mathematics

MDPI

Article

CC-DETR: DETR with Hybrid Context and Multi-Scale Coordinate Convolution for Crowd Counting

Yanhong Gu [1,2], Tao Zhang [1,2], Yuxia Hu [3] and Fudong Nian [1,2,3,*]

[1] School of Advanced Manufacturing Engineering, Hefei University, Hefei 230601, China; guyh@hfuu.edu.cn (Y.G.); 18075366582@163.com (T.Z.)
[2] Anhui Provincial Engineering Technology Research Center of Intelligent Vehicle Control and Integrated Design Technology, Hefei 230601, China
[3] Anhui International Joint Research Center for Ancient Architecture Intellisencing and Multi-Dimensional Modeling, Anhui Jianzhu University, Hefei 230601, China
* Correspondence: nianfd@hfuu.edu.cn

Abstract: Prevailing crowd counting approaches primarily rely on density map regression methods. Despite wonderful progress, significant scale variations and complex background interference within the same image remain challenges. To address these issues, in this paper we propose a novel DETR-based crowd counting framework called Crowd Counting DETR (CC-DETR), which aims to extend the state-of-the-art DETR object detection framework to the crowd counting task. In CC-DETR, a DETR-like encoder–decoder structure (Hybrid Context DETR, i.e., HCDETR) is proposed to tackle complex visual information by fusing features from hybrid semantic levels through a transformer. In addition, we design a Coordinate Dilated Convolution Module (CDCM) to effectively employ position-sensitive context information in different scales. Extensive experiments on three challenging crowd counting datasets (ShanghaiTech, UCF-QNRF, and NWPU) demonstrate that our model is effective and competitive when compared against SOTA crowd counting models.

Keywords: crowd counting; transformer; DETR

MSC: 68U10

check for updates

Citation: Gu, Y.; Zhang, T.; Hu, Y.; Nian, F. CC-DETR: DETR with Hybrid Context and Multi-Scale Coordinate Convolution for Crowd Counting. *Mathematics* **2024**, *12*, 1562. https://doi.org/10.3390/math12101562

Academic Editor: Jakub Nalepa

Received: 9 April 2024
Revised: 9 May 2024
Accepted: 15 May 2024
Published: 17 May 2024

Copyright: © 2024 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (https://creativecommons.org/licenses/by/4.0/).

1. Introduction

Crowd counting pertains to the estimation of individual presence within a specified area utilizing images as the primary data source [1]. This discipline has garnered substantial interest due to its extensive applicability across diverse sectors such as surveillance, crowd management, urban planning, and retail analytics.

In the realm of crowd counting via images, two primary challenges persist. As illustrated in Figure 1, larger scales are proximal to the camera, while smaller scales are located in more distant areas. Currently, two predominant methods are utilized. The first approach employs multiple columns with distinct convolutional kernel networks to extract features, subsequently fusing them to generate feature maps [2]. However, the constrained receptive field of mainstream CNN networks presents difficulties in capturing optimal global information. A novel enhancement method was introduced in [3] which involves merging semantic features extracted at varying resolutions from the backbone. As features at different resolutions express characteristics differently across various scales, higher-resolution feature maps are more apt for representing small-scale features. Savner et al. [4] achieved crowd estimation by merging features from four distinct resolutions, followed by a straightforward regression head. Nevertheless, modeling global feature information yields suboptimal results. Furthermore, in traditional convolutional neural networks, the perception of positional information during image processing primarily originates from the local receptive field of the convolution operations. This local receptive field may not

provide ample global positional information, particularly when dealing with large inputs, necessitating more comprehensive global background information.

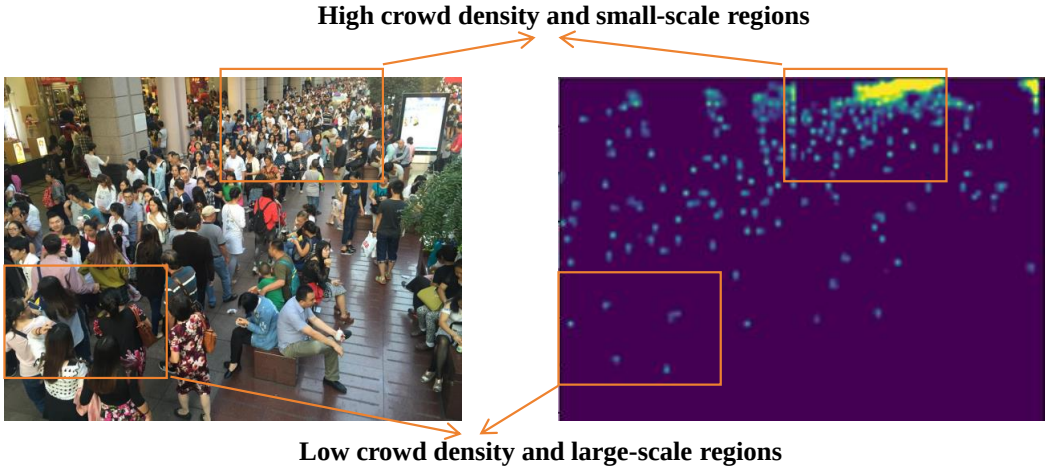

Figure 1. An illustration of the challenge of crowd counting concerning the simultaneous presence of low density in nearer large-scale regions and high density in more distant small-scale regions.

The recent surge in transformer models [5], which employ global self-attention mechanisms, has significantly improved the performance of numerous natural language processing tasks. However, it was not until the advent of vision transformers (ViT) [6] and the segmentation of images into patches to address local structural inductive biases that transformer models could compete with and even outperform CNN models in visual tasks. The evolution of vision transformers underscores the pivotal role of global self-attention mechanisms and local inductive biases in visual tasks. Such models adeptly capture the global context of crowds to model long-distance dependencies, thereby enhancing counting performance. In crowd counting tasks, most methods [7,8] feed the highest-level features extracted from the backbone into the transformer in order to further capture the global context of the crowd. However, this approach does not fully exploit the influence of feature maps extracted at different resolutions by the backbone network in capturing multi-scale information about the crowd. Capturing multi-scale information about crowds presents a significant challenge.

