electronics

MDPI

Article

Improving MLP-Based Weakly Supervised Crowd-Counting Network via Scale Reasoning and Ranking

Ming Gao, Mingfang Deng, Huailin Zhao *, Yangjian Chen and Yongqi Chen

School of Electrical and Electronic Engineering, Shanghai Institute of Technology, Shanghai 201400, China; gaoming_one@163.com (M.G.); dengmingfang2021@163.com (M.D.); 236102102@mail.sit.edu.cn (Y.C.); cheny2495@gmail.com (Y.C.)

* Correspondence: tyoukr@163.com

Abstract: MLP-based weakly supervised crowd counting approaches have made significant advancements over the past few years. However, owing to the limited datasets, the current MLP-based methods do not consider the problem of region-to-region dependency in the image. For this, we propose a weakly supervised method termed SR2. SR2 consists of three parts: scale-reasoning module, scale-ranking module, and regression branch. In particular, the scale-reasoning module extracts and fuses the region-to-region dependency in the image and multiple scale feature, then sends the fused features to the regression branch to obtain estimated counts; the scale-ranking module is used to understand the internal information of the image better and expand the datasets efficiently, which will help to improve the accuracy of the estimated counts in the regression branch. We conducted extensive experiments on four benchmark datasets. The final results showed that our approach has better and higher competing counting performance with respect to other weakly supervised counting networks and with respect to some popular fully supervised counting networks.

Keywords: weakly supervised counting; MLP; graph neural networks; ranking mechanism

check for updates

Citation: Gao, M.; Deng, M.; Zhao, H.; Chen, Y.; Chen, Y. Improving MLP-Based Weakly Supervised Crowd-Counting Network via Scale Reasoning and Ranking. *Electronics* **2024**, *13*, 471. https://doi.org/10.3390/electronics13030471

Academic Editor: Ping-Feng Pai

Received: 19 December 2023
Revised: 10 January 2024
Accepted: 18 January 2024
Published: 23 January 2024

1. Introduction

Crowd counting is a classical computer vision task employed to generate an estimated count in an image or a dynamic video sequence. In scenes such as tourist attractions and public gatherings, overcrowding can cause crowd crushing, blockages, and even stampedes. Thus, accurately determining the number of people in a crowd in an image or video has become an increasingly important application [1].

Because of heavy occlusions and multiscale variations in heads in crowds, crowd counting is a challenging task. In recent years, there have been many algorithms [2,3] for crowd counting, the density-estimation methods being the mainstream ones. For example, the MCNN [2] adopted three columns with different receptive fields to capture the scale variations of heads in crowds. CSRNet [3] adopted dilated convolutional layers with different dilated rates, which can capture useful information about the context to better distinguish between the foreground and the background. However, these methods require fine-grained point-level annotations, which require much human effort.

In order to reduce the dependence on fine-grained point-level annotations, some works have made full use of weakly supervised learning paradigms to calculate the number of people in an image. For example, Liang et al. [4] proposed Transcrowd, which explored the potential of the vision Transformer [5] for weakly supervised crowd counting. Savner et al. [6] adopted the pyramid vision Transformer [7] with count-level annotations to capture the multiscale information. However, Transformer requires a great amount of computational resources, making it challenging for real applications.

Since 2020, some scholars have explored more diverse models beyond convolutional neural networks and Transformer models. Recently, the re-emergence of multilayer perceptrons (MLPs) has achieved excellent classification performance, benefiting from the inherent

advantages of the underlying fully connected layers: much more global receptive fields than the CNN model and simpler Self-Attention layers than in the Transformer model. Meanwhile, the potential of the MLP model in regression tasks has not been fully explored. Wang et al. [8] proposed a weakly supervised crowd-counting network, CrowdMLP, by constructing a multi-particle MLP regulator to capture global information.

However, the CrowdMLP network has the following limitations: (1) CrowdMLP ignores the dependencies between regions. It uses multiple columns to predict different densities of regions independently. Effectively, different densities of areas are correlated in the scenes. In congested scenarios, there are certain configuration rules for an approximately constant density of crowds per square meter in the physical world. This configuration rule is a constant approximate change in density along the direction away from the camera. The density distribution in many scenes (e.g., streets, squares, stadiums, etc.) is governed by the configuration rules. These rules can be used to further improve the capabilities of crowd counting. (2) CrowdMLP [8] introduces the ranking mechanism and designs auxiliary branches to improve the accuracy of predicting counts, but these auxiliary branches are cut in the original images and put into the basic network, which is equivalent to computing the model three times in parallel for each training, which inadvertently increases the computational load of the network.

To overcome the above two issues, we designed a scale-reasoning module based on the graph convolutional network and a scale-ranking module based on the ranking mechanism, respectively. The role of the scale-reasoning module is to capture the dependencies between the regions in the image, and the role of the scale-ranking module is to solve the overfitting of complex networks due to limited datasets. For the scale-reasoning module, we considered graph neural networks [9], which have been demonstrated to be a suitable way for relational modeling and inference, which contain nodes and edges, where the nodes stand for the pixels in the image and the edges represent the tightness of the relationship between the nodes.

For the scale-ranking module, instead of using the ranking mechanism directly to process the original images, we directly applied the ranking mechanism to the final extracted feature maps. We used the scale-ranking module as an auxiliary branch to improve the accuracy of the estimated counts in the images. Based on the proposed scale reasoning and ranking modules, we propose a novel weakly supervised MLP-based crowd-counting network, termed SR2, which contains the scale-reasoning module, scale-ranking module, and regression branch. Finally, we performed extensive experiments on the popular crowd-counting datasets. Compared with other outstanding approaches, our proposed SR2 achieved promising counting results on ShanghaiTech [2], UCF-QNRF [10], JHU-CROWD++ [11], and NWPU-Crowd [12]. For more details, please visit our code which has been released at https://github.com/MingfangDeng/GRMlpCrowd-main (accessed on 18 December 2022).

Broadly speaking, our efforts can be summarized as the following three aspects:

- We designed a scale-reasoning module fusing the information of the region-to-region dependencies that can capture the multiscale information by the pyramid nodes.
- We propose a scale-ranking module fusing the tightness of the whole image with the selected region and reducing the computational load.
- SR2 provides higher accuracy compared to either the fully supervised counting method or the weakly supervised counting method.

2. Related Works

2.1. Counting with Fully Supervised Paradigm

Convolution neural networks: With the blisteringly fast growth of deep learning for computer vision [5,13–16], a number of methods [2,17–27] based on the CNN have been introduced to generate the predicted density maps that can be regressed to obtain the estimated counts. Among these methods based on the CNN, Zhang et al. proposed a multiple-column convolutional neural network with three branches containing convolutional kernels of

different sizes, namely the MCNN [2]. However, the MCNN does not effectively capture multiscale information in the image. The work of [17] proposed a density classifier called Switch CNN [17], which was used to select the Optimum Regressor from three different branches adaptively to complement the inadequacy of the MCNN. However, these models have the problem of a large number of parameters, so some scholars have lightened the models. For example, PDDNet [22] adopted a lightweight network with different dilated convolutional layers to capture different scales. To address the scale variation and complex backgrounds effectively, Sun et al. [23] proposed a novel Multi-Scale Guided Self-Attention network that utilizes Self-Attention mechanisms to capture multi-scale contextual information for crowd counting. Dong et al. [24] proposed a multi-scale dilated convolution network based on crowd density map estimation. They used dilated convolution to expand the receptive field and extract high-level semantic information. Yan et al. [25] proposed to predict the density map at one resolution, but measure the density map at multiple resolutions. They formulated the crowd counting task as a probability maximization problem and derived the optimization loss for the deep learning model by maximizing the posterior probability. Liu et al. [26] designed a supervision target reassignment strategy for training to reduce ranking inconsistency and proposed an anchor pyramid scheme to adaptively determine the anchor density in each image region. Ge et al. [27] propose a neural Attention Learning approach (NEAL), which helps the RPN attend to objects and enables the classifier to pay more attention to the premier positive samples.

