Rethinking Counting and Localization in Crowds: A Purely Point-Based Framework

Qingyu Song[1*] Changan Wang[1*] Zhengkai Jiang[1] Yabiao Wang[1]
Ying Tai[1] Chengjie Wang[1] Jilin Li[1] Feiyue Huang[1†] Yang Wu[2]
[1]Tencent Youtu Lab, [2]Applied Research Center (ARC), Tencent PCG
qingyusong@zju.edu.cn, {changanwang, zhengkjiang, caseywang}@tencent.com
{yingtai, jasoncjwang, jerolinli, garyhuang, dylanywu}@tencent.com

Abstract

Localizing individuals in crowds is more in accordance with the practical demands of subsequent high-level crowd analysis tasks than simply counting. However, existing localization based methods relying on intermediate representations (i.e., density maps or pseudo boxes) serving as learning targets are counter-intuitive and error-prone. In this paper, we propose a purely point-based framework for joint crowd counting and individual localization. For this framework, instead of merely reporting the absolute counting error at image level, we propose a new metric, called density Normalized Average Precision (nAP), to provide more comprehensive and more precise performance evaluation. Moreover, we design an intuitive solution under this framework, which is called Point to Point Network (P2PNet). P2PNet discards superfluous steps and directly predicts a set of point proposals to represent heads in an image, being consistent with the human annotation results. By thorough analysis, we reveal the key step towards implementing such a novel idea is to assign optimal learning targets for these proposals. Therefore, we propose to conduct this crucial association in an one-to-one matching manner using the Hungarian algorithm. The P2PNet not only significantly surpasses state-of-the-art methods on popular counting benchmarks, but also achieves promising localization accuracy. The codes will be available at: TencentYoutuResearch/CrowdCounting-P2PNet.

1. Introduction

Among all the related concrete tasks of crowd analysis, crowd counting is a fundamental pillar, aiming to estimate the number of individuals in a crowd. However, simply giving a single number is obviously far from being able to support the practical demands of subsequent higher-level crowd analysis tasks, such as crowd tracking, activity recognition, abnormality detection, flow/behavior prediction, *etc*.

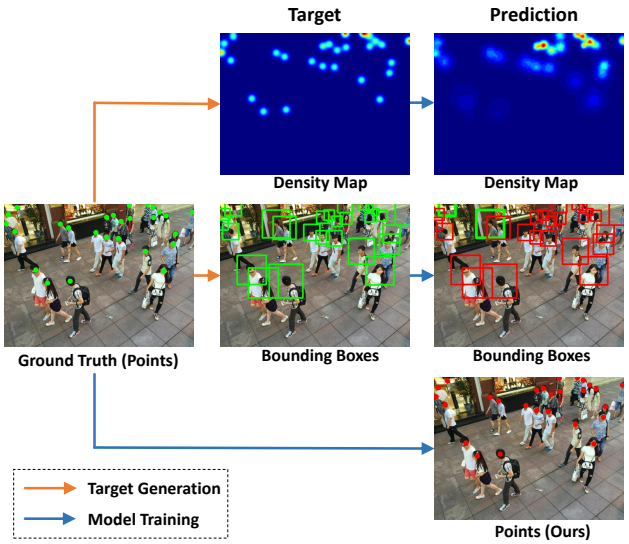

Figure 1. Illustrations for the comparison of our pipeline with existing methods, in which the predictions are marked in Red while the ground truths are marked as Green. **Top flow:** The dominated density map learning based methods fail to provide the exact locations of individuals. **Middle flow:** The estimated inaccurate ground truth bounding boxes make the detection based methods error-prone, such as the missing detections as indicated, especially for the NMS-like process. **Bottom flow:** Our pipeline directly predicts a set of points to represent the locations of individuals, which is simple, intuitive and competitive as demonstrated, bypassing those error-prone steps. Best viewed in color.

In fact, there is an obvious tendency in this field for more challenging fine-grained estimation (*i.e.*, the locations of individuals) beyond simply counting. Specifically, some approaches cast crowd counting as a head detection problem, but leaving more efforts on labor-intensive annotation for tiny-scale heads. Other approaches [26, 30] attempted to generate the pseudo bounding boxes of heads with only

*Equal contribution. †Corresponding author.

point annotations provided, which however appears to be tricky or inaccurate at least. Also trying to directly locate individuals, several methods [14, 20] got stuck in suppressing or splitting over-close instance candidates, making themselves error-prone due to the extreme head scale variation, especially for highly-congested regions. To eschew the above problems, we propose a purely point-based framework for jointly counting and localizing individuals in crowds. This framework directly uses point annotations as learning targets and simultaneously outputs points to locate individuals, benefiting from the high-precision localization property of point representation and its relatively cheaper annotation cost. The pipeline is illustrated in Figure 1.

