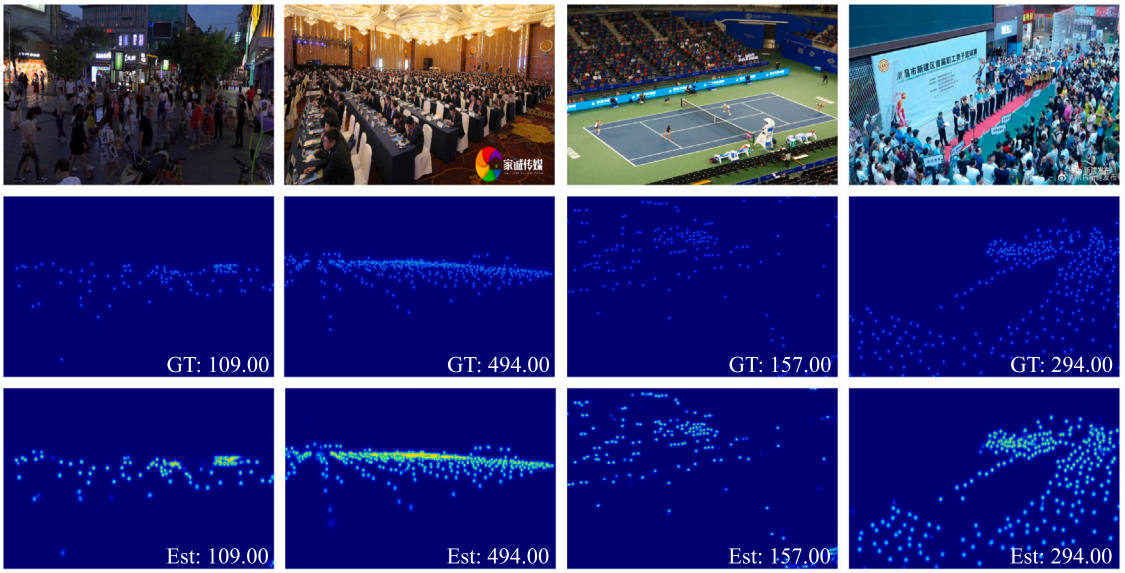



**Fig. 10.** Subjective results of NWPU-Crowd dataset.

**Table 6**
Comparison results on the CARPK and PUCPR+ datasets. The best results are highlighted in **bold**.

| Methods | CARPK | | PUCPR+ | |
|---|---|---|---|---|
| | MAE | RMSE | MAE | RMSE |
| YOLO [65] | 102.89 | 110.02 | 156.72 | 200.54 |
| FRCN [66] | 74.40 | 82.30 | 109.20 | 144.50 |
| LEP [67] | 51.83 | – | 15.17 | – |
| LPN [36] | 23.80 | 36.79 | 22.76 | 34.46 |
| SSD [68] | 28.20 | 23.30 | 32.90 | 42.10 |
| RetinaNet [69] | 16.62 | 22.30 | 24.58 | 33.12 |
| One-Look Regression [70] | 59.46 | 66.84 | 21.88 | 36.73 |
| MCNN [32] | 39.10 | 43.30 | 21.86 | 29.53 |
| SCRDet [71] | 11.10 | 25.40 | 9.10 | 13.50 |
| FCOS [72] | 10.70 | 13.60 | 16.00 | 23.80 |
| CSRNet [49] | 11.48 | 13.32 | 8.65 | 10.24 |
| BL [63] | 9.58 | 11.38 | 6.54 | 8.13 |
| PSGCNet [64] | 8.15 | 10.46 | 5.24 | 7.36 |
| TASNet [26] | 7.16 | 10.23 | 5.16 | **6.67** |
| DLPTNet (ours) | **4.94** | **6.62** | **4.52** | 6.68 |

### *4.9. Ablation studies*

**Ablation study on the pivotal components** To validate the effectiveness of the HA and DAT modules in DLPTNet, we conduct ablation experiments on the ShanghaiTech Part A dataset. The ablation experiments on the HA module and DAT module are presented in Table 7. The components are illustrated as follows,