In this paper, we tackle the shortcomings of traditional convolutions' weak global capture ability and transformers' multi-scale constraints in crowd information capture. We introduce CC-DETR, a novel framework to extend DETR (DEtection TRansformer) [9], a popular recent object detection method, to crowd counting tasks. The proposed CC-DETR has two novel modules, i.e., HCDETR and CDCM. Our proposed Hybrid Context DETR (HCDETR) module enhances DETR's global information utilization through its attention mechanism. We employ an encoder–decoder structure for feature extraction, where the encoder processes information extracted from the backbone and provides it as input to the decoder. The decoder then extracts features from both the backbone and the encoder's output. Notably, we incorporate feature information from the backbone into the decoder's input for the first time, significantly aiding the decoder in learning semantic crowd information. Moreover, by utilizing inputs from various levels of the backbone in both the encoder and decoder, we effectively mitigate scale variation issues. Additionally, we propose a Coordinate Dilated Convolution Module (CDCM) to manage scale variations and convolutional coordinate transformations [10]. Our approach exhibits exceptional performance in high-density and small-scale areas as well as in low-density and large-scale areas.

In summary, the contributions of this paper are threefold:

- We propose a novel method called CC-DETR which extends the recent DETR-based object detection framework into crowd counting task.

- To handle complex semantics we propose the HCDETR module, which utilizes a DETR-like encoder–decoder structure for hybrid context understanding. By employing features from hybrid levels of the backbone as inputs for the encoder and decoder, it accomplishes simultaneous learning of global features and fusion across different scales.
- We propose a regression head with a CDCM module, which integrates coordinate convolution and parallel dilated convolution with different dilation factors to achieve location-sensitive multi-scale information modeling.
- We conduct extensive experiments on multiple benchmark datasets to demonstrate the effectiveness of the proposed method, revealing that CC-DETR outperforms several SOTA crowd counting methods.

2. Related Works

Presently, mainstream crowd counting methodologies can be broadly categorized into two types: detection-based methods and density-based methods.

2.1. Detection-Based Methods

Early approaches relied on human detection [11,12], in which a detection model was constructed using Convolutional Neural Networks (CNN) to predict bounding boxes for each person. The count of people was represented by the number of detected boxes. However, these methods face challenges in scenes with severe occlusion. In densely populated areas, the accumulation of detection boxes often leads to inaccuracies.

2.2. Density-Based Methods

To achieve high performance in crowded scenes, density-based methods were introduced to regress density maps from input images [13]. The density map [14] reflects the probability estimate of the crowd, with each pixel value representing the likelihood of a person being at that location. The sum of pixel values indicates the count of people. This approach largely avoids detection errors caused by an excessive number of bounding boxes [12,15]. The primary focus of this method is on extracting a feature map that effectively captures the semantic features of the crowd.

CNNs are popularly used to generate predicted density maps, as they possess translation equivariance and are effective in extracting local details. To tackle scale variations, employing multiple receptive fields is effective for learning from people of various sizes. Sam et al. [16] applied multi-column operations to integrate features of different resolutions for regression in order to obtain the predicted count of the crowd. Zeng et al. [17] introduced further improvements, including multi-scale mechanisms. Optimal transport [18], another CNN-based approach, has also shown its effectiveness.

With the vision transformer (ViT) algorithm attempting to directly apply the standard transformer structure to images, a significant amount of work in the field of crowd counting tasks has begun to leverage transformer structures. Liu et al. [19] and Yan et al. [20] both proposed using a scale attention mechanism [5], which relies on Gaussian or convolutional kernels of different sizes to generate density maps for scale variation regions, to alleviate the impact of scale variations on crowd counting. Lin et al. [7] introduced Learnable Region Attention, dynamically assigning specialized attention to each feature position. Sun et al. [21] incorporated tokens to learn crowd features in images for better feature extraction. Du et al. [22] divided the crowd into different density levels, assigning different weights to predictions based on density levels. Tian et al. [23] employed convolution with different dilation factors to better integrate multi-scale information. Visual transformers have recently gained popularity in computer vision. In particular, DETR utilizes a transformer decoder to model object detection in an end-to-end pipeline [24]. Building upon DETR, Conditional DETR further incorporates spatial queries and keywords associated with objects' endpoints or regions, thereby expediting the convergence of DETR. In the context of crowd analysis, Liang et al. [25] proposed TransCrowd, which redefines

the weakly supervised counting problem from a sequence-to-count perspective. Several approaches have demonstrated the effectiveness of visual transformers in point-supervised crowd counting tasks. These methods employ attention from the Swin Transformer [26] for crowd counting.

3. Our Method

The architecture of our proposed method is depicted in Figure 2. Initially, the input image is processed through a feature extraction network based on the Pyramid Transformer backbone, producing features at varying resolutions. Following this, the feature map corresponding to the lowest resolution is introduced into the transformer encoder of the HCDETR module to extract global features. The output from both the transformer encoder and the higher-resolution feature map derived from the Pyramid Transformer backbone are subsequently flattened into a one-dimensional sequence, which serves as the input for the transformer decoder within the HCDETR module. Subsequently, the output from the HCDETR module is amalgamated with the upsampled features from the lower resolution of the backbone and fed into the Coordinate Dilated Convolution Module (CDCM) for density map regression. The final density map is utilized along with the cumulative sum of all its pixel values to construct the loss function in a fully supervised manner.