Additionally, in terms of the evaluation metrics, some farsighted works [7, 32] encourage to adopt patch-level metrics for fine-grained evaluation, but they only provide a rough measure for localization. Other existing localization aware metrics either ignore the significant density variation across crowds [26, 30] or lack the punishment for duplicate predictions [30, 35]. Instead, we propose a new metric called density Normalized Average Precision (nAP) to provide a comprehensive evaluation metric for both localization and counting errors. The nAP metric supports both box and point representation as inputs (*i.e.*, predictions or annotations), without the defects mentioned above.

Finally, as an intuitive solution under this new framework, we develop a novel method to directly predict a set of point proposals with the coordinates of heads in an image and their confidences. Specifically, we propose a Point-to-Point Network (P2PNet) to directly receive a set of annotated head points for training and predict points too during inference. Then to make such an idea work correctly, we delve into the ground truth target assignation process to reveal the crucial of such association. The conclusion is that either the case when *multiple* proposals are matched to a *single* ground truth, or the opposite case, can make the model confused during training, leading to over-estimated or under-estimated counts. So we propose to perform an one-to-one matching by Hungarian algorithm to associate the point proposals with their ground truth targets, and the unmatched proposals should be classified as negatives. We empirically show that such a matching is beneficial to improving the nAP metric, serving as a key component for our solution under the new framework. This simple, intuitive and efficient design yields state-of-the-art counting performance and promising localization accuracy.

The major contributions of this work are three-fold:

1. We propose a purely point-based framework for joint counting and individual localization in crowds. This framework encourages fine-grained predictions, benefiting the practical demands of downstream tasks in crowd analysis.

2. We propose a new metric termed density Normalized Average Precision to account for the evaluation of both localization and counting, as a comprehensive evaluation metric under the new framework.

3. We propose P2PNet as an intuitive solution following this conceptually simple framework. The method achieves state-of-the-art counting accuracy and promising localization performance, and might also be inspiring for other tasks relying on point predictions.

2. Related Works

In this section, we review two kinds of crowd counting methods in recent literature. They are grouped according to whether locations of individuals could be provided. Since we focus on the estimation of locations, existing metrics accounting for localization errors are also discussed.

Density Map based Methods. The adoption of density map is a common choice of most state-of-the-art crowd counting methods, since it was firstly introduced in [15]. And the estimated count is obtained by summing over the predicted density maps. Recently, many efforts have been devoted to pushing forward the counting performance frontier of such methods. They either conduct a pixel-wise density map regression [16, 28, 11, 1, 25, 8], or resort to classify the count value of local patch into several bins [39, 21, 22]. Although many compelling models have been proposed, these density map learning based models still fail to provide the exact locations of individuals in crowds, not to mention their inherent flaws as pointed out in [1, 27, 21]. Whereas the proposed method goes beyond counting and focuses on the direct prediction for locations of individuals, eschewing the defects of density maps and also benefiting the downstream practical applications.

Localization based Methods. These methods typically achieve counting by firstly predicting the locations of individuals. Motivating by cutting-edge object detectors, some counting methods [17, 26, 30] try to predict the bounding boxes for heads of individuals. However, with only the point annotations available, these methods rely on heuristic estimation for ground truth bounding boxes, which is error-prone or even infeasible. These inaccurate bounding boxes not only confuse the model training process, but also make the post-process, *i.e.*, NMS, fail to suppress false detections. Without those inaccurate targets introduced, other methods locate individuals by points [20] or blobs [14], but leaving more efforts to remove duplicates or split over-close detected individuals in congest regions. Instead, bypassing these tricky post-processing with an one-to-one matching, we propose to streamline the framework to directly estimate the point locations of individuals.