- "Baseline" represents the configuration without the inclusion of HA and DAT modules, which consist solely of the front-end VGG19 and the back-end decoder. It can be observed that the performance is not the best, with an MAE of 63.8 and an RMSE of 105.8.
- "Baseline+HA" refers to adding only the HA module to the baseline model. There is a reduction of 4.8% in MAE and a decrease of 4.9% in RMSE, which indicates a performance improvement. This proves that the HA module can indeed address occlusion issues in crowded scenes by extracting more feature information from locally occluded regions.
- "Baseline+DAT" indicates adding only the DAT module to the baseline model, which leads to a decrease in MSE by 2.1% but an increase in RMSE. Compared with the HA module, the DAT module extracts less information from occluded regions in some samples and thus leads to a rise in RMSE.
- "Baseline+DAT+HA" means that the DAT module is placed before the HA module, which leads to a 3.9% decrease in MAE but the worst RMSE performance. This indicates that the order of DAT and HA modules affects model performance.

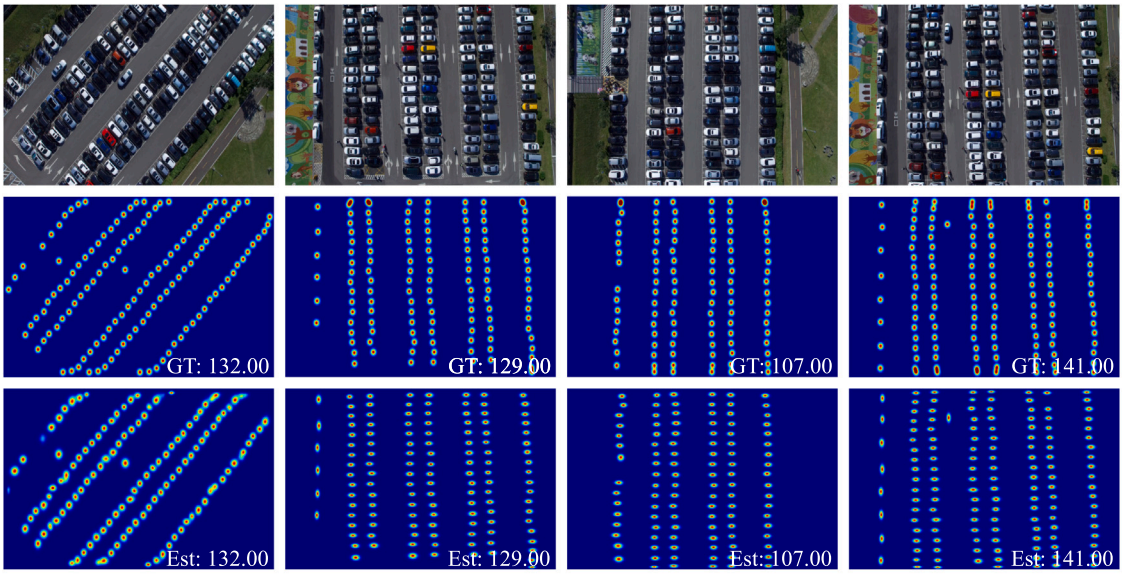



**Fig. 11.** Subjective results of CARPK dataset.

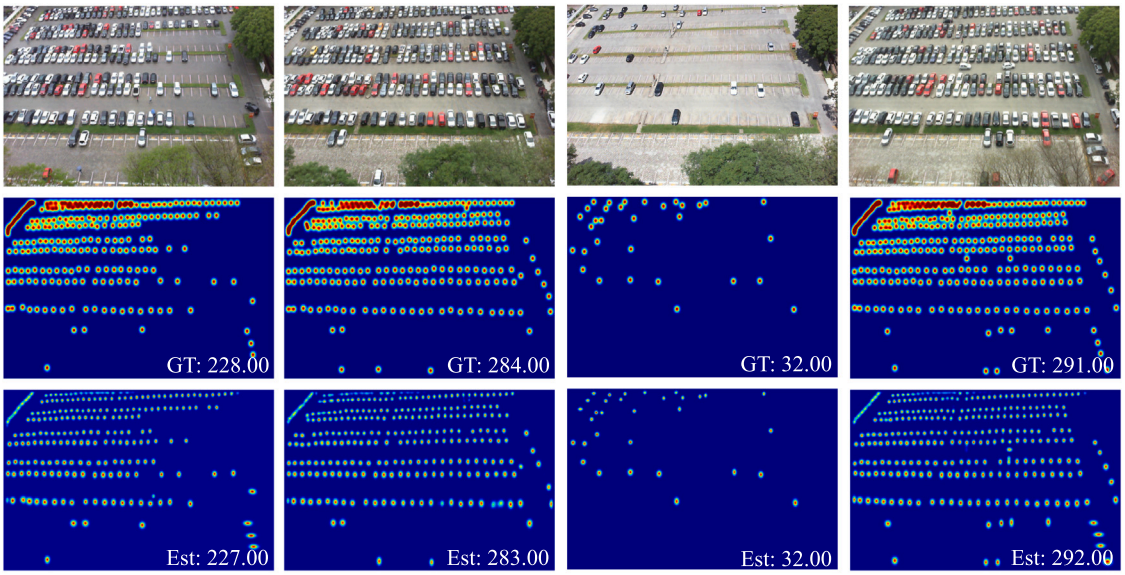



**Fig. 12.** Subjective results of PUCPR+ dataset.

- "Baseline+HA+DAT", *i.e.*, the proposed DLPTNet, performs the best, with an 8.5% decrease in MAE and a 9.3% decrease in RMSE. This demonstrates that the DAT module further mitigates the impact of occlusion on counting performance in crowded scenes, built upon the HA module.

**Ablation study on the number of local clients** To assess the impact of varying the number of clients on model performance, we conduct ablation experiments on the ShanghaiTech Part A dataset. The objective comparison results of the counting performance based on the number of local clients are shown in Table 8.

To investigate the impact of the number of local clients on counting performance, we employ a strategy of evenly distributing the training dataset. The rationale behind this approach is to analyze the trade-off between the decentralized nature of the framework and the resulting counting performance. Specifically, when there are $n$ local clients, we divided the original training dataset into $n$ equal parts, with each local client receiving a subset representing $1/n$ of the total training data. Meanwhile, the validation and test sets remain the same across all experiments. It is observed that as the number of clients increased, the performance gradually