Transformer architecture networks: To address the limitation of the convolution kernel, many scholars tried to utilize the Transformer models for counting tasks, which can obtain global information. For example, CCTrans [28] captured multiscale information based on Twins [29] to obtain the sharp density maps. Gao et al. [30] combined a Swin Transformer encoder [16] and an FPN decoder to generate more-accurate density maps. Lin et al. [31] proposed a novel Self-Attention that replaces the Self-Attention in the Vision Transformer [5] to generate more-accurate density maps. However, both the CNN models and Transformer models ignore the region-to-region dependencies in the images, and such dependencies can improve the counting accuracy.

Graph neural networks: To overcome the limitations of the CNN and Transformer models, researchers have introduced graph convolutional networks (GCNs) [9] into crowd counting. GCNs exhibit robust performance in modeling and inferring relationships among different regions, effectively addressing the challenges associated with region-to-region dependencies in crowd counting tasks. Luo et al. [32] proposed a HyGnn, which incorporates a hybrid graph that jointly represents task-specific feature maps at different scales as nodes, aiming to capture the multiscale information. Compared with HyGnn, which uses the different sizes of Global Average Pooling (GAP) operations, we designed a novel module (scale-reasoning module) using different dilated rates to capture multiscale information rather than the GAP operations, which lose the detailed information of the images. However, point-level annotations are both time-consuming and laborious.

2.2. Counting with Weakly Supervised Paradigm

Convolution neural networks: Lie et al. [33] presented the weakly supervised crowd counting model MATT, which has a few point-level annotations and a large number of count-level annotations. Yang et al. [34] proposed a sorting network that directly returns to counts without point-level annotations. However, CNN architectures have contextual limitations due to the shape of the convolution kernel, which can impact the ability to distinguish the foreground and impact the accuracy of counting.

Transformer architecture networks: With the development of the Transformers, many researchers have found that the Transformer model is a good solution to the limitations on the size of the convolutional receptive fields. Aiming at better extraction of global information, some researchers have extended the Transformer architecture into the field of weakly supervised counting. For instance, the work of [4] employed the ViT [5] model and eventually acquired the estimated counts using an operation of GAP. The work of [35] used

the Swin Transformer [16] combined with the convolutional network and directly summed the generated feature vectors to obtain the estimated counts. The work of [6] adopted PVT [7] as the basic network, obtained multiscale information by extracting and fusing its middle information, and directly obtained the estimated counts by GAP. However, there are some problems such as the slowness of the Transformer models in processing images and increasing the computational load.

MLP architecture networks: In order to solve the problems of the Transformer models and capture the global information in the images, recently, Wang et al. [36] triggered a series of studies utilizing the inherent advantages of MLPs. Although MLPs perform well in image classification, their potential in regression remains to be explored. However, only a few researchers have proposed the methods based on MLPs. Wang et al. [8] proposed the multi-branch MLP encoder to add information on token embeddings. This method introduces the ranking mechanism and designs auxiliary branches to improve the accuracy of predicting counts, but these auxiliary branches are cut in the original images and put into the basic network, which is equivalent to computing the model three times in parallel for each training, which inadvertently increases the computational load.

3. Method

SR2 consists of the MLP encoder, scale-reasoning module, scale-ranking module, and the regression branch, as illustrated in Figure 1. Specifically, given an imported image, it is firstly fed into the MLP encoder, which has a tokenizer, which can extract the initial features, the convolution stage, which improves the connectivity of the space, and two MLP stages for extracting global features FM. Then, the features FM are fed into the scale-reasoning module. The scale-reasoning module adopts the different dilated rates to capture the multiscale information. Furthermore, the scale-reasoning module has a graph neural network that contains nodes and edges, where the nodes stand for the pixels in the image and the edges represent the tightness of the relationship between the nodes. Thus, the scale-reasoning module can fuse the region-to-region dependency in the images. Then, the fused feature maps are sent to the regression branch and scale-ranking module, respectively.

The regression branch can obtain the estimated counts by GAP. The scale-ranking module adopts the ranking mechanism, selecting a random resolution rectangular region (I_1) from the input features FM, and obtains sub-rectangular regions (I_2, I_3) at a certain downsampling rate, followed by the GAP to obtain the part estimated counts, as illustrated in Figure 1. Finally, the scale-ranking module uses the ranking loss to improve the accuracy of the estimated counts in the regression branch.

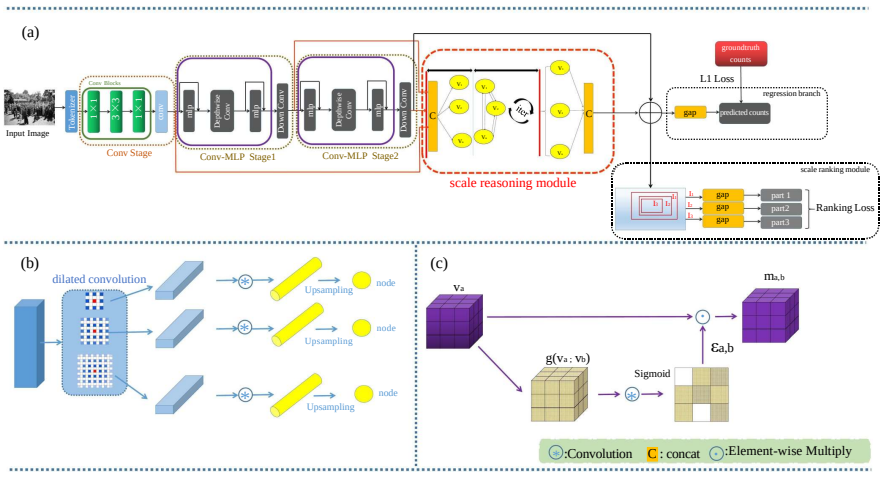

Figure 1. (**a**) The framework of SR2. SR2 has four parts, which are the MLP encoder, scale-reasoning module, scale-ranking module, and regression branch. The regression branch uses the L_1 loss, and the ranking branch uses the rank loss. (**b**) Node generation in scale-reasoning module. (**c**) The information update of the edges in scale-reasoning module.