Localization Aware Metrics. Traditional universally agreed evaluation metrics only measure the counting errors, entirely ignoring the significant spatial variation of estimation errors in single image. To provide a more accurate eval-

uation, some works [7, 23, 32] advocate to adopt patch-level or pixel-level absolute counting error as criteria, in lieu of the commonly used image-level metric. Other research [30] proposes Mean Localization Error to compute the average pixel distance between the predictions and ground truths, merely evaluating the localization errors. Inspired by evaluation metric used in object detection, [10] proposes to use the area under the Precision-Recall curve after a greedy association, which however ignores the punishment for duplicate predictions. Hence, [20] proposes to adopt a sequential matching and then use the standard Average Precision (AP) for evaluation. In this paper, we propose a new metric, termed density Normalized Average Precision (nAP), as a comprehensive evaluation metric for both localization errors and false detections. In particular, the nAP metric introduces a density normalization to account for the large density variation problem in crowds.

3. Our Work

We firstly introduce the proposed framework in detail (Sec. 3.1), and the new evaluation metric nAP is also presented (Sec. 3.2). Then we conduct a thorough analysis to reveal the key issue in improving the nAP metric under the new framework (Sec. 3.3). Inspired by the insightful analysis, we introduce the proposed P2PNet (Sec. 3.4), which directly predicts a set of point proposals to represent heads.

3.1. The Purely Point-based Framework

The proposed framework directly receives point annotations as its learning targets and then provides the exact locations for individuals in a crowd, rather than simply counting the number of individuals within it. And the locations of individuals are typically indicated by the center points of heads, possibly with optional confidence scores.

Formally, given an image with N individuals, we use $p_i = (x_i, y_i)$, $i \in \{1, .., N\}$, to represent the head's center point of the i-th individual, which is located in (x_i, y_i). Then the collection of the center points for all individuals could be further denoted as $\mathcal{P} = \{p_i | i \in \{1, .., N\}\}$. Assuming a well-designed model $\mathcal{M}$ is trained to instantiate this new framework. And the model $\mathcal{M}$ predicts another two collections $\hat{\mathcal{P}} = \{\hat{p}_j | j \in \{1, .., M\}\}$ and $\hat{\mathcal{C}} = \{\hat{c}_j | j \in \{1, .., M\}\}$, in which M is the number of predicted individuals, and $\hat{c}_j$ is the confidence score of the predicted point $\hat{p}_j$. Without loss of generality, we may assume that $\hat{p}_j$ is exactly the prediction for the ground truth point p_i. *Then our goal is to ensure that the distance between $\hat{p}_j$ and p_i is as close as possible with a sufficiently high score $\hat{c}_j$.* As a byproduct, the number of predicted individuals M should also be close enough to the ground truth crowd number N. In a nutshell, the new framework could simultaneously achieve crowd counting and individual localization.

Compared with traditional counting methods, the individual locations provided by this framework are helpful to those motion based crowd analysis tasks, such as crowd tracking [42], activity recognition [6], abnormality detection [3], etc. Besides, without relying on labor-intensive annotations, inaccurate pseudo boxes or tricky post-processing, this framework benefits from the high-precision localization property of original point representation, especially for highly-congested regions in crowds.

Therefore, this new framework is worth more attentions due to its advantages and practical values over traditional crowd counting. However, since the existence of severe occlusions, density variations, and annotation errors, it is quite challenging to tackle with such a task [20, 26, 30], which even is considered as ideal but infeasible in [10].

3.2. Density Normalized Average Precision

It is natural to ask that how to evaluate the performance of model $\mathcal{M}$ under the above new framework. In fact, a well-performed model following this framework should not only produce as few as false positives or false negatives, but also achieve competitive localization accuracy. Therefore, motivated by the mean Average Precision (mAP) [19] metric widely used in Object Detection, we propose a density Normalized Average Precision (nAP) to evaluate both the localization errors and counting performance.

The nAP is calculated based on the Average Precision, which is the area under the Precision-Recall (PR) curve. And the PR curve could be easily obtained by accumulating a binary list following the common practice in [19]. In the binary list, a True Positive (TP) prediction is indicated by 1, and a False Positive (FP) prediction is indicated by 0. Specifically, given all predicted head points $\hat{\mathcal{P}}$, we firstly sort the point list with their confidence scores from high to low. Then we sequentially determine that the point under investigation is either TP or FP, according to a pre-defined density aware criterion. Different from the greedy association used in [10, 30], we apply a sequential association in which those higher scored predictions are associated firstly. In this way, these TP predictions could be easily obtained by a simple threshold filtering during inference.