**Table 7**
The impact of pivotal modules in DLPTNet on counting performance based on the ShanghaiTech Part A dataset. The best results are highlighted in **bold**.

| Methods | MAE | RMSE |
|---|---|---|
| Baseline | 63.8 | 105.8 |
| Baseline+HA | 60.8 | 100.6 |
| Baseline+DAT | 62.4 | 110.8 |
| Baseline+DAT+HA | 61.3 | 117.7 |
| Baseline+HA+DAT | **58.4** | **96.0** |

**Table 8**
Objective comparison results of counting performance based on the number of local clients on the ShanghaiTech Part A dataset. The best results are highlighted in **bold**.

| Methods | MAE | RMSE |
|---|---|---|
| Baseline | 63.8 | 105.8 |
| DLPTNet (n = 1) | **58.4** | **96.0** |
| DLPTNet (n = 2) | 63.1 | 104.0 |
| DLPTNet (n = 3) | 68.2 | 117.5 |

decreased. The reason for this decline is the even distribution of training data among the clients, which leads to a reduction in the number of training samples and an increased risk of overfitting. However, we find that when two clients participated in training together, compared with the baseline model, MAE and RMSE decreased by 1.0% and 1.7%, respectively. This indicates the effectiveness of the proposed DLPTNet.

## 5. Conclusion

To address the problems of occlusion in dense scenes and data privacy in crowd counting, we propose a privacy-aware crowd counting method termed DLPTNet in this paper. The DLPTNet adeptly balances counting accuracy and privacy concerns in crowd counting. It consists of two pivotal modules, the HA module and the DAT module. The HA module has a large perceptual range, and thus enhances its ability to perceive obscured areas around targets and extracts more extensive information from local regions to address occlusion issues in dense scenes. Additionally, we propose the DAT module, which builds upon the HA module to further address the occlusion issue in dense scenes. It leverages the Transformer to capture long-range dependencies between different regions and thus allows the model to gain a learning of the correlations and density distribution across various areas within the dense crowd at the global level. Experimental results demonstrate that by harnessing the capabilities of parallel Transformers and decentralized learning, the proposed DLPTNet achieves remarkable performance while safeguarding data privacy.

## Funding

There is no funding to support this manuscript.