3.1. Problem Formulation

We define the crowd-counting method of weak supervision according to [4,33]. The input image I is putinto the counting networks, and the acquired feature map is returned to the estimated counts $\hat{C}_i$ using the GAP method. More specifically, the estimated counts $\hat{C}_i$ are modeled as follows:

$$\begin{aligned} Fm &= \mathcal{F}(I_i), \\ \hat{C}_i &= Pool(Fm), \end{aligned} \tag{1}$$

where $\hat{C}_i$ represents the estimated counts of the i-th input image I_i. $\mathcal{F}$ stands for the crowd-counting network. $Pool(\cdot)$ represents the GAP. Fm is the feature map following the counting network.

The counting method learns the drift among the estimated counts and the ground truth counts of the i-th image. We opted for the $\mathcal{L}_1$ loss to improve the precision of the crowd-counting network. The $\mathcal{L}_1$ is defined as:

$$\mathcal{L}_1 = \frac{1}{N}\sum_{i=1}^{N} |C_i - \hat{C}_i|, \tag{2}$$

where C_i is a ground truth count from the i-th input image.

3.2. The MLP Encoder

The MLP encoder consists of the convolution tokenizer, the convolution stage, the MLP stage, and the graph stage. Next, we will introduce these parts.

3.2.1. Convolution Tokenizer

Unlike other Transformer-based models, we adopted the convolution tokenizer to extract the preliminary feature maps. It comprises triple convolutional blocks. Each block comprises a 3×3 convolution and normalization and activation functions. The tokenizer is succeeded by the maximum pooling layer.

3.2.2. Convolution Stage

To decrease the calculations and optimize the connectivity of the space, we used a pure convolution stage after tokenization, which can produce a feature map. The convolution stage has three blocks, where each block consists of two 1×1 convolution layers having a 3×3 convolution layer in the center. As stated, we adopted the convolution stage to augment the connectivity of the space.

3.2.3. MLP Stage

Most current MLP-based models use the spatial MLP approach, which has constraints on the visual representation. Spatial MLP accepts only fixed-resolution inputs, making it difficult to transfer to downstream tasks. To reduce the input dimensionality constraints, we adopted an MLP stage, which contains only the channel MLP, which means allowing feature extraction between different channels. However, using only the channel MLP ignores the spatial information in the feature map. Therefore, we added a 3×3 depth-wise convolution in each MLP stage to compensate for the missing spatial interactions. Meanwhile, inspired by the use of the linear layer-based patch-merging method in the Swin Transformer to downsample the feature maps to obtain the overlap of the spatial information, we introduced the down convolution blocks, aiming to improve the regression accuracy by obtaining the overlap of spatial information in the dense population. The down convolution block uses a 3×3 convolutional layer with a stride of 2 to replace the Swin Transformer's patch merging. Compared with the Swin Transformer's patch merge, our down convolution block introduces only a few parameters.

3.3. Scale-Reasoning Module

Meanwhile, we designed the scale-reasoning module to capture the multiscale information and mixture of the region-to-region dependency in the images. In our approach, the scale-reasoning module is described as a function $f_\theta : X \to M$, with the arguments θ; the input space X reflects the space of the input images; the target M is the relation map, which contains the relationship of the heads and the background, specifically the input feature map $x \in X$, and we study the mapping function f_θ, which can deduce the relation map $m \in M$. We represent the extracted multiple scales features $\mathcal{V} = \{v_1, v_2, \ldots, v_n\}$ with a directed graph $\mathcal{G} = (\mathcal{V}, \varepsilon)$, where ε represents the edges among the $\mathcal{V}$. For each node v_i in $\mathcal{V}$, we learn its renewable representation, called $v_i^{(t)}$, by assembling the representations of its neighbors. Finally, the renewable features are merged to yield a final relation map.

Multiple scale nodes: Given the feature maps FM, we employed different dilated rates for the dilated convolution and, then, extracted multiple scale features characterizing the same modality (n scale) as the initial node by convolutional and interpolation layers, obtaining a total of $N = n$ nodes. Their original node denotations $v_i^{(0)} \in \mathbb{R}^{h \times w \times c}$ can be calculated as:

$$v_i^{(0)} = \mathcal{G}_{h \times w}(Conv(\mathcal{H}(FM; s_i))), \tag{3}$$

where $\mathcal{H}(\cdot; s_i)$ stands for the dilated convolution operation with a rate of s_i. $\mathcal{G}_{h \times w}(\cdot)$ denotes the interpolation operation to ensure that multiscale feature maps have the same dimensions $h \times w$.

$$\varepsilon_{a,b}^t = Conv(g(v_a^t; v_b^t)), \tag{4}$$

where v_a^t and v_b^t are nodes in $\mathcal{V}$. $g(\cdot;\cdot)$ represents a function that binds the node embeddings v_b^t and v_b^t. $\varepsilon_{a,b}^t$ is the edge between the node a and the node b in set ε.

The edges are updated by all its neighboring nodes. The edge information $i_{a,b}^{(t)}$ is passed from all neighboring nodes v_a and v_b, and the edge information function $I(\cdot;\cdot)$:

$$i_{a,b}^{(t)} = I(v_a^{(t-1)}; \varepsilon_{a,b}^{(t-1)}) = sigmoid(\varepsilon_{a,b}^{(t-1)}) \cdot v_a^{(t-1)}, \tag{5}$$

where $i_{a,b}^{(t)}$ represents the edge information of node a and node b at t time.

Finally, we used the edge's information to upgrade the information of nodes. The node is updated by the following formula:

$$v_a^t = GRU(v_a^{(t-1)}, i_{a,b}^{(t-1)}), \tag{6}$$

where $GRU(\cdot,\cdot)$ denotes the Gated Recurrent Unit [37].

Finally, we adopted the merge and interpolate operation to generate the final relation map $\mathcal{M}$:

$$\mathcal{M} = \mathcal{U}_{H \times W}(\mathcal{O}(\mathcal{F}_{merge}\{\mathcal{U}(v_i^{(t)})_{i=1}^n\})), \tag{7}$$

where $\mathcal{F}_{merge}(\cdot)$ is the merge function, which comprises a connected layer after a 3×3 convolutional layer. $\mathcal{O}(\cdot)$ indicates the readout function, which is used for mapping the learned representations. The $\mathcal{U}$ is used to resize the generated results.

3.4. Regression Branch

The regression branch has an operation called GAP. Specifically, we concatenated the relation map ($\mathcal{M}$) and the output features of the MLP stages ($\mathcal{J}$), followed by a regression head, which is the GAP operation to regress the estimated counts. The equation is as follows:

$$\hat{C}_i = Pool2D(\mathcal{M} + \mathcal{J}). \tag{8}$$

3.5. Scale-Ranking Module

The scale-ranking module is guided by the fact that a smaller region must have fewer crowd numbers than or equal crowd numbers as a larger region. This fact led us to understand the internal information of the image better and expand the datasets efficiently. We randomly selected the initial rectangle area I_1 in the feature map FM, followed by the downsampling ratio r to obtain the subrectangle area I_2 and I_3. Finally, we adopted the GAP to generate the part estimated counts. The equation is as follows:

$$\hat{C}_i^j = Pool2D(DS(I_1, r)), \tag{9}$$

where $DS(\cdot, r)$ stands for the subrectangle regions with the following downsampling rate r. The $\hat{C}_i^j$ is the part estimated counts of the j-th subrectangle areas (I_j) in the i-th images. I_1 is the first randomly selected region in the input image.