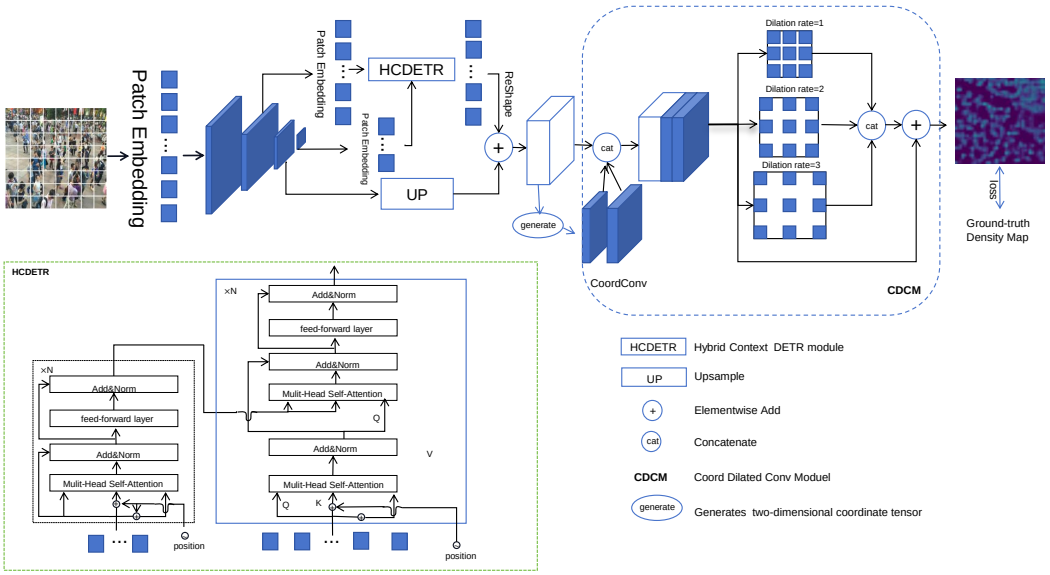

Figure 2. Overview of the proposed CC-DETR. Initially, the input image is transformed into a one-dimensional sequence, then the output is fed into a backbone-based architecture. We employ a Pyramid Transformer to capture global context through various downsampling stages. Subsequently, Stage 2 and Stage 4 serve as the inputs for the decoder and encoder, respectively, of the HCDETR. The resulting sequence is reshaped into a 2D feature map, which is fused with the upsampled output from Stage 3. This combined input is then passed into the CDCM module for regression, resulting in the final predicted density map. The loss is computed by comparing it with the ground truth labels.

3.1. Backbone

We adopt the Twins Pyramid Transformer backbone from [27], which is distinguished by alternating local and global attention. Thanks to possessing both local and global receptive fields, Twins can capture both short-term and long-term relationships. As an extension of the Transformer architecture, Twins introduces the Spatially Separable Self-Attention (SSSA) model to enhance computational efficiency. In the Locally-grouped Self-Attention (LSA) stage, the input sequence is reshaped into a 2D feature map and local windows perform self-attention calculations independently. This results in local features. The Global Sub-sampled Attention (GSA) stage further reduces computational cost by generating representatives for each sub-window through convolutional operations.

These representatives consolidate key information, enabling communication between sub-windows and capturing global features. In summary, Twins combines LSA for local features and GSA for global features, enhancing efficiency in self-attention calculations.

For an input image with dimensions $H \times W \times 3$, after undergoing processing in five layers, we obtain feature maps with different resolutions. In layer 0, the input image $R^{H\times W\times 3}$ is processed to yield an output with a shape of $R^{\frac{H}{2}\times\frac{W}{2}\times\frac{d}{16}}$. Similarly, the output for layer 1 is $R^{\frac{H}{4}\times\frac{W}{4}\times\frac{d}{8}}$, layer 2 is $R^{\frac{H}{8}\times\frac{W}{8}\times\frac{d}{4}}$, layer 3 is $R^{\frac{H}{16}\times\frac{W}{16}\times\frac{d}{2}}$, and layer 4 is $R^{\frac{H}{32}\times\frac{W}{32}\times d}$. The spatially separable self-attention (SSSA) for each stage can be expressed as follows.

$$z_{ij}^{l} = LSA(LayerNorm(z_{ij}^{l-1})) + z_{ij}^{l-1} \tag{1}$$

$$z_{ij}^{l} = MLP(LayerNorm(z_{ij}^{l})) + z_{ij}^{l} \tag{2}$$

$$Z^{l} = GSA(LayerNorm(Z^{l})) + Z^{l} \tag{3}$$

$$Z^{l} = MLP(LayerNorm(Z^{l})) + Z^{l} \tag{4}$$

$$i \in 1, 2..., k_1, j \in 1, 2..., k_2$$

Here, z_{ij}^{l} represents the small sub-window in the i-th row and j-th column of layer l, LSA represents local grouped self-attention within each sub-window, and GSA interacts with the global sub-sampled attention through representative keys from each sub-window (generated by the sub-sampling function).

3.2. Hybrid Context DETR

Hybrid Context DETR consists of two parts: a transformer encoder and a transformer decoder.

3.2.1. Transformer Encoder

Initially, the encoder takes a sequence as input. During this process, the spatial dimensions of the feature map $R^{\frac{H}{32}\times\frac{W}{32}\times d}$ from layer 3 are compressed to one dimension, resulting in a $R^{\frac{HW}{32^2}\times d}$ feature map. A learnable position encoding is then added to the feature map. Adding extra learnable parameters (initialized to zero by default) as position encodings is intended to provide more spatial information for spatial perception tasks, thereby enhancing the neural network's ability to perceive spatial information and improving both performance and robustness in handling spatial perception tasks. The transformer encoder is composed of N stacked multi-head self-attention layers and a feed-forward (FC) layer. The input sequence passes through these transformer encoder layers to produce the final output feature $F_e(R^{\frac{H}{32}\times\frac{W}{32}\times d})$. The multi-head self-attention is defined as follows:

$$Q_i = QW_i^{Q}, \quad K_i = KW_i^{K}, \quad V_i = VW_i^{V} \tag{5}$$

$$\text{head}_i = \text{softmax}\left(\frac{Q_i K_i^{T}}{\sqrt{d_k}}\right) V_i \tag{6}$$

$$MSA(Q, K, V) = \text{Concat}(\text{head}_1, \ldots, \text{head}_i) W^{O} \tag{7}$$

where Q, K, and V are obtained from the same input (F_0), head_i denotes the i-th attention head, W_i^{Q}, W_i^{K}, W_i^{V}, and W^{O} represent learnable weight matrices, and $\sqrt{d_k}$ stands for the scaling factor, which is numerically equivalent to the channels.