## CRediT authorship contribution statement

**Jinyong Chen:** Conceptualization, Writing – original draft. **Mingliang Gao:** Conceptualization, Investigation, Supervision, Writing – review & editing, Project administration, Resources. **Qilei Li:** Conceptualization, Methodology, Validation. **Xiangyu Guo:** Software, Visualization. **Jianyong Wang:** Software, Writing – original draft. **Jing'an Cheng:** Data curation, Resources, Validation. **Xuening Xing:** Validation, Writing – review & editing.

## Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

## Data availability

Data will be made available on request.

## References


[1] A.B. Chan, Z.-S.J. Liang, N. Vasconcelos, Privacy preserving crowd monitoring: Counting people without people models or tracking, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, CVPR, IEEE, 2008, pp. 1–7.

[2] J. Shao, K. Kang, C. Change Loy, X. Wang, Deeply learned attributes for crowded scene understanding, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, CVPR, 2015, pp. 4657–4666.

[3] C. Zhang, Y. Xie, H. Bai, B. Yu, W. Li, Y. Gao, A survey on federated learning, Knowl.-Based Syst. 216 (2021) 106775.

[4] Q. Li, Z. Wen, Z. Wu, S. Hu, N. Wang, Y. Li, X. Liu, B. He, A survey on federated learning systems: Vision, hype and reality for data privacy and protection, IEEE Trans. Knowl. Data Eng. 35 (4) (2023) 3347–3366.

[5] I.S. Topkaya, H. Erdogan, F.M. Porikli, Counting people by clustering person detector outputs, in: 2014 11th IEEE International Conference on Advanced Video and Signal Based Surveillance, AVSS, 2014, pp. 313–318.

[6] S.D. Khan, S. Basalamah, Scale and density invariant head detection deep model for crowd counting in pedestrian crowds, Vis. Comput. 37 (8) (2021) 2127–2137.

[7] I. Ahmed, M. Anisetti, G. Jeon, An IoT-based human detection system for complex industrial environment with deep learning architectures and transfer learning, Int. J. Intell. Syst. 37 (12) (2022) 10249–10267.

[8] A.B. Chan, N. Vasconcelos, Bayesian poisson regression for crowd counting, in: Proceedings of the International Conference on Computer Vision, ICCV, IEEE, 2009, pp. 545–551.

[9] X. Tan, C. Tao, T. Ren, J. Tang, G. Wu, Crowd counting via multi-layer regression, in: Proceedings of the ACM International Conference on Multimedia, ACM MM, 2019, pp. 1907–1915.

[10] X. Liu, J. Yang, W. Ding, T. Wang, Z. Wang, J. Xiong, Adaptive mixture regression network with local counting map for crowd counting, in: Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XXIV 16, Springer, 2020, pp. 241–257.

[11] G. Gao, J. Gao, Q. Liu, Q. Wang, Y. Wang, Cnn-based density estimation and crowd counting: A survey, 2020, arXiv preprint arXiv:2003.12783.

[12] X. Guo, K. Song, M. Gao, W. Zhai, Q. Li, G. Jeon, Crowd counting in smart city via lightweight Ghost Attention Pyramid Network, Future Gener. Comput. Syst. 147 (2023) 328–338.

[13] W. Zhai, Q. Li, Y. Zhou, X. Li, J. Pan, G. Zou, M. Gao, DA2Net: a dual attention-aware network for robust crowd counting, Multimedia Syst. 29 (5) (2023) 3027–3040.

[14] Z. Fan, H. Zhang, Z. Zhang, G. Lu, Y. Zhang, Y. Wang, A survey of crowd counting and density estimation based on convolutional neural network, Neurocomputing 472 (2022) 224–251.

[15] M.A. Khan, H. Menouar, R. Hamila, Revisiting crowd counting: State-of-the-art, trends, and future perspectives, Image Vis. Comput. (2022) 104597.

[16] K. Han, Y. Wang, H. Chen, X. Chen, J. Guo, Z. Liu, Y. Tang, A. Xiao, C. Xu, Y. Xu, et al., A survey on vision transformer, IEEE Trans. Pattern Anal. Mach. Intell. 45 (1) (2022) 87–110.

[17] S. Khan, M. Naseer, M. Hayat, S.W. Zamir, F.S. Khan, M. Shah, Transformers in vision: A survey, ACM Comput. Surv. (CSUR) 54 (10s) (2022) 1–41.