3.6. Loss Function

To train SR2, we directly improved the accuracy of the estimated counts by minimizing the gap between the estimated counts and the ground truth. Specifically, we used the L_1 loss and the rank loss in the regression branch and the ranking branch. The loss function $\mathcal{L}$ is shown as:

$$\mathcal{L} = \mathcal{L}_1 + \gamma \mathcal{L}_{rank}, \tag{10}$$

where the best experimentally analyzed γ is 0.7.

The rank loss $\mathcal{L}_{rank}$ is as follows:

$$\begin{aligned} \hat{c}(I_i) &= Pool2D(I_i), \\ \mathcal{L}_{rank} &= \sum_{j=1}^{S} \sum_{k=j+1}^{S} max(0, \hat{c}_i^j - \hat{c}_i^k + margin), \end{aligned} \tag{11}$$

where S represents the amount of the selected patches.

4. Experiment

In this part, we first present the execution details and the setting of the experiment, and we utilized the popular crowd counting datasets. Meanwhile, we measured our approach against other superior approaches. Lastly, we carried out ablation tests to validate the efficiency and effectiveness of every component included in our approach.

4.1. Execution Details

We applied the Adam optimizer, which was trained for 1000 epochs. We configured the batch size as 8, the weight decay as 1×10^{-4}, and the learning rate as 1×10^{-5}, and with more than 300 epochs of training, the rate of learning was decreased to 0.1-times the original learning rate. Meanwhile, the weights pre-trained on the ImageNet dataset were applied to initialize the MLP. Furthermore, the s_i in the regression branch was (4, 8, 16), and the downsampling ratio r was set to 0.75 in the ranking branch. The margin we set was 0.03 in the rank loss. Lastly, it was implemented on a single NVIDIA RTX 3060Ti GPU (Santa Clara, CA, USA) with the Pytorch framework.

4.2. The Datasets Used

ShanghaiTech [2] is grouped into two segments, which are ShanghaiTechA, which comprises 300 images used for training and 182 images for testing, and ShanghaiTechB, which has 400 images used for training and 316 images used for testing.

UCF-QNRF [10] has one million annotations of 1535 images. The range of counts is extensive, spanning from 49 to 12,865. Additionally, it involves 1201 images for training and 334 images for testing.

NWPU-Crowd [12] is a dataset that is enormous and challenging. It includes 5109 images, and the number of instances is 2,133,375 with detailed annotations. Furthermore, the

dataset was stochastically separated into two segments: the training dataset and the test dataset, which comprised 3,109,500 and 1500 images, respectively.

JHU-CROWD++ [11] consists of 2722 images used for training, 500 images used for estimating, and 1600 images for testing from a wide range of scenes. The total number of people in each image varies from 0 to 25,791.

EoCo [38] comprises two parts, Part A and Part B, and includes a total of 6885 images, with 2859 images in Part A and 4026 images in Part B. Part A is divided into six classes: person, jujube, cherry, tulip, chicken, and vehicle. All samples in the dataset are sourced from public datasets or competitions and are classified into four categories: face, wheat, person (ShanghaiTech Part B), and penguin.

4.3. Evaluate Metrics

We opted for the mean absolute error along with the mean-squared error as evaluation metrics to assess the counting results of our approach:

$$MAE = \frac{1}{N}\sum_{i=1}^{N}|C_i - \hat{C}_i|, MSE = \sqrt{\frac{1}{N}\sum_{i=1}^{N}|\hat{C}_i - C_i|^2}, \tag{12}$$

where N is the amount of testing images and $\hat{C}_i$ and C_i are the predicted and ground truth count of the i-th image, respectively.

4.4. Compared Crowd Counts

We carried out an extensive experiment on four prevailing datasets [10–12] to verify the usefulness of the proposed approach. In this part, we compare with previous state-of-the-art methods. Finally, we demonstrate the ablation studies to verify the effectiveness of the different parts of our approach.

Comparing the counting methods with full supervision: Specifically, on the ShanghaiTechA dataset, our approach improved by 6.5% on the MAE and 14.1% on the MSE with respect to CSRNet, which benefited from the ranking branch to understand more inner information. On the ShanghaiTechB dataset, our approach improved by 20.8% on the MAE and 12.5% on the MSE with respect to CSRNet and improved by 5% on the MAE and 6.8% on the MSE compared with BL, which benefited from the MLP architecture and could obtain the global information. The specifics of the ShanghaiTech datasets are shown in Table 1.For the QNRF dataset, we found that our approach improved by 23.3% on the MAE and 16.5% on the MSE with respect to CSRNet and benefited from the the MLP architecture, which could acquire more global information and the pyramid pooling in the graph stage. The results of the QNRF dataset are shown in Table 2. For the NWPU dataset, our approach improved by 4.6% on the MAE with respect to CSRNet and improved by 4.3% on the MSE with respect to PCC-Net-VGG, which benefited from the MLP architecture and the ranking branch. Besides, our approach improved by 8.9% on the MAE and 0.5% on the MSE with respect to C3F-VGG on the NWPU dataset.

Comparing the counting methods with weak supervision: Specifically, on the ShanghaiTechA dataset, our approach improved by 8.3% on the MAE and 15.7% on the MSE with respect to MATT due to the graph stage, capturing more information between the heads and the background. Our approach also improved by 5.1% on the MAE and 7.1% on the MSE with respect to Transcrowd-GAP, which benefited from the convolution tokenizer being replaced by the traditional tokenizer, which could acquire the global information.For the QNRF dataset, our approach improved by 3.3% on the MAE and 5.8% on the MSE with respect to Transcrowd [4]. The specifics of the QNRF dataset can be seen in Table 2. Furthermore, we also validated the effectiveness of our approach on the JHU-Crowd++ dataset and the NWPU dataset, as illustrated in Tables 3 and 4. For the JHU-Crowd++ dataset, our approach improved by 6.2% on the MAE and improved by 8.0% on the MSE with respect to Transcrowd [4]. For the NWPU dataset, our approach improved by 3.3% on the MAE and 5.6% on the MSE compared to the Transcrowd [4].For the NWPU dataset, our

approach improved by 1.5% on the MAE and 5.8% on the MSE with respect to Transcrowd-Token. For the JHU-Crowd++ dataset, our approach improved by 6.2% on the MAE and 8.0% on the MSE with respect to Transcrowd-Token. To verify the generalization ability of our model, we conducted a validation on the EoCo dataset, and the performance on this dataset was also very excellent, as shown in Table 5. For the cherry class, our approach improved by 43.31% on the MAE and 27.23% on the MSE with respect to the MCNN and improved by 14.55% on the MAE and 13.89% on the MSE with respect to CSRNet. As shown in Table 5, we also demonstrate that counting performance was comparable to the popular fully supervised networks on other classes of the EoCo dataset.

Table 1. Comparison (MAE and MSE) of quantitatively different crowd-counting methods on the ShanghaiTech dataset. The localization of the training labels denotes point-level annotations, and the crowd number of the training labels stands for the count-level annotations. The bold values of the MAE and MSE represent the best performance.