We introduce our density aware criterion as follows. A predicted point $\hat{p}_j$ is classified as TP only if it could be matched to certain ground truth p_i, in which p_i must not be matched before by any higher-ranked point. The matching process is guided by a pixel-level Euclidean distance based criterion $\mathbb{1}(\hat{p}_j, p_i)$. However, directly using the pixel distance to measure the affinity ignores the side effects from the large density variation across crowds. Thus, we introduce a density normalization for this matching criterion to mitigate the density variation problem. The density around a certain ground truth point is estimated following [41].

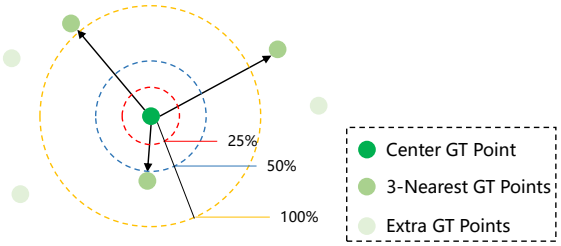

Figure 2. Illustration for different levels of localization accuracy in nAP (k=3). The yellow circle indicates the region within $d_{k\text{NN}}(p_i)$ pixels from the center GT point p_i. A typical value for δ is 0.5, as indicated by the blue circle, which means that the nearest GT point of most pixels within this region should be p_i. The red circle represents a threshold (δ=0.25) for stricter localization accuracy.

Formally, the final criterion used in nAP is defined as:

$$\mathbb{1}(\hat{p}_j, p_i) = \begin{cases} 1, & \text{if } d(\hat{p}_j, p_i)/d_{k\text{NN}}(p_i) < \delta, \\ 0, & \text{otherwise}, \end{cases} \quad (1)$$

where $d(\hat{p}_j, p_i) = ||\hat{p}_j - p_i||_2$ denotes to the Euclidean distance, and $d_{k\text{NN}}(p_i)$ denotes the average distance to the k nearest neighbors of p_i. We use a threshold δ to control the desired localization accuracy, as shown in Figure 2.

3.3. Our Approach

Our approach is an intuitive solution following the proposed framework, which directly predict a set of point proposals to represent the center points for heads of individuals. In fact, the idea of point prediction is not new to the vision community, although it is quite different here. To name a few, in the field of pose estimation, some methods adopt heatmap regression [4, 37] or direct point regression [33, 38] to predict the locations of pre-defined keypoints. Since the number of the keypoints to be predicted is fixed, the learning targets for these point proposals could be determined entirely before the training. Differently, the proposed framework aims to predict a point set of unknown size and is an open-set problem by nature [39]. Thus, one crucial problem of such a methodology is to determine which ground truth point should the current prediction be responsible for.

We propose to solve this key problem with a mutually optimal one-to-one association strategy during the training stage. Let us conduct a thorough analysis to show the defects of the other two strategies for the ground truth targets assignment. Firstly, for each ground truth point, the proposal with the nearest distance should produce the best prediction. However, if we select the nearest proposal for every ground truth point, it is likely that one proposal might be matched to multiple ground truth points, as shown in Figure 3 (a). In such a case, only one ground truth could be correctly predicted, leading to under-estimated counts, especially for the congested regions. Secondly, for each point proposal, we may assign the nearest ground truth point as its target. Intuitively, this strategy might be helpful to alleviate the overall overhead of the optimization, since the nearest ground truth point is relatively easier to predict. However, in such an assignment, there may exist multiple proposals which simultaneously predict the same ground truth, as shown in Figure 3 (b). Because there are no scale annotations available, it is tricky to suppress these duplicate predictions, which might lead to over-estimated. Consequently, the association process should take both sides into consideration and produces the mutually optimal one-to-one matching results, as shown in Figure 3 (c).

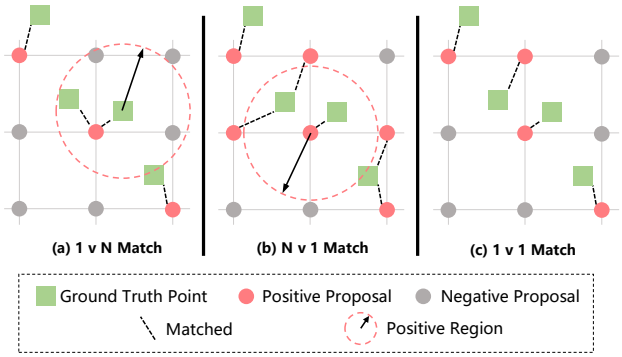

Figure 3. (a) Multiple ground truth points might be matched to the same proposal when selecting the nearest proposal for each of them, which leads to under-estimated counts. (b) Multiple proposals might be matched to the same ground truth point when selecting the nearest ground truth for each of them, which leads to over-estimated counts. (c) Our One-to-One match is without the above two defects, thus is suitable for direct point prediction.