[18] D. Liang, X. Chen, W. Xu, Y. Zhou, X. Bai, Transcrowd: weakly-supervised crowd counting with transformers, Sci. China Inf. Sci. 65 (6) (2022) 160104.

[19] Z. Liu, W. Wu, Y. Tan, G. Zhang, RGB-T multi-modal crowd counting based on transformer, 2023, arXiv preprint arXiv:2301.03033.

[20] N.H. Tran, T.D. Huy, S.T. Duong, P. Nguyen, D.H. Hung, C.D.T. Nguyen, T. Bui, S.Q. Truong, J. VinBrain, Improving local features with relevant spatial information by vision transformer for crowd counting, in: British Machine Vision Conference, 2022, pp. 1–15.

[21] B. McMahan, E. Moore, D. Ramage, S. Hampson, B.A. y Arcas, Communication-efficient learning of deep networks from decentralized data, in: Artificial Intelligence and Statistics, PMLR, 2017, pp. 1273–1282.

[22] R. Senthilkumar, S. Ritika, M. Manikandan, B. Shyam, Crowd counting using federated learning and domain adaptation, in: International Conference on Information, Communication and Computing Technology, Springer, 2022, pp. 97–111.

[23] Y. Pang, Z. Ni, X. Zhong, Federated learning for crowd counting in smart surveillance systems, IEEE Internet Things J. (2023).

[24] M.M.K. Tan, Indoor UAV Crowd Investigation Part 2 Via Computer Vision Applications and Federated Learning Methods, Nanyang Technological University, 2021.

[25] Y. Jiang, R. Cong, C. Shu, A. Yang, Z. Zhao, G. Min, Federated learning based mobile crowd sensing with unreliable user data, in: 2020 IEEE 22nd International Conference on High Performance Computing and Communications; IEEE 18th International Conference on Smart City; IEEE 6th International Conference on Data Science and Systems, HPCC/SmartCity/DSS, IEEE, 2020, pp. 320–327.

[26] X. Guo, M. Anisetti, M. Gao, G. Jeon, Object counting in remote sensing via triple attention and scale-aware network, Remote Sens. 14 (24) (2022) 6363.

[27] W. Zhai, M. Gao, A. Souri, Q. Li, X. Guo, J. Shang, G. Zou, An attentive hierarchy ConvNet for crowd counting in smart city, Cluster Comput. 26 (2) (2023) 1099–1111.

[28] X. Guo, M. Gao, W. Zhai, Q. Li, K.H. Kim, G. Jeon, Dense attention fusion network for object counting in IoT system, Mob. Netw. Appl. (2023) 1–10.

[29] W. Zhai, M. Gao, Q. Li, G. Jeon, M. Anisetti, FPANet: feature pyramid attention network for crowd counting, Appl. Intell. (2023) 1–18.

[30] A. Vaswani, P. Ramachandran, A. Srinivas, N. Parmar, B. Hechtman, J. Shlens, Scaling local self-attention for parameter efficient visual backbones, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, CVPR, 2021, pp. 12894–12904.

[31] D. Liang, W. Xu, Y. Zhu, Y. Zhou, Focal inverse distance transform maps for crowd localization, IEEE Trans. Multimed. 25 (2022) 6040–6052.

[32] Y. Zhang, D. Zhou, S. Chen, S. Gao, Y. Ma, Single-image crowd counting via multi-column convolutional neural network, CVPR, 2016, pp. 589–597.

[33] H. Idrees, M. Tayyab, K. Athrey, D. Zhang, S. Al-Maadeed, N. Rajpoot, M. Shah, Composition loss for counting, density map estimation and localization in dense crowds, in: Proceedings of the European Conference on Computer Vision, ECCV, 2018, pp. 532–546.

[34] V.A. Sindagi, R. Yasarla, V.M. Patel, Jhu-crowd++: Large-scale crowd counting dataset and a benchmark method, IEEE Trans. Pattern Anal. Mach. Intell. 44 (5) (2020) 2594–2609.

[35] Q. Wang, J. Gao, W. Lin, X. Li, NWPU-crowd: A large-scale benchmark for crowd counting and localization, IEEE Trans. Pattern Anal. Mach. Intell. 43 (6) (2020) 2141–2149.