Methods	Publish	Training Label		Part A		Part B	
		Localization	Crowd Number	MAE	MSE	MAE	MSE
MCNN [2]	CVPR16	✓	✓	110.1	174.0	27.1	51.3
CSRNet [3]	CVPR18	✓	✓	68.4	116.0	10.7	16.1
BL [39]	ICCV19	✓	✓	62.8	101.8	8.0	13.1
S3 [40]	IJCAI21	✓	✓	57.1	97.3	8.4	12.7
UOT [41]	AAAI21	✓	✓	58.1	92.5	7.7	12.5
MATT [33]	PR21	×	✓	69.7	118.2	10.6	19.9
Transcrowd-Token [4]	SCIS22	×	✓	69.7	118.2	10.6	19.9
Transcrowd-GAP [4]	SCIS22	×	✓	67.4	107.2	9.4	16.3
SR2 (ours)	-	×	✓	**63.9**	**99.6**	**8.4**	**14.0**

Table 2. Comparison (MAE and MSE) of quantitatively different crowd-counting methods on the QNRF dataset. The bold values of the MAE and MSE represent the best performance.

Methods	Publish	Training Label		QNRF	
		Localization	Crowd Number	MAE	MSE
MCNN [2]	CVPR16	✓	✓	277.1	426.1
CSRNet [3]	CVPR18	✓	✓	124.1	196.2
BL [39]	ICCV19	✓	✓	88.7	154.8
S3 [40]	IJCAI21	✓	✓	80.6	139.8
UOT [41]	AAAI21	✓	✓	83.3	142.3
MATT [33]	PR21	×	✓	98.9	176.1
Transcrowd-Token [4]	SCIS22	×	✓	98.0	175.1
Transcrowd-GAP [4]	SCIS22	×	✓	97.3	168.4
SR2 (ours)	-	×	✓	**96.5**	**153.1**

Moreover, we also show the results of the training loss value with the number of training epochs on ShanghaiTech Part A in Figure 2. As can be seen, our model had rapid convergence within the first 100 epochs of the training phase and consistently stayed within the designated convergence interval.

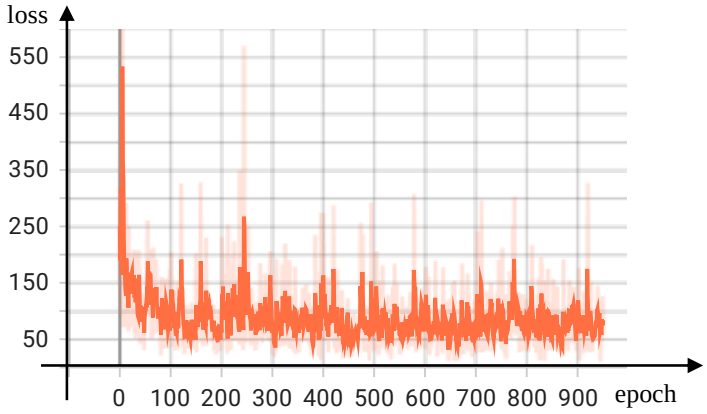

Figure 2. Graph of the training loss function in relation to the number of training epochs.

Table 3. Comparisons of the counting results on NWPU-Crowd. The bold values of the MAE and MSE represent the best performance.

Methods	Publish	Training Label		Testing Set	
		Localization	Crowd Number	MAE	MSE
C3F-VGG [42]	Tech19	✓	✓	127.1	439.5
CSRNet [3]	CVPR18	✓	✓	121.3	387.8
PCC-Net-VGG	CVPR19	✓	✓	112.2	457.1
CAN [43]	CVPR19	✓	✓	106.1	386.6
SFCN [44]	CVPR19	✓	✓	105.6	424.2
BL [39]	ICCV19	✓	✓	105.4	454.2
KDMG [45]	PAMI20	✓	✓	100.6	415.6
NoisyCC [46]	NeuralPS20	✓	✓	96.5	534.1
DM-Count [47]	NeuralPS20	✓	✓	88.4	388.6
UOT [41]	AAAI21	✓	✓	83.6	346.8
S3 [40]	IJCAI21	✓	✓	87.9	387.6
Transcrowd-Token [4]	SCIS22	×	✓	119.6	463.9
Transcrowd-GAP [4]	SCIS22	×	✓	117.7	451.0
SR2 (ours)	-	×	✓	**115.7**	**437.2**

Table 4. Comparison (MAE and MSE) of quantitatively different crowd-counting methods on the JHU-Crowd++ dataset. The bold values of the MAE and MSE represent the best performance.

Methods	Publish	Training Label		Testing Set	
		Location	Crowd Number	MAE	MSE
MCNN [2]	CVPR16	✓	✓	188.8	483.5
CMTL [48]	AVSS17	✓	✓	157.9	490.5
CAN [43]	CVPR19	✓	✓	100.2	314.1
SANet [49]	ECCV18	✓	✓	91.2	320.5
CSRNet [3]	CVPR18	✓	✓	85.8	309.1
BL [39]	ICCV19	✓	✓	75.1	299.8
UOT [41]	AAAI21	✓	✓	60.6	252.6
S3 [40]	IJCAI21	✓	✓	59.5	244.1
Transcrowd-Token [4]	SCIS22	×	✓	76.4	319.8
Transcrowd-GAP [4]	SCIS22	×	✓	74.9	295.6
SR2(ours)	-	×	✓	**71.7**	**294.1**

Table 5. Comparison (MAE and MSE) of quantitatively different crowd-counting methods on the EoCo dataset.

Methods	Part A								Part B							
	Cherry		Chickens		Tulips		Vehicles		Jujubes		Wider Face		Wheat		Penguins	
	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE
SR2 (ours)	**39.24**	**71.69**	**49.11**	101.02	**87.61**	96.59	**35.29**	**57.04**	**92.11**	**114.67**	**68.43**	**98.34**	**120.10**	**240.77**	**31.67**	**68.49**
MCNN	69.23	98.51	98.27	126.78	88.26	**95.21**	70.32	96.55	112.58	147.32	112.1	186.52	150.74	267.43	82.69	100.33
CSRNet	45.92	83.25	62.74	**94.38**	97.64	101.13	68.55	89.71	100.37	139.60	69.74	117.42	148.33	251.96	73.54	99.28

4.5. Visualization and Analysis

To discern whether our proposed method learned the crowd changes in the spatial scales and spatial semantics, we divided the image into equal parts and predicted the count of each part separately, as illustrated in Figure 3. Specifically, the leftmost image (yellow) cluster had large-scale variation. The rightmost image (green) cluster had auniformly distributed crowd.