3.2.2. Transformer Decoder

In the transformer decoder stage, the channel count of inputs derived from the encoder's output F_e and the layer 2 output $R^{\frac{H}{32}\times\frac{W}{32}\times\frac{d}{4}}$ from the backbone is adjusted to $R^{\frac{H}{8}\times\frac{W}{8}\times d}$ through convolution. The transformer decoder is composed of N concatenated decoder layers, with each layer comprising three sub-layers: (1) the Multi-Head Self-

Attention (MSA) layer; (2) the Cross-Attention (CA) layer; and (3) a Feedforward (FC) layer. Initially, we compress the spatial dimensions of $R^{\frac{H}{8}\times\frac{W}{8}\times\frac{d}{4}}$ into one dimension, obtaining a feature map $R^{\frac{HW}{8^2}\times d}$, which is then input into the Multi-Head Self-Attention layer to produce an output. Subsequently, the output features $F_e(R^{\frac{H}{32}\times\frac{W}{32}\times d})$ from the transformer encoding layer are utilized as the K and V for the Cross-Attention layer. The output Q_n from the Multi-Head Self-Attention layer serves as the Q for the Cross-Attention layer. F_d is computed using the formula below:

$$F_d = \text{softmax}\left(\frac{QK^T}{\sqrt{d_k}}\right)V. \tag{8}$$

The output F_d from the Cross-Attention (CA) layer is fed into the Feedforward (FC) layer to obtain the final decoder output $F_h(R^{\frac{HW}{8^2}\times d})$. We reshape $F_h(R^{\frac{HW}{8^2}\times d})$ into $R^{\frac{H}{8}\times\frac{W}{8}\times d}$ and perform elementwise addition by upsampling the output $R^{\frac{H}{16}\times\frac{W}{16}\times\frac{d}{2}}$ from layer 3 of the backbone; the result is then used for regression in the CDCM module.

3.3. Coordinate Dilated Convolution Module

At this point, our transformer-based backbone and the HCDETR module have already captured sufficient global information. Therefore, we utilize an efficient regression head to precisely regress the density map. Specifically, we construct a multi-scale receptive field module to detect the global scale and density variance. A straightforward approach is to stack dilated convolutional layers. The DConv (Dilated Convolution) layer can expand the receptive field while maintaining spatial resolution. However, this design requires careful consideration of the dilation factors for each layer to avoid grid effects [28], resulting in some pixels being lost in subsequent convolutions. Grid effects significantly impact crowd counting results, as a missing pixel in advanced feature maps causes the regression head to lose valuable crowd information. CNN models primarily rely on convolutional operations for spatial information perception, lacking a mechanism to directly process global coordinate information from the input. In crowd counting tasks, global coordinate information may be crucial for ensuring the model's resilience against background interference.

Therefore, our proposed CDCM module is mainly composed of two parts: an added croodconv and a set of four parallel dilated convolutions with dilation rates of 0, 1, 2, and 3. First, we concatenate two croodconv layers on the feature map output by the HCDETR. This introduces the coordinate information of the feature map as additional channels, allowing the network to better understand the spatial structure of the feature map. We design multi-scale dilated convolutions by stacking DConv layers with different dilation rates in parallel to avoid grid effects, preventing the loss of crowd information during training. To address the multiscale issue, we considered various choices of kernel size and dilation rate in our model design. Considering that feature maps shrink after feature extraction, we opted to set the corresponding kernel size and dilation rate to be as small as possible in order to accommodate crowd counting scenes filled with small-scale objects. This approach facilitates better capture of fine-grained features in images, thereby enhancing the model's ability to detect small-scale objects. Additionally, the utilization of different kernel sizes and dilation rates helps to augment the model's perception of information across various scales, thereby enhancing its adaptability in handling multiscale scenes. Each convolutional layer is followed by a batch normalization (BN) layer and a ReLU activation function. We concatenate the output feature maps of dilation rates 1, 2, and 3, then add them to the column with dilation rate 0 to leverage multi-scale features. Finally, we use a 1 × 1 convolutional layer to regress the density map.

3.4. Loss Function Design

Our design is based on popular loss functions [29] used in crowd counting, comprising a weighted sum of the counting loss (L_{count}), optimal transport loss (L_{OT}), and total varia-

tion loss (L_{TV}). While L_{OT} performs well in dense regions, its performance is suboptimal in low-density areas. To address this issue, we introduce L_{TV}. L_{TV} penalizes differences between adjacent pixels, resulting in a smoother density map. In low-density regions where the population distribution is sparse, L_{OT} may be susceptible to extreme values, leading to overfitting and producing unreasonable density estimates. The smoothing penalty of L_{TV} can mitigate this overfitting phenomenon, making the model's density estimation more stable in low-density areas. For the predicted density map D and its ground truth, the loss function is defined as follows:

$$L_{total} = L_{count}(P, G) + \lambda_1 L_{OT} + \lambda_2 L_{TV}(D, D') \tag{9}$$

where P and G represent the counts of D and D', respectively (with P obtained by summing all pixels in the predicted density map D), the L_{count} calculates the L1 loss between P and G, and λ_1 and λ_2 are the loss coefficients, which were set to 0.01 and 0.1, respectively, in our experiments. We explored alternative values for optimal selection; these experimental comparisons are presented in the Experiments section.

4. Experiments

4.1. Datasets

Following previous crowd counting work [23,30], we evaluated our model on three widely used datasets: ShanghaiTech, UCF_QNRF, and NWPU-Crowd. Several images selected from these datasets are shown in Figure 3. The sources, scenes, colors, shooting angles, and number of people in these datasets are all different, as can be seen from the following specific information about each.

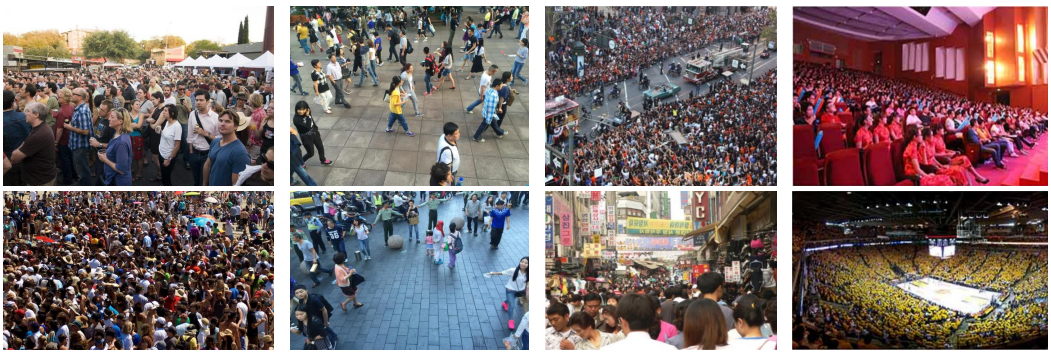

Figure 3. Example images from the different datasets.