Additionally, both the other two strategies have to determine a negative threshold, and the proposals whose distance with their matched targets are above this threshold will be considered as negatives. While using the one-to-one matching, those unmatched proposals are automatically remained as negatives, without any hyperparameter introduced. *In a nutshell, the key to solve the open-set direct point prediction problem is to ensure a mutually optimal one-to-one matching between predicted and ground truth points.*

After the ground truth targets are obtained, these point proposals could be trained through an end-to-end optimization. Finally, the positive proposals should be pushed toward their targets, while those negative proposals would be simply classified as backgrounds. Since the point proposals are dynamically updated along with the training process, those proposals which have the potential to perform better could be gradually selected by the one-to-one matching to serve as the final predictions.

Actually, the distance used in above matching could be any other cost measure beyond pixel distance, such as a combination of confidence score and pixel distance. We empirically show that taking confidence scores of proposals into consideration during the one-to-one matching is helpful to improve the proposed nAP metric. Let us consider

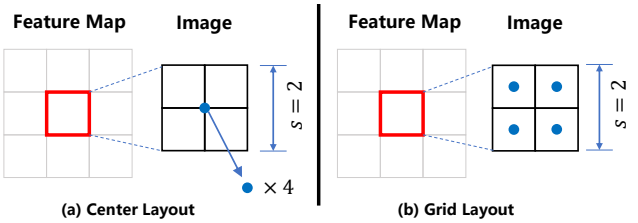

Figure 4. Two types of layout for reference points ($s = 2$, $K = 4$).

two predicted proposals around the same ground truth point p_i. If they have the same confidence score, the one closer to p_i should be matched as positive and encouraged to achieve higher localization accuracy. While the other one proposal should be matched as negative and supervised to lower its confidence, thus might not be matched again during next training iteration. On the contrary, if the two proposals share the same distance from p_i, the one with higher confidence should be trained to be closer to p_i with a much higher confidence. Both the above two cases would encourage the positive proposals to have more accurate locations as well as relatively higher confidences, which is beneficial to the improvement of nAP under the proposed framework.

3.4. The P2PNet Model

In this part, we present the detailed pipeline of the proposed Point to Point Network (P2PNet). Begining with the generation of point proposals, we introduce our one-to-one association strategy in detail. Then we present the loss function and the network architecture for the P2PNet.

Point Proposal Prediction. Let us denote the deep feature map outputted from the backbone network by $\mathcal{F}_s$, in which s is the downsampling stride and $\mathcal{F}_s$ is with a size of $H \times W$. Then based on $\mathcal{F}_s$, we adopt two parallel branches for point coordinate regression and proposal classification. For the classification branch, it outputs the confidence scores with a Softmax normalization. For the regression branch, it resorts to predict the offsets of the point coordinates due to the intrinsic translation invariant property of convolution layers. Specifically, each pixel on $\mathcal{F}_s$ should correspond to a patch of size $s \times s$ in the input image. In that patch, we firstly introduces a set of fixed reference points $\mathcal{R} = \{R_k | k \in \{1, ..., K\}\}$ with pre-defined locations $R_k = (x_k, y_k)$. These reference points could be either densely arranged on the patch or just set to the center of that patch, as shown in Figure 4. Since there are K reference points for each location on $\mathcal{F}_s$, the regression branch should produce totally $H \times W \times K$ point proposals. Assuming the reference point R_k predicts offsets $(\Delta^k_{jx}, \Delta^k_{jy})$ for its point proposal $\hat{p}_j = (\hat{x}_j, \hat{y}_j)$, then the coordinate of $\hat{p}_j$ is calculated as follows:

$$\begin{aligned} \hat{x}_j &= x_k + \gamma \Delta^k_{jx}, \\ \hat{y}_j &= y_k + \gamma \Delta^k_{jy}, \end{aligned} \tag{2}$$

where γ is a normalization term, which scales the offsets to rectify the relatively small predictions.