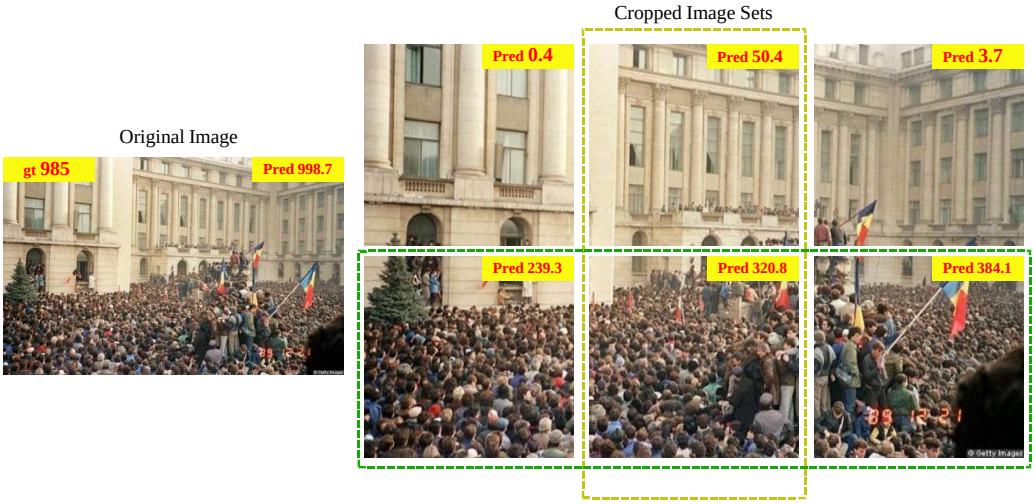

Figure 3. Visualization of a crowd scene example with drastic scale and density. The total predicted count is 998.7, whereas the ground truth is 985.

4.6. Inference Time Calculation Comparison

As illustrated in Table 6, we compared two popular counting approaches with full supervision, BL [39] and CSRNet [3]; we also make a comparison between two popular weakly supervised counting methods, Transcrowd-Token [4] and Transcrowd-GAP [4]. The test was performed on an NVIDIA RTX 3060Ti GPU. Despite the longer run time of SR2 compared to the other methods, excellent performance could be achieved with only half the parameters due to the fact that our approach optionally merges tokens to represent the signature of larger objects while corresponding to certain tokens to maintain the fine-grained features.

Table 6. Calculation resource use of different approaches compared. The bold values of the MAE and MSE represent the best performance.

Methods	Resolution	Parameters ↓	Backbone	FPS ↑
CSRNet [3]	384 × 384	**16.2 M**	VGG16	21.67
BL [39]	384 × 384	21.6 M	VGG19	**45.66**
Transcrowd-Token [4]	384 × 384	86.8 M	vision Transformer	46.41
Transcrowd-GAP [4]	384 × 384	90.4 M	vision Transformer	**46.73**
SR2(ours)	384 × 384	**58.6 M**	ConvMLP	41.56

4.7. Ablation Study

Scale-ranking module: To demonstrate the impact of our proposed ranking branch, we removed the scale-ranking module from our proposed method. Specifically, we only used the MLP encoder, the regression branch, and $\mathcal{L}_1$ loss, which were compared with SR2 on the ShanghaiTechA dataset, as illustrated in Table 7. From Table 7, we can find the impact of the scale-ranking module, which could learn more inner information in the images. Meanwhile, there is a ratio r that determines the size of $I_1, I_2, \ldots, I_n$ in Figure 1. Thus, we selected different ratios to obtain the best performance of the scale-ranking module. The results are shown in Table 8. From Table 8, we found that the ratio of 0.75 had the best performance. Meanwhile, we also tested the capabilities of the amount of $I = \{I_1, I_2, \ldots, I_n\}$, as illustrated in Table 9, and we obtained the best performance when the amount of I was five.

Table 7. The contrast between our proposed method SR2 and its architecture without the ranking branch and graph stage. The bold values of the MAE and MSE represent the best performance.

Methods	Part A		Part B		JHU-Crowd++		QNRF		NWPU-Crowd	
	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE
SR2	**63.9**	**99.6**	**8.4**	**14.0**	**71.7**	**294.1**	**63.9**	**99.6**	**115.7**	**437.2**
SR2 (w/o scale-ranking module)	70.2	116.3	10.7	23.1	75.4	297.6	65.7	103.4	125.6	444.6
SR2 (w/o scale-reasoning module)	64.2	104.2	9.5	16.3	72.9	300.8	65.5	99.8	117.8	441.9
SR2 (w/o convolution stage)	81.2	130.6	13.6	32.7	80.0	350.4	80.7	120.1	126.9	487.5

Table 8. The results of different ratios for SR2 on the datasets. The bold values of the MAE and MSE represent the best performance. The bold values of the MAE and MSE represent the best performance.

Ratios	Part A		Part B		JHU-Crowd++		QNRF		NWPU-Crowd	
	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE
0.7	65.9	102.1	9.8	15.7	74.2	299.6	70.1	103.4	117.3	441.2
0.75	**63.9**	**99.6**	**8.4**	**14.0**	**71.7**	**294.1**	**63.9**	**99.6**	**115.7**	**437.2**
0.85	67.0	106.2	10.3	16.1	75.2	297.1	69.5	101.0	115.8	438.7

Scale-reasoning module: We further showed the impact of the scale-reasoning module. Therefore, we removed the scale-reasoning module from SR2, as illustrated in Table 7. Because the scale-reasoning module can reflect the relationship between the heads and backgroundin the images well, we found a 4.6% improvement in the MSE.

Table 9. The comparisons of different numbers of ranking images in the scale-ranking module on the datasets. The bold values of the MAE and MSE represent the best performance.

n	Part A		Part B		JHU-Crowd++		QNRF		NWPU-Crowd	
	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE
1	66.2	106.6	10.1	21.2	77.2	211.3	68.8	102.1	128.9	503.2
3	64.4	108.1	9.8	21.1	75.8	207.6	64.7	104.3	117.4	472.1
5	**63.9**	**99.6**	**8.4**	**14.0**	**71.7**	**204.1**	**63.9**	**99.6**	**115.7**	**437.2**
6	74.0	113.7	8.7	16.2	72.1	204.1	64.1	99.8	116.9	486.7

The amount of the MLP stages: The MLP stage was used to extract the feature maps from the input images. When we set the amount of MLP stages as three, the capabilities of our approach were the best. The details can be seen in Table 10.

Table 10. Comparisons of different numbers of MLP stages for the basic network. The bold values of the MAE and MSE represent the best performance.

The Amount of MLP Stages	MAE	MSE
2	65.7	102.4
3	63.9	99.6
4	**64.1**	**98.5**
5	66.2	101.1

The convolution stage: In order to solve the limitations of the dimension of the input feature map, we only adopted the channel MLP. However, only using the channel MLP may ignore the spatial information. Thus, we added the convolution stage before the MLP stages, which can increase the interaction of the spatial information in the images. The specifics of the convolution stage are shown in Table 7.
The loss function: We set the loss function as $\mathcal{L}_1 + \gamma\mathcal{L}_{rank}$. γ is a weighting factor reflecting the ranking loss proportion. We found that the results were better when we set γ as 0.7. The details can be seen in Table 11.

Table 11. The comparisons of γ for the loss function on the ShanghaiTechA dataset. The bold values of the MAE and MSE represent the best performance.