ShanghaiTech [2] is divided into two parts, A and B. Part A consists of 300 training images and 182 test images, both sourced from the internet, depicting densely populated scenes. Part B contains 400 training images and 316 test images. Part B captures real-life scenes of bustling streets in Shanghai, with objects relatively sparsely distributed.

UCF_QNRF [31] is a dense dataset, comprising 1535 images (1201 for training and 334 for testing) and around one million annotations. The average number of pedestrians per image is 815, with a maximum of 12,865. UCF-QNRF serves as a valuable resource for training and evaluating models designed for large-scale crowd density estimation. UCF-QNRF is distinguished from similar datasets by its diverse array of scenes, multiple perspectives, varying lighting conditions, and density fluctuations in annotated human figures. Consequently, it is highly conducive for training neural networks.

NWPU-Crowd [32] is a large-scale dataset collected from various scenes, totaling 5109 images. These images are randomly distributed into training (3109), validation (500), and test (1500) sets. The dataset provides both point-level and box-level annotations. NWPU-Crowd stands out as the largest dataset for crowd density estimation to date. It encompasses some negative samples, such as extremely dense crowds, which enhance the robustness of trained models. Additionally, the images in this dataset exhibit higher

resolutions compared to others, and the range of annotated entities per image is notably broad, spanning from 0 to 20,033.

4.2. Implementation Details

The backbone was the official Twins-SVT-large model pretrained on the ImageNet-1k dataset. Only random cropping and random horizontal flipping operations were used as data augmentations for all experiments. The crop size of both ShanghaiTech Part B and UCF_QNRF was 512. The crop sizes for ShanghaiTech Part A and NWPU-Crowd were set to 256 and 384, respectively. Our model was optimized end-to-end with the AdamW optimizer using a batch size of 8. The initial learning rate was set to 1×10^{-5}. We set the value of d in the HCDETR module to 1024. Meanwhile, an L2 regularization term with a weight of 0.0001 was adopted to avoid overfitting. All experiments were based on PyTorch and used two NVIDIA RTX3090 GPUs.

4.3. Evaluation Metrics

The Mean Absolute Error (MAE) and Mean Square Error (MSE) were used as the counting metrics, while the mean Normalized Absolute Error (NAE) was used as an extra metric, defined as follows:

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

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

$$NAE = \frac{1}{N}\sum_{i=1}^{N}\frac{|P_i - G_i|}{G_i} \tag{12}$$

where N is the number of testing images and P_i and G_i are the predicted and ground-truth number of crowds in the i-th image, respectively.

4.4. Comparison With State-of-the-Art Methods

We compared our CC-DETR method with currently popular alternatives, including both supervised and unsupervised approaches. Unsupervised methods can reduce the data annotation workload to an extent, as they do not need to use positional information, but tend to have slightly inferior performance compared to supervised methods. As shown in Table 1, our proposed CC-DETR method outperforms unsupervised approaches, achieving an improvement of no less than 8% on the ShanghaiTech Part A dataset, no less than 7% on the ShanghaiTech Part B dataset, and no less than 3% on the UCF-QNRF dataset. Compared to supervised methods, CC-DETR demonstrates an improvement of no less than 1% on the ShanghaiTech Part A dataset, no less than 1.5% on the ShanghaiTech Part B dataset, and a lower MAE on the UCF-QNRF dataset compared to the other datasets.

In addition, we compared our method with popular crowd counting approaches on the NWPU-Crowd dataset, with our method exhibiting excellent performance. The comparative results are presented in Table 2.

4.5. Visualizations

Below, we provide qualitative visualizations to analyze the effectiveness of our method. The images depicted in Figure 4 were randomly selected from each dataset for presentation. It can be observed that our method exhibits outstanding performance in resisting background interference. It effectively focuses more attention on crowd information, highlighting its ability to mitigate background distractions.

Table 1. Comparison of counting performance on the UCF-QNRF, ShanghaiTech Part A, and ShanghaiTech Part B datasets. MAE: Mean Absolute Error; MSE: Mean Square Error. The best results were shown in boldface.

Method	Position	Part A		Part B		QNRF	
		MAE	MSE	MAE	MSE	MAE	MSE
L2SM [33]	×	64.2	98.4	7.2	11.1	104.7	173.6
TransCrowd [25]	×	66.1	105.1	9.3	16.1	97.2	168.5
MATT [34]	×	80.1	129.4	11.7	17.5	-	-
AMSNet [35]	×	**56.7**	**93.4**	**6.7**	**10.2**	101.8	**163.2**
KDMG [36]	×	63.8	99.2	7.8	12.7	**99.5**	173.0
DSSI-Net [37]	✓	60.6	96.0	6.8	10.3	99.1	159.2
LibraNet [38]	✓	55.9	97.1	7.3	11.3	88.1	143.7
AMRNet [39]	✓	61.5	98.3	7.0	11.0	86.6	152.2
DM-Count[29]	✓	59.7	95.7	7.4	11.8	85.6	148.3
BCCT [21]	✓	53.1	82.2	7.3	11.3	83.3	143.4
P2PNet [30]	✓	52.7	85.1	6.3	9.9	85.3	154.5
CCTrans [23]	✓	52.3	84.9	6.2	9.9	82.8	**142.3**
CC-DETR (ours)	✓	**51.8**	**83.3**	**6.1**	**9.7**	**82.2**	144.6

Table 2. Counting results of various methods on the NWPU validation and test sets. MAE: Mean Absolute Error; MSE: Mean Square Error; NAE: Normalized Absolute Error. The best results were shown in boldface.

Method	Position	Validation Set		Test Set		
		MAE	MSE	MAE	MSE	NAE
MCNN [2]	×	218.5	218.5	232.5	714.6	-
CSRNet [14]	×	104.8	433.4	121.3	387.8	-
SFCN [40]	×	95.4	608.3	105.4	424.1	-
TransCrowd [25]	×	88.4	400.5	117.7	451.0	0.244
BL	✓	93.6	470.4	105.4	454.2	0.203
DM-Count [29]	✓	70.5	357.6	88.4	388.6	0.169
BCCT [21]	✓	53.0	170.3	82.0	366.9	0.164
P2PNet [30]	✓	-	-	77.4	362.0	-
CC-DETR (ours)	✓	**41.80**	**110.37**	**75.76**	**344.17**	**0.150**

4.6. Ablation Studies

An ablation study was conducted on the ShanghaiTech Part A dataset, which encompasses images from various scenes and environments such as streets, squares, and malls. This diversity aids in evaluating the model's generalization performance under different backgrounds and lighting conditions.