Proposal Matching. Following the symbols defined in Sec. 3.1, we assign the ground truth target from $\hat{\mathcal{P}}$ for every point proposal in $\mathcal{P}$ using an one-to-one matching strategy $\Omega(\mathcal{P}, \hat{\mathcal{P}}, \mathcal{D})$. The $\mathcal{D}$ is a pair-wise matching cost matrix with the shape $N \times M$, which measures the distance between two points in a pair. Instead of simply using the pixel distance, we also consider the confidence score of that proposal, since we encourage the positive proposals to have higher confidences. Formally, the cost matrix $\mathcal{D}$ is defined as follows:

$$\mathcal{D}(\mathcal{P}, \hat{\mathcal{P}}) = \left(\tau \left|\left|p_i - \hat{p}_j\right|\right|_2 - \hat{c}_j\right)_{i \in N, j \in M}, \tag{3}$$

where $||\cdot||_2$ denotes to the l_2 distance, and $\hat{c}_j$ is the confidence score of the proposal $\hat{p}_j$. τ is a weight term to balance the effect from the pixel distance.

Based on the pair-wise cost matrix $\mathcal{D}$, we conduct the association using the Hungarian algorithm [13, 29, 36] as the matching strategy Ω. Note that in our implementation, we ensure $M > N$ to produce many enough predictions, since those redundant proposals would be classified as negatives. From the perspective of the ground truth points, let us use a permutation ξ of $\{1, ..., M\}$ to represent the optimal matching result, *i.e.*, $\xi = \Omega(\mathcal{P}, \hat{\mathcal{P}}, \mathcal{D})$. That is to say, the ground truth point p_i is matched to the proposal $\hat{p}_{\xi(i)}$. Furthermore, those matched proposals (positives) could be represented as a set $\hat{\mathcal{P}}_{pos} = \{\hat{p}_{\xi(i)} | i \in \{1, ..., N\}\}$, and those unmatched proposals in the set $\hat{\mathcal{P}}_{neg} = \left\{\hat{p}_{\xi(i)} | i \in \{N+1, ..., M\}\right\}$ are labeled as negatives.

Loss Design. After the ground truth targets have been obtained, we calculate the Euclidean loss $\mathcal{L}_{loc}$ to supervise the point regression, and use Cross Entropy loss $\mathcal{L}_{cls}$ to train the proposal classification. The final loss function $\mathcal{L}$ is the summation of the above two losses, which is defined as:

$$\mathcal{L}_{cls} = -\frac{1}{M}\left\{\sum_{i=1}^{N} \log \hat{c}_{\xi(i)} + \lambda_1 \sum_{i=N+1}^{M} \log\left(1 - \hat{c}_{\xi(i)}\right)\right\}, \tag{4}$$

$$\mathcal{L}_{loc} = \frac{1}{N}\sum_{i=1}^{N} \left|\left|p_i - \hat{p}_{\xi(i)}\right|\right|_2^2, \tag{5}$$

$$\mathcal{L} = \mathcal{L}_{cls} + \lambda_2 \mathcal{L}_{loc}, \tag{6}$$

where $||\cdot||_{l_2}$ denotes to the Euclidean distance, λ_1 is a re-weight factor for negative proposals, and λ_2 is a weight term to balance the effect of the regression loss.

Network Design. As illustrated in Figure 5, we use the first 13 convolutional layers in VGG-16_bn [31] to extract deep features. With the outputted feature map, we upsample its spatial resolution by a factor of 2 using nearest neighbor interpolation. Then the upsampled map is merged with