[36] M.-R. Hsieh, Y.-L. Lin, W.H. Hsu, Drone-based object counting by Spatially Regularized Regional proposal network, in: Proceedings of the International Conference on Computer Vision, ICCV, ICCV, 2017, pp. 4165–4173.

[37] V. Sindagi, V. Patel, CNN-based cascaded multi-task learning of high-level prior and density estimation for crowd counting, in: Proceedings of the IEEE International Conference on Advanced Video and Signal Based Surveillance, AVSS, 2017, pp. 1–6.

[38] Q. Wang, T. Han, J. Gao, Y. Yuan, Neuron linear transformation: Modeling the domain shift for crowd counting, IEEE Trans. Neural Netw. Learn. Syst. 33 (8) (2021) 3238–3250.

[39] D. Babu Sam, S. Surya, R. Venkatesh Babu, Switching convolutional neural network for crowd counting, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, CVPR, CVPR, 2017, pp. 5744–5752.

[40] X. Shi, X. Li, C. Wu, S. Kong, J. Yang, L. He, A real-time deep network for crowd counting, in: ICASSP 2020-2020 IEEE International Conference on Acoustics, Speech and Signal Processing, ICASSP, IEEE, 2020, pp. 2328–2332.

[41] S. Amirgholipour, X. He, W. Jia, D. Wang, M. Zeibots, A-CCNN: adaptive CCNN for density estimation and crowd counting, in: 2018 25th IEEE International Conference on Image Processing, ICIP, IEEE, 2018, pp. 948–952.