4.6.1. Effect of N

The model employs an encoder–decoder structure, with both the encoder and decoder consisting of a multi-layer architectures. Through comparative experiments, we verified that adopting an $N = 8$ structure yields optimal results. The comparison is presented in Figure 5. Upon analyzing and comparing the above results, we posit that a smaller value of N hinders the model's ability to capture the intricate hierarchical structures and abstract features within crowd data, thereby constraining the model's representational capacity. Conversely, a larger N may result in a deeper encoder, potentially leading to overfitting on the training data and consequently restricting the model's generalization capabilities.

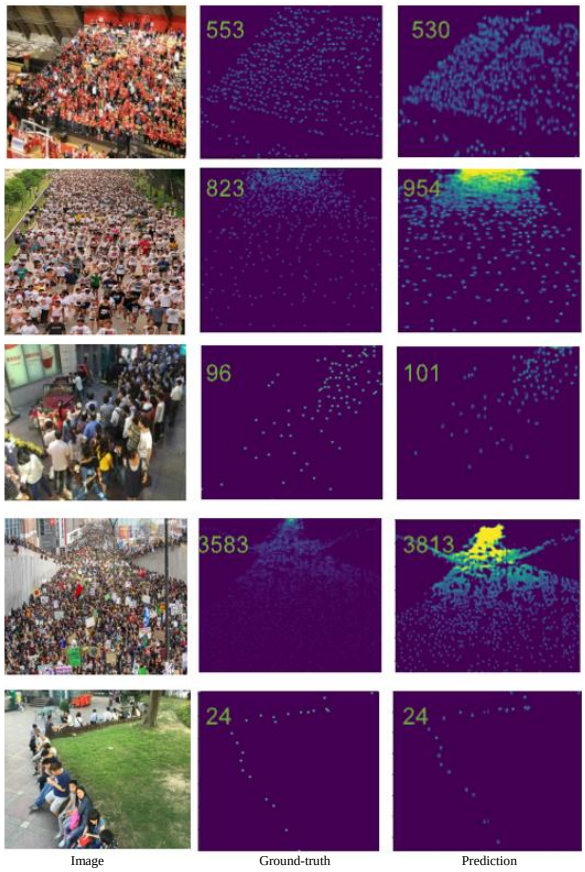

Figure 4. A density map predicted by the model based on images containing various crowd densities, ranging from large to medium to small.

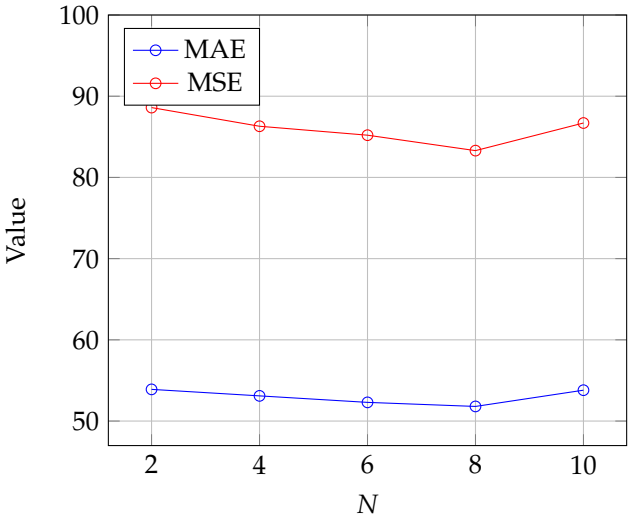

Figure 5. Impact of the choice of N on the ShanghaiTech Part A dataset.

4.6.2. Effect of HCDETR

The impact on performance of DETR with Hybrid Context is listed in Table 3. Upon analyzing and comparing the above results, it is apparent that different scales of information can capture features at various levels of granularity. Larger-scale information aids in capturing global contextual details, while smaller-scale information focuses more on local details. By introducing multiple scales of information into the model, a broader and richer receptive field can be achieved, enhancing the model's understanding of complex scenes. This promotes the updating of the decoder.

Table 3. Impact of DETR with Hybrid Context on the ShanghaiTech Part A dataset. MAE: Mean Absolute Error; MSE: Mean Square Error.

Hybrid Context	MAE	MSE
×	54.8	92.7
✓	51.8	83.3

4.6.3. Effect of CDCM

To validate the effectiveness of the CDCM module, we conducted comparative experiments with the methods listed in Table 4. It is obvious that that both coordinate encoding and dilated convolution contribute significantly to the enhancement of counting tasks. Coordinate encoding enables the network to learn more flexible receptive fields, as each position contains distinct coordinate information. This adaptability aids the network in better accommodating the structure and shape of the data. Meanwhile, dilated convolution allows the convolution operations to capture a broader range of input data, facilitating a more comprehensive understanding of the global structure. The combined effect of both significantly promotes the efficiency of model learning.

Table 4. The impact of coordinate encoding and dilated convolution on the ShanghaiTech Part A dataset. MAE: Mean Absolute Error; MSE: Mean Square Error.

Method	MAE	MSE
Coord	54.4	91.5
Dilated Conv	53.6	86.7
Coord + Dilated Conv	51.8	83.3

4.6.4. Combined Effects of HCDETR and CDCM

In this section, we compare the combined effects of the HCDETR and CDCM modules on the proposed CC-DETR. We contrast the individual impact of each module and evaluate the combined impact when both modules are applied, as outlined in Table 5, which demonstrates that the proposed to modules enhance each other.

Table 5. Effects of HCDETR and CDCM on the ShanghaiTech Part A dataset. MAE: Mean Absolute Error; MSE: Mean Square Error.

HCDETR	CDCM	MAE	MSE
×	✓	53.3	86.9
✓	×	53.8	88.6
✓	✓	51.8	83.3

4.6.5. Effect of Hyperparameters

During the loss computation process, determining the balance of weights for the three losses is crucial. Thus, we conducted comparative experiments, as outlined in Table 6, to explore the influence of hyperparameters on this weighting scheme. It is obvious that the best performance is achieved when $\lambda_1 = 0.1$ and $\lambda_2 = 0.01$.