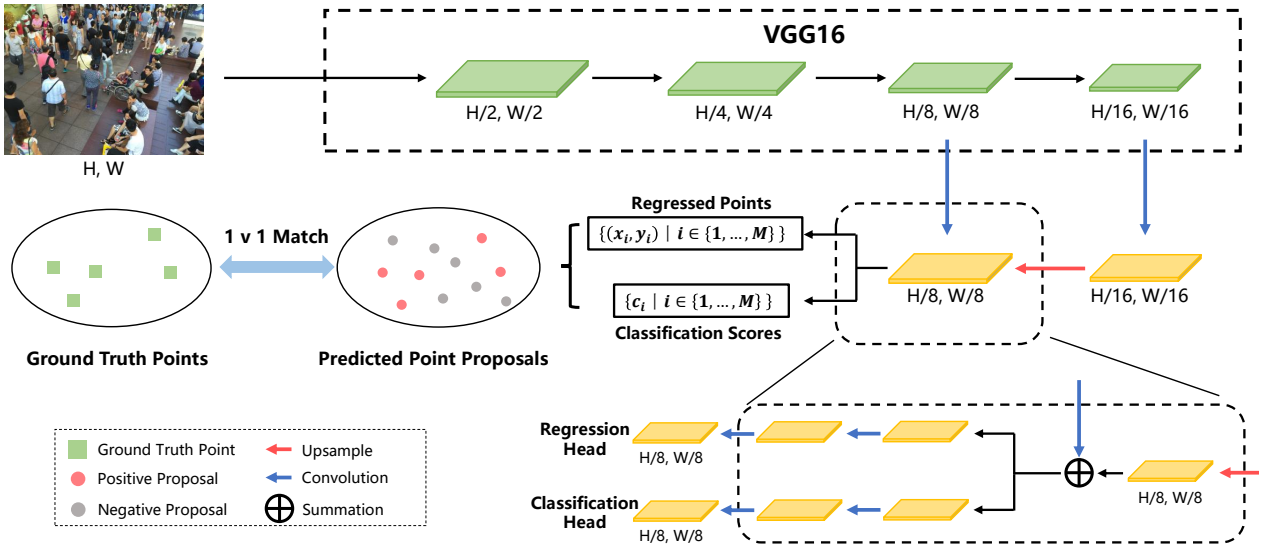

Figure 5. The overall architecture of the proposed P2PNet. Built upon the VGG16, it firstly introduce an upsampling path to obtain fine-grained deep feature map. Then it exploits two branches to simultaneously predict a set of point proposals and their confidence scores. The key step in our pipeline is to ensure an one-to-one matching between point proposals and ground truth points, which determines the learning targets of those proposals.

the feature map from a lateral connection by element-wise addition. This lateral connection is used to reduce channel dimensions of the feature map after the fourth convolutional blcok. Finally, the merged feature map undergoes a 3×3 convolutional layer to get $\mathcal{F}_s$, and the convolution in which is used to reduce the aliasing effect due to the upsampling.

The prediction head in our P2PNet is consisted of two branches, which are both fed with $\mathcal{F}_s$ and produce point locations and confidence scores respectively. For simplicity, the architecture of the two branches are kept same, which is consisted of three stacked convolutions interleaved with ReLU activations. We have empirically found this simple structure yield competitive results.

4. Experiments

4.1. Implementation Details

Dataset. We exploit existing publicly available datasets in crowd counting to demonstrate the superiority of our method. Specifically, extensive experiments are conducted on four challenging datasets, including ShanghaiTech PartA and PartB [41], UCF_CC_50 [9], UCF-QNRF [10] and NWPU-Crowd [35]. For experiments on UCF_CC_50, we conduct a five-fold cross validation following [9].

Data Augmentations. We firstly adopt random scaling with its scaling factor selected from [0.7, 1.3], keeping the shorter side not less than 128. Then we randomly crop an image patches with a fixed-size of 128×128 from the resized image. Finally, random flipping with a probability of 0.5 is also adopted. For the datasets containing extremely large resolution, *i.e.*, QNRF and NWPU-Crowd, we keep the max size of image no longer than 1408 and 1920, respectively, and keep the original aspect ratio.

Hyperparameters. We use the feature map of stride $s = 8$ for the prediction. The number K of the reference points is set to 4 (8 for QNRF dataset). And K is set according to the dataset statistics to ensure $M > N$. For the point regression, we set the γ to 100. The weight term τ during the matching is set as 5e-2. In the loss function, the λ_1 is set to 0.5, and λ_2 is set to 2e-4. Adam algorithm [12] with a fixed learning rate 1e-4 is used to optimize the model parameters. Since the weights in the backbone network have been pre-trained on the ImageNet, thus, we use a smaller learning rate 1e-5. The training batch size is set to 8.

4.2. Model Evaluation

As a comprehensive criteria, the proposed nAP metric is firstly reported to evaluate the performance of our P2PNet model. As shown in Table 1, the nAP is reported using three different thresholds of δ, which corresponds to the average precision under different localization accuracies of the predicted individual points. Typically, $\text{nAP}_{0.5}$ could satisfy the requirements of most practical applications, which means that the ground truth point is exactly the nearest neighbor for most points within this region. Besides, $\text{nAP}_{0.1}$ and $\text{nAP}_{0.25}$ are reported to account for some requirements of high localization accuracy. Following recent detection methods which report the average of AP under several thresholds to provide a single number for the overall performance, we adopt a similar metric. Specifically, we calculate multiple nAP_{δ} with the δ starting from 0.05 to 0.50, with steps of 0.05. Then an average is done to

nAP_δ	SHTech PartA	SHTech PartB	UCF_CC_50	UCF-QNRF	NWPU-Crowd
$\delta = 0.05$	10.9%	23.8%	5.0%	5.9%	12.9%
$\delta = 0.25$	70.3%	84.2%	54.5%	55.4%	71.3%
$\delta = 0.50$	90.1%	94.1%	88.1%	83.2%	89.1%
$\delta = \{0.05 : 0.05 : 0.50\}$	64.4%	76.3%	54.3%	53.1%	65.0%

Table 1. The overall performance of our P2PNet.