γ	MAE	MSE
0.2	67.1	103.8
0.4	67.6	107.3
0.5	67.2	110.0
0.6	64.3	100.7
0.7	**63.9**	**99.6**
0.8	64.1	103.2

5. Conclusions

In this paper, we proposed SR2 for weakly supervised crowd counting. SR2 adopts an MLP-based framework. The framework leverages the convolution stage and the MLP stage for deep feature extraction, which can obtain efficient crowd feature representations. Considering that the MLP architectures ignore the spatial information and the information of the inner image, we added the scale-reasoning module to fuse the region-to-region dependency in the images and capture the multiscale information. Then, the fused features were fed into two branches, which were the regression branch for generating estimated counts and the scale-ranking module for improving the accuracy of the estimated counts in the regression branch. SR2 was assessed on four popular datasets for crowd counting, showing superior results with respect to other excellent approaches. The effectiveness of our approach was proven by both the quantitative and qualitative results. However, SR2 has many network parameters and is unsuitable for hardware with restricted computational resources. Besides, we only took into account crowd counting from images and did not explore crowd counting in videos, which would by appropriate for real-world applications. In the next step in this process, we will try to extend weakly supervised crowd counting to video tasks.

Author Contributions: Conceptualization, M.G., M.D. and H.Z.; methodology, M.G., M.D. and H.Z.; software, M.D. and M.G.; validation, H.Z., M.D. and M.G.; formal analysis, H.Z.; investigation, M.G. and M.D.; resources, H.Z.; data curation, M.D. and Y.C. (Yangjian Chen); writing—original draft preparation, M.G. and M.D.; writing—review and editing, M.G., H.Z. and M.D.; visualization, M.D. and Y.C. (Yongqi Chen); supervision, H.Z.; project administration, H.Z. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Data are contained within the article.

Conflicts of Interest: The authors declare no conflicts of interest.

References

1. Khan, K.; Khan, R.U.; Albattah, W.; Nayab, D.; Qamar, A.M.; Habib, S.; Islam, M. Crowd counting using end-to-end semantic image segmentation. *Electronics* **2021**, *10*, 1293. [CrossRef]
2. Zhang, Y.; Zhou, D.; Chen, S.; Gao, S.; Ma, Y. Single-image crowd counting via multi-column convolutional neural network. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, NV, USA, 27–30 June 2016; pp. 589–597.
3. Li, Y.; Zhang, X.; Chen, D. Csrnet: Dilated convolutional neural networks for understanding the highly congested scenes. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18–23 June 2018; pp. 1091–1100.
4. Liang, D.; Chen, X.; Xu, W.; Zhou, Y.; Bai, X. Transcrowd: Weakly supervised crowd counting with transformers. *Sci. China Inf. Sci.* **2022**, *65*, 160104. [CrossRef]
5. Kolesnikov, A.; Dosovitskiy, A.; Weissenborn, D.; Heigold, G.; Uszkoreit, J.; Beyer, L.; Minderer, M.; Dehghani, M.; Houlsby, N.; Gelly, S.; et al. An image is worth 16 $\times$ 16 words: Transformers for image recognition at scale. In Proceedings of the International Conference on Learning Representations, Virtual Event, 3–7 May 2021.
6. Savner, S.S.; Kanhangad, V. CrowdFormer: Weakly supervised Crowd counting with Improved Generalizability. *arXiv* **2022**, arXiv:2203.03768.
7. Wang, W.; Xie, E.; Li, X.; Fan, D.P.; Song, K.; Liang, D.; Lu, T.; Luo, P.; Shao, L. Pyramid vision transformer: A versatile backbone for dense prediction without convolutions. In Proceedings of the IEEE/CVF International Conference on Computer Vision, Montreal, QC, Canada, 10–17 October 2021; pp. 568–578.
8. Wang, M.; Zhou, J.; Cai, H.; Gong, M. CrowdMLP: Weakly Supervised Crowd Counting via Multi-Granularity MLP. *Pattern Recognit.* **2023**, *144*, 109830. [CrossRef]
9. Godwin, J.; Schaarschmidt, M.; Gaunt, A.L.; Sanchez-Gonzalez, A.; Rubanova, Y.; Veličković, P.; Kirkpatrick, J.; Battaglia, P. Simple gnn regularisation for 3d molecular property prediction and beyond. In Proceedings of the International Conference on Learning Representations, Virtual Event, 3–8 May 2021.
10. Idrees, H.; Tayyab, M.; Athrey, K.; Zhang, D.; Al-Maadeed, S.; Rajpoot, N.; Shah, M. Composition loss for counting, density map estimation and localization in dense crowds. In Proceedings of the European Conference on Computer Vision (ECCV), Munich, Germany, 8–14 September 2018; pp. 532–546.
11. Sindagi, V.; Yasarla, R.; Patel, V.M. Jhu-crowd++: Large-scale crowd counting dataset and a benchmark method. *IEEE Trans. Pattern Anal. Mach. Intell.* **2020**, *44*, 2594–2609. [CrossRef] [PubMed]
12. Wang, Q.; Gao, J.; Lin, W.; Li, X. NWPU-crowd: A large-scale benchmark for crowd counting and localization. *IEEE Trans. Pattern Anal. Mach. Intell.* **2020**, *43*, 2141–2149. [CrossRef] [PubMed]
13. He, K.; Zhang, X.; Ren, S.; Sun, J. Deep residual learning for image recognition. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, NV, USA, 27–30 June 2016; pp. 770–778.
14. Yu, R.; Wang, S.; Lu, Y.; Di, H.; Zhang, L.; Lu, L. SAF: Semantic Attention Fusion Mechanism for Pedestrian Detection. In Proceedings of the Pacific Rim International Conference on Artificial Intelligence, Yanuca Island, Fiji, 26–30 August 2019; pp. 523–533.
15. Chen, D.; Lu, L.; Lu, Y.; Yu, R.; Wang, S.; Zhang, L.; Liu, T. Cross-domain scene text detection via pixel and image-level adaptation. In Proceedings of the International Conference Neural Information Processing, Sydney, NSW, Australia, 12–15 December 2019; pp. 135–143.
16. Liu, Z.; Lin, Y.; Cao, Y.; Hu, H.; Wei, Y.; Zhang, Z.; Lin, S.; Guo, B. Swin transformer: Hierarchical vision transformer using shifted windows. In Proceedings of the IEEE/CVF International Conference on Computer Vision, Montreal, QC, Canada, 10–17 October 2021; pp. 10012–10022.
17. Babu Sam, D.; Surya, S.; Venkatesh Babu, R. Switching convolutional neural network for crowd counting. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, HI, USA, 21–26 July 2017; pp. 5744–5752.
18. Wang, S.; Lu, Y.; Zhou, T.; Di, H.; Lu, L.; Zhang, L. SCLNet: Spatial context learning network for congested crowd counting. *Neurocomputing* **2020**, *404*, 227–239. [CrossRef]
19. Xie, Y.; Lu, Y.; Wang, S. Rsanet: Deep recurrent scale-aware network for crowd counting. In Proceedings of the 2020 IEEE International Conference on Image Processing (ICIP), Anchorage, AL, USA, 19–22 September 2020; pp. 1531–1535.
20. Chen, X.; Yu, X.; Di, H.; Wang, S. Sa-internet: Scale-aware interaction network for joint crowd counting and localization. In Proceedings of the Pattern Recognition and Computer Vision, Nashville, TN, USA, 20–25 June 2021; pp. 203–215.
21. Duan, Z.; Wang, S.; Di, H.; Deng, J. Distillation remote sensing object counting via multi-scale context feature aggregation. *IEEE Trans. Geosci. Remote Sens.* **2021**, *60*, 5613012. [CrossRef]
22. Liang, L.; Zhao, H.; Zhou, F.; Ma, M.; Yao, F.; Ji, X. PDDNet: Lightweight congested crowd counting via pyramid depth-wise dilated convolution. *Appl. Intell.* **2022**, *53*, 10472–10484. [CrossRef]