Table 6. Effects of HCDETR and CDCM on the ShanghaiTech Part A dataset. MAE: Mean Absolute Error; MSE: Mean Square Error.

HCDETR	CDCM	MAE	MSE
×	✓	53.3	86.9
✓	×	53.8	88.6
✓	✓	51.8	83.3

4.7. Complexity and Efficiency Analysis

Because the backbone used in our CC-DETR is Twins-large [27], CCTrans [23], which also employs Twins-large as its backbone, was utilized as a comparative baseline for complexity and computational efficiency. Specifically, we compared the two methods along two dimensions: the number of model parameters, and the training iterations required to achieve optimal performance. The results are shown in Table 7, where it can be seen that the proposed CC-DETR exhibits superior performance compared to the baseline, with a remarkable two-thirds reduction in training time despite a one-third increase in parameters. This improvement in efficiency is complemented by significant enhancements in performance metrics, validating our model's efficacy. Through comprehensive evaluations across various datasets, our model demonstrates robustness and generalization capabilities, showcasing its applicability to a wide range of scenarios.

Table 7. The number of model parameters (Mb) and number of iterations (Epoch) required to obtain the optimal model. MAE: Mean Absolute Error; MSE: Mean Square Error. The best results were shown in boldface.

Method	Backbone	Parameters	Epoch	Part A		Part B	
				MAE	MSE	MAE	MSE
CCTrans [23]	Twins-large	**104 M**	1500	52.3	84.9	6.2	9.9
CC-DETR (ours)	Twins-large	154 M	**500**	**51.8**	**83.3**	**6.1**	**9.7**

4.8. Limitations

Although the quantitative and qualitative experimental results presented above fully demonstrate the effectiveness of the proposed CC-DETR, significant flaws still exist in its practical application. Due to the utilization of the main backbone from the DETR object detection model in our proposed CC-DETR, which consists of a large number of transformer encoding and decoding layers, the enormous parameter size and high computational complexity of the end-to-end training method and attention mechanism result in relatively slower training and inference speeds compared to convolutional neural networks. This may limit its applicability on resource-constrained edge computing devices.

5. Conclusions

In this paper, we propose a DETR-based crowd counting framework (CC-DETR). We first adopt a backbone network with alternating local and global attention mechanisms. Inspired by the DETR architecture, our encoder–decoder structure improves feature fusion ability through the proposed HCDETR module. Our regression head provides multiscale receptive fields through the proposed CCDM module. Extensive experiments on three crowd counting datasets demonstrate that our method significantly reduces background interference and scale effects. We hope that our work can help to solve the problem of crowd counting and provide new insights for future research. For example, crowd counting can be utilized in public safety and security applications such as monitoring crowd density in public spaces, transportation hubs, or large events to prevent overcrowding and ensure that safety protocols are followed. In the future, crowd counting could aid in monitoring patient waiting times, optimizing staffing levels in hospitals and clinics, and implementing social distancing measures during pandemics or outbreaks.

Author Contributions: Methodology, Y.H.; Software, T.Z.; Validation, F.N.; Writing—original draft, Y.G. All authors have read and agreed to the published version of the manuscript.

Funding: This work was supported in part by the Anhui Provincial Key Research and Development Program (No. 2022a05020042), Natural Science Research Project of Anhui Educational Committee (No. 2022AH051783), Anhui International Joint Research Center for Ancient Architecture Intellisencing and Multi-Dimensional Modeling (No. GJZZX2021KF01), and National Natural Science Foundation of China (No. 62105002).

Data Availability Statement: The code for reproducing our results is available at: https://github.com/ZZZ429/CC-DETR, accessed on 15 January 2024.