[42] L. Zhang, M. Shi, Q. Chen, Crowd counting via scale-adaptive convolutional neural network, in: 2018 IEEE Winter Conference on Applications of Computer Vision, WACV, IEEE, 2018, pp. 1113–1121.
[43] Y. Lei, Y. Liu, P. Zhang, L. Liu, Towards using count-level weak supervision for crowd counting, Pattern Recognit. 109 (2021) 107616.
[44] M. Zhu, X. Wang, J. Tang, N. Wang, L. Qu, Attentive multi-stage convolutional neural network for crowd counting, Pattern Recognit. Lett. 135 (2020) 279–285.
[45] J. Gao, Q. Wang, X. Li, PCC net: Perspective crowd counting via spatial convolutional network, IEEE Trans. Circuits Syst. Video Technol. 30 (2020) 3486–3498.
[46] L. Zhang, Z. Shi, M.-M. Cheng, Y. Liu, J.-W. Bian, J.T. Zhou, G. Zheng, Z. Zeng, Nonlinear regression via deep negative correlation learning, IEEE Trans. Pattern Anal. Mach. Intell. 43 (3) (2019) 982–998.
[47] D.B. Sam, N.N. Sajjan, R.V. Babu, M. Srinivasan, Divide and grow: Capturing huge diversity in crowd images with incrementally growing cnn, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, CVPR, 2018, pp. 3618–3626.
[48] Z. Zou, Y. Cheng, X. Qu, S. Ji, X. Guo, P. Zhou, Attend to count: Crowd counting with adaptive capacity multi-scale CNNs, Neurocomputing 367 (2019) 75–83.
[49] Y. Li, X. Zhang, D. Chen, Csrnet: Dilated convolutional neural networks for understanding the highly congested scenes, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, CVPR, 2018, pp. 1091–1100.
[50] J. Gao, Q. Wang, Y. Yuan, SCAR: Spatial-/channel-wise attention regression networks for crowd counting, Neurocomputing 363 (2019) 1–8.
[51] L. Liu, J. Jiang, W. Jia, S. Amirgholipour, Y. Wang, M. Zeibots, X. He, Denet: A universal network for counting crowd with varying densities and scales, IEEE Trans. Multimed. 23 (2020) 1060–1068.
[52] C. Zhang, H. Li, X. Wang, X. Yang, Cross-scene crowd counting via deep convolutional neural networks, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, CVPR, 2015, pp. 833–841.
[53] H. Idrees, I. Saleemi, C. Seibert, M. Shah, Multi-source multi-scale counting in extremely dense crowd images, in: CVPR, CVPR, 2013, pp. 2547–2554.
[54] D.B. Sam, S.V. Peri, M.N. Sundararaman, A. Kamath, R.V. Babu, Locate, size, and count: accurately resolving people in dense crowds via detection, IEEE Trans. Pattern Anal. Mach. Intell. 43 (8) (2020) 2739–2751.
[55] X. Cao, Z. Wang, Y. Zhao, F. Su, Scale aggregation network for accurate and efficient crowd counting, in: Proceedings of the European Conference on Computer Vision, ECCV, 2018, pp. 734–750.
[56] C. Liu, X. Weng, Y. Mu, Recurrent attentive zooming for joint crowd counting and precise localization, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, CVPR, 2019, pp. 1217–1226.
[57] J. Wan, A. Chan, Adaptive density map generation for crowd counting, in: Proceedings of the International Conference on Computer Vision, ICCV, 2019, pp. 1130–1139.
[58] L. Wen, D. Du, P. Zhu, Q. Hu, Q. Wang, L. Bo, S. Lyu, Detection, tracking, and counting meets drones in crowds: A benchmark, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2021, pp. 7812–7821.
[59] W. Zhai, M. Gao, M. Anisetti, Q. Li, S. Jeon, J. Pan, Group-split attention network for crowd counting, J. Electron. Imaging 31 (4) (2022) 041214.
[60] Y. Meng, H. Zhang, Y. Zhao, X. Yang, X. Qian, X. Huang, Y. Zheng, Spatial uncertainty-aware semi-supervised crowd counting, in: Proceedings of the International Conference on Computer Vision, ICCV, 2021, pp. 15529–15539.
[61] S. Abousamra, M. Hoai, D. Samaras, C. Chen, Localization in the crowd with topological constraints, in: Proceedings of the AAAI Conference on Artificial Intelligence, Vol. 35, 2021, pp. 872–881.
[62] Q. Wang, J. Gao, W. Lin, Y. Yuan, Learning from synthetic data for crowd counting in the wild, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, CVPR, 2019, pp. 8190–8199.
[63] Z. Ma, X. Wei, X. Hong, Y. Gong, Bayesian loss for crowd count estimation with point supervision, in: Proceedings of the International Conference on Computer Vision, ICCV, 2019, pp. 6141–6150.
[64] G. Gao, Q. Liu, Z. Hu, L. Li, Q. Wen, Y. Wang, PSGCNet: A pyramidal scale and global context guided network for dense object counting in remote-sensing images, IEEE Trans. Geosci. Remote Sens. 60 (2022) 1–12.
[65] J. Redmon, S.K. Divvala, R.B. Girshick, A. Farhadi, You only look once: Unified, real-time object detection, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, CVPR, 2016, pp. 779–788.
[66] S. Ren, K. He, R.B. Girshick, J. Sun, Faster R-CNN: Towards real-time object detection with region proposal networks, TPAMI 39 (2015) 1137–1149.
[67] T. Stahl, S.L. Pintea, J.C.V. Gemert, Divide and count: Generic object counting by image divisions, IEEE Trans. Image Process. 28 (2019) 1035–1044.
[68] W. Liu, D. Anguelov, D. Erhan, C. Szegedy, S.E. Reed, C.-Y. Fu, A.C. Berg, SSD: Single shot MultiBox detector, in: Proceedings of the European Conference on Computer Vision, ECCV, ECCV, 2016, pp. 21–37.
[69] T.-Y. Lin, P. Goyal, R.B. Girshick, K. He, P. Dollár, Focal loss for dense object detection, IEEE Trans. Pattern Anal. Mach. Intell. 42 (2020) 318–327.
[70] T.N. Mundhenk, G. Konjevod, W.A. Sakla, K. Boakye, A large contextual dataset for classification, detection and counting of cars with deep learning, in: Proceedings of the European Conference on Computer Vision, ECCV, Springer, 2016, pp. 785–800.
[71] X. Yang, J. Yang, J. Yan, Y. Zhang, T. Zhang, Z. Guo, X. Sun, K. Fu, Scrdet: Towards more robust detection for small, cluttered and rotated objects, in: Proceedings of the International Conference on Computer Vision, ICCV, 2019, pp. 8232–8241.
[72] Z. Tian, C. Shen, H. Chen, T. He, Fcos: Fully convolutional one-stage object detection, in: Proceedings of the International Conference on Computer Vision, ICCV, 2019, pp. 9627–9636.