23. Sun, Y.; Li, M.; Guo, H.; Zhang, L. MSGSA: Multi-Scale Guided Self-Attention Network for Crowd Counting. *Electronics* **2023**, *12*, 2631. [CrossRef]
24. Dong, J.; Zhao, Z.; Wang, T. Crowd Counting by Multi-Scale Dilated Convolution Networks. *Electronics* **2023**, *12*, 2624. [CrossRef]
25. Yan, Z.; Qi, Y.; Li, G.; Liu, X.; Zhang, W.; Yang, M.H.; Huang, Q. Progressive Multi-resolution Loss for Crowd Counting. *IEEE Trans. Circuits Syst. Video Technol.* **2023**. [CrossRef]
26. Liu, X.; Li, G.; Qi, Y.; Han, Z.; Huang, Q.; Yang, M.H.; Sebe, N. Consistency-Aware Anchor Pyramid Network for Crowd Localization. *arXiv* **2022**, arXiv:2212.04067.
27. Ge, C.; Song, Y.; Ma, C.; Qi, Y.; Luo, P. Rethinking Attentive Object Detection via Neural Attention Learning. *IEEE Trans. Image Process.* **2023**. [CrossRef] [PubMed]
28. Tian, Y.; Chu, X.; Wang, H. Cctrans: Simplifying and improving crowd counting with transformer. *arXiv* **2021**, arXiv:2109.14483.
29. Chu, X.; Tian, Z.; Wang, Y.; Zhang, B.; Ren, H.; Wei, X.; Xia, H.; Shen, C. Twins: Revisiting the design of spatial attention in vision transformers. *Adv. Neural Inf. Process. Syst.* **2021**, *34*, 9355–9366.
30. Gao, J.; Gong, M.; Li, X. Congested crowd instance localization with dilated convolutional Swin transformer. *arXiv* **2021**, arXiv:2108.00584.
31. Lin, H.; Ma, Z.; Ji, R.; Wang, Y.; Hong, X. Boosting Crowd Counting via Multifaceted Attention. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, New Orleans, LA, USA, 18–24 June 2022; pp. 19628–19637.
32. Luo, A.; Yang, F.; Li, X.; Nie, D.; Jiao, Z.; Zhou, S.; Cheng, H. Hybrid graph neural networks for crowd counting. In Proceedings of the AAAI Conference on Artificial Intelligence, New York, NY, USA, 7–12 February 2020; Volume 34, pp. 11693–11700.
33. Lei, Y.; Liu, Y.; Zhang, P.; Liu, L. Towards using count-level weak supervision for crowd counting. *Pattern Recognit.* **2021**, *109*, 107616. [CrossRef]
34. Yang, Y.; Li, G.; Wu, Z.; Su, L.; Huang, Q.; Sebe, N. Weakly supervised crowd counting learns from sorting rather than locations. In Proceedings of the European Conference on Computer Vision (ECCV), Glasgow, UK, 23–28 August 2020; pp. 1–17.
35. Wang, F.; Liu, K.; Long, F.; Sang, N.; Xia, X.; Sang, J. Joint CNN and Transformer Network via weakly supervised Learning for efficient crowd counting. *arXiv* **2022**, arXiv:2203.06388.
36. Tolstikhin, I.O.; Houlsby, N.; Kolesnikov, A.; Beyer, L.; Zhai, X.; Unterthiner, T.; Yung, J.; Steiner, A.; Keysers, D.; Uszkoreit, J.; et al. Mlp-mixer: An all-mlp architecture for vision. *Adv. Neural Inf. Process. Syst.* **2021**, *34*, 24261–24272.
37. Dey, R.; Salem, F.M. Gate-variants of gated recurrent unit (GRU) neural networks. In Proceedings of the 2017 IEEE 60th International Midwest Symposium on Circuits and Systems (MWSCAS), Boston, MA, USA, 6–9 August 2017; pp. 1597–1600.
38. Jiang, S.; Wang, Q.; Cheng, F.; Qi, Y.; Liu, Q. A Unified Object Counting Network with Object Occupation Prior. *IEEE Trans. Circuits Syst. Video Technol.* **2023**. [CrossRef]
39. Ma, Z.; Wei, X.; Hong, X.; Gong, Y. Bayesian loss for crowd count estimation with point supervision. In Proceedings of the IEEE/CVF International Conference on Computer Vision, Seoul, Republic of Korea, 27 October–2 November 2019; pp. 6142–6151.
40. Lin, H.; Hong, X.; Ma, Z.; Wei, X.; Qiu, Y.; Wang, Y.; Gong, Y. Direct Measure Matching for Crowd Counting. *arXiv* **2021**, arXiv:2107.01558.
41. Ma, Z.; Wei, X.; Hong, X.; Lin, H.; Qiu, Y.; Gong, Y. Learning to count via unbalanced optimal transport. In Proceedings of the AAAI Conference on Artificial Intelligence, Vancouver, BC, Canada, 2–9 February 2021; Volume 35, pp. 2319–2327.
42. Gao, J.; Lin, W.; Zhao, B.; Wang, D.; Gao, C.; Wen, J. C^3 framework: An open-source pytorch code for crowd counting. *arXiv* **2019**, arXiv:1907.02724.
43. Liu, W.; Salzmann, M.; Fua, P. Context-aware crowd counting. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Long Beach, CA, USA, 15–20 June 2019; pp. 5099–5108.
44. Wang, Q.; Gao, J.; Lin, W.; Yuan, Y. Learning from synthetic data for crowd counting in the wild. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Long Beach, CA, USA, 15–20 June 2019; pp. 8198–8207.
45. Wan, J.; Wang, Q.; Chan, A.B. Kernel-Based Density Map Generation for Dense Object Counting. *IEEE Trans. Pattern Anal. Mach. Intell.* **2022**, *44*, 1357–1370. [CrossRef]
46. Wan, J.; Chan, A. Modeling noisy annotations for crowd counting. *Adv. Neural Inf. Process. Syst.* **2020**, *33*, 3386–3396.
47. Shi, Z.; Zhang, L.; Liu, Y.; Cao, X.; Ye, Y.; Cheng, M.M.; Zheng, G. Crowd counting with deep negative correlation learning. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18–23 June 2018; pp. 5382–5390.
48. Sindagi, V.A.; Patel, V.M. Cnn-based cascaded multi-task learning of high-level prior and density estimation for crowd counting. In Proceedings of the 2017 14th IEEE International Conference on Advanced Video and Signal Based Surveillance (AVSS), Lecce, Italy, 23 August–1 September 2017; pp. 1–6.
49. Cao, X.; Wang, Z.; Zhao, Y.; Su, F. Scale aggregation network for accurate and efficient crowd counting. In Proceedings of the European Conference on Computer Vision (ECCV), Munich, Germany, 8–14 September 2018; pp. 734–750.