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

References

1. Deng, L.; Zhou, Q.; Wang, S.; Górriz, J.M.; Zhang, Y. Deep learning in crowd counting: A survey. *CAAI Trans. Intell. Technol.* **2023**, 1–35, *early view status*.
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. Lin, T.Y.; Dollár, P.; Girshick, R.; He, K.; Hariharan, B.; Belongie, S. Feature pyramid networks for object detection. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, HI, USA, 21–26 July 2017; pp. 2117–2125.
4. Savner, S.S.; Kanhangad, V. CrowdFormer: Weakly-supervised crowd counting with improved generalizability. *J. Vis. Commun. Image Represent.* **2023**, *94*, 103853. [CrossRef]
5. Vaswani, A.; Shazeer, N.; Parmar, N.; Uszkoreit, J.; Jones, L.; Gomez, A.N.; Kaiser, Ł.; Polosukhin, I. Attention is all you need. *Adv. Neural Inf. Process. Syst.* **2017**, *30*, 15. [CrossRef]
6. Dosovitskiy, A.; Beyer, L.; Kolesnikov, A.; Weissenborn, D.; Zhai, X.; Unterthiner, T.; Dehghani, M.; Minderer, M.; Heigold, G.; Gelly, S.; et al. An image is worth 16x16 words: Transformers for image recognition at scale. *arXiv* **2020**, arXiv:2010.11929.
7. Lin, H.; Ma, Z.; Ji, R.; Wang, Y.; Hong, X. Boosting crowd counting via multifaceted attention. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, New Orleans, LA, USA, 19–20 June 2022; pp. 19628–19637.
8. Yang, S.; Guo, W.; Ren, Y. CrowdFormer: An overlap patching vision transformer for top-down crowd counting. In Proceedings of the Thirty-First International Joint Conference on Artificial Intelligence, Vienna, Austria, 23–29 July 2022; pp. 23–29.
9. Carion, N.; Massa, F.; Synnaeve, G.; Usunier, N.; Kirillov, A.; Zagoruyko, S. End-to-end object detection with transformers. In Proceedings of the European Conference on Computer Vision, Glasgow, UK, 23–28 August 2020; Springer: Berlin/Heidelberg, Germany, 2020; pp. 213–229.
10. Liu, R.; Lehman, J.; Molino, P.; Petroski Such, F.; Frank, E.; Sergeev, A.; Yosinski, J. An intriguing failing of convolutional neural networks and the coordconv solution. *Adv. Neural Inf. Process. Syst.* **2018**, *31*. [CrossRef]
11. Liu, Y.; Shi, M.; Zhao, Q.; Wang, X. Point in, box out: Beyond counting persons in crowds. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Long Beach, CA, USA, 15–20 June 2019; pp. 6469–6478.
12. Redmon, J.; Divvala, S.; Girshick, R.; Farhadi, A. You only look once: Unified, real-time object detection. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, NV, USA, 26 June–1 July 2016; pp. 779–788.
13. Pham, V.Q.; Kozakaya, T.; Yamaguchi, O.; Okada, R. Count forest: Co-voting uncertain number of targets using random forest for crowd density estimation. In Proceedings of the IEEE International Conference on Computer Vision, Santiago, Chile, 3–17 December 2015; pp. 3253–3261.
14. 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, 18–22 June 2018; pp. 1091–1100.
15. Ren, S.; He, K.; Girshick, R.; Sun, J. Faster r-cnn: Towards real-time object detection with region proposal networks. *Adv. Neural Inf. Process. Syst.* **2015**, *28*. [CrossRef]
16. 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.
17. Zeng, L.; Xu, X.; Cai, B.; Qiu, S.; Zhang, T. Multi-scale convolutional neural networks for crowd counting. In Proceedings of the 2017 IEEE International Conference on Image Processing (ICIP), Beijing, China, 17–20 September 2017; pp. 465–469.
18. Shu, W.; Wan, J.; Tan, K.C.; Kwong, S.; Chan, A.B. Crowd counting in the frequency domain. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, New Orleans, LA, USA, 18–24 June 2022; pp. 19618–19627.
19. Liu, W.; Salzmann, M.; Fua, P. Context-aware crowd counting. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Long Beach, CA, USA, 15–20 June 2019; pp. 5099–5108.
20. Yan, Z.; Yuan, Y.; Zuo, W.; Tan, X.; Wang, Y.; Wen, S.; Ding, E. Perspective-guided convolution networks for crowd counting. In Proceedings of the IEEE/CVF International Conference on Computer Vision, Seoul, Republic of Korea, 27 October–2 November 2019; pp. 952–961.
21. Sun, G.; Liu, Y.; Probst, T.; Paudel, D.P.; Popovic, N.; Van Gool, L. Boosting crowd counting with transformers. *arXiv* **2021**, arXiv:2105.10926.
22. Du, Z.; Shi, M.; Deng, J.; Zafeiriou, S. Redesigning multi-scale neural network for crowd counting. *IEEE Trans. Image Process.* **2023**, *32*, 3664–3678. [CrossRef] [PubMed]
23. Tian, Y.; Chu, X.; Wang, H. Cctrans: Simplifying and improving crowd counting with transformer. *arXiv* **2021**, arXiv:2109.14483.
24. Liang, D.; Xu, W.; Bai, X. An end-to-end transformer model for crowd localization. In Proceedings of the European Conference on Computer Vision, Tel Aviv, Israel, 23–27 October 2022; Springer: Berlin/Heidelberg, Germany, 2022; pp. 38–54.

25. 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]
26. 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, BC, Canada, 11–17 October 2021; pp. 10012–10022.
27. 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.
28. Fang, Y.; Li, Y.; Tu, X.; Tan, T.; Wang, X. Face completion with hybrid dilated convolution. *Signal Process. Image Commun.* **2020**, *80*, 115664. [CrossRef]
29. Wang, B.; Liu, H.; Samaras, D.; Nguyen, M.H. Distribution matching for crowd counting. *Adv. Neural Inf. Process. Syst.* **2020**, *33*, 1595–1607.
30. Song, Q.; Wang, C.; Jiang, Z.; Wang, Y.; Tai, Y.; Wang, C.; Li, J.; Huang, F.; Wu, Y. Rethinking counting and localization in crowds: A purely point-based framework. In Proceedings of the IEEE/CVF International Conference on Computer Vision, Montreal, BC, Canada, 11–17 October 2021; pp. 3365–3374.
31. 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.
32. 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]
33. Xu, C.; Qiu, K.; Fu, J.; Bai, S.; Xu, Y.; Bai, X. Learn to scale: Generating multipolar normalized density maps for crowd counting. In Proceedings of the IEEE/CVF International Conference on Computer Vision, Seoul, Republic of Korea, 27 October–2 November 2019; pp. 8382–8390.
34. Lei, Y.; Liu, Y.; Zhang, P.; Liu, L. Towards using count-level weak supervision for crowd counting. *Pattern Recognit.* **2021**, *109*, 107616. [CrossRef]
35. Hu, Y.; Jiang, X.; Liu, X.; Zhang, B.; Han, J.; Cao, X.; Doermann, D. Nas-count: Counting-by-density with neural architecture search. In Proceedings of the Computer Vision—ECCV 2020: 16th European Conference, Glasgow, UK, 23–28 August 2020; Proceedings, Part XXII 16; Springer: Berlin/Heidelberg, Germany, 2020; pp. 747–766.
36. Wan, J.; Wang, Q.; Chan, A.B. Kernel-based density map generation for dense object counting. *IEEE Trans. Pattern Anal. Mach. Intell.* **2020**, *44*, 1357–1370. [CrossRef] [PubMed]
37. Liu, L.; Qiu, Z.; Li, G.; Liu, S.; Ouyang, W.; Lin, L. Crowd counting with deep structured scale integration network. In Proceedings of the IEEE/CVF International Conference on Computer Vision, Seoul, Republic of Korea, 27 October–2 November 2019; pp. 1774–1783.
38. Liu, L.; Lu, H.; Zou, H.; Xiong, H.; Cao, Z.; Shen, C. Weighing counts: Sequential crowd counting by reinforcement learning. In Proceedings of the Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, 23–28 August 2020; Proceedings, Part X 16; Springer: Berlin/Heidelberg, Germany, 2020; pp. 164–181.
39. Liu, X.; Yang, J.; Ding, W.; Wang, T.; Wang, Z.; Xiong, J. Adaptive mixture regression network with local counting map for crowd counting. In Proceedings of the Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, 23–28 August 2020; Proceedings, Part XXIV 16. Springer: Berlin/Heidelberg, Germany, 2020; pp. 241–257.
40. Wang, Q.; Gao, J.; Lin, W.; Yuan, Y. Learning from synthetic data for crowd counting in the wild. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Long Beach, CA, USA, 15–20 June 2019; pp. 8198–8207.

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.