Methods	Venue	SHTech PartA		SHTech PartB		UCF_CC_50		UCF-QNRF	
		MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE
CAN [24]	CVPR'19	62.3	100.0	7.8	12.2	212.2	**243.7**	107.0	183.0
Bayesian+ [27]	ICCV'19	62.8	101.8	7.7	12.7	229.3	308.2	88.7	154.8
S-DCNet [39]	ICCV'19	58.3	95.0	6.7	10.7	204.2	301.3	104.4	176.1
SANet + SPANet [5]	ICCV'19	59.4	92.5	6.5	**9.9**	232.6	311.7	-	-
SDANet [28]	AAAI'20	63.6	101.8	7.8	10.2	227.6	316.4	-	-
ADSCNet [1]	CVPR'20	55.4	97.7	6.4	11.3	198.4	267.3	**71.3**	**132.5**
ASNet [11]	CVPR'20	57.78	90.13	-	-	174.84	251.63	91.59	159.71
AMRNet [25]	ECCV'20	61.59	98.36	7.02	11.00	184.0	265.8	86.6	152.2
AMSNet [8]	ECCV'20	56.7	93.4	6.7	10.2	208.4	297.3	101.8	163.2
DM-Count [34]	NeurIPS'20	59.7	95.7	7.4	11.8	211.0	291.5	85.6	148.3
Ours	-	**52.74**	**85.06**	**6.25**	**9.9**	**172.72**	256.18	85.32	154.5

Table 2. Comparison of the counting accuracy with state-of-the-art methods.

get the overall average precision $nAP_{\{0.05:0.05:0.50\}}$.

From the Table 1, we observe that our P2PNet achieves a promising average precision under different levels of localization accuracy. Specifically, its overall metric $nAP_{\{0.05:0.05:0.50\}}$ is around 60% on all datassets, which should already meet the requirements of many practical applications. In terms of the primary indicator $nAP_{0.5}$, the P2PNet generally achieves a promising precision of more than 80%. For most datasets, the P2PNet could achieve a $nAP_{0.5}$ of nearly 90%, which demonstrates the effectiveness of our approach on individual localization. Even for the stricter metric $nAP_{0.25}$, the precision is still higher than 55%. These results are encouraging, since we did not use any techniques like coordinate refinement in [2, 40] or exploiting multiple feature levels [18], which are both orthogonal to our contributions and should bring more improvements. Besides, the P2PNet achieves a relatively lower precision on the $nAP_{0.05}$, which is reasonable since the effects of the labeling deviations might gradually become apparent under such high localization accuracy.

Besides, we also notice that the NWPU-Crowd dataset [35] provides scarce yet valuable box annotations, so we report our localization performance using their metrics to compare with other competitors. And our P2PNet achieves an F1-measure/Precision/Recall of 71.2%/72.9%/69.5%, which is the best among published methods with similar backbones. For other localization based methods with official codes available, we also report their results in nAP metric (much lower than ours) in **Supplementary**.

Furthermore, we also evaluate the counting accuracy of our model. The estimated crowd number of our P2PNet is obtained by counting the predicted points with confidence scores higher than 0.5. We compare the P2PNet with state-of-the-art methods on several challenging datasets with various densities. Similar to [41], we also adopt Mean Absolute Error (MAE) and Mean Squared Error (MSE) as the evaluation metrics. The results are illustrated in Table 2 and Table 3. The top performance is indicated by bold numbers and the second best is indicated by underlined numbers.

Methods	NWPU-Crowd			
	MAE[O]	MSE[O]	MAE[L]	MAE[S]
CSRNet [16]	121.3	387.8	112.0	522.7
Bayesian+ [27]	105.4	454.2	115.8	750.5
S-DCNet [39]	90.2	370.5	**82.9**	567.8
DM-Count [34]	88.4	388.6	88.0	**498.0**
Ours	**77.44**	**362**	83.28	553.92

Table 3. Comparison on the NWPU-Crowd dataset.

ShanghaiTech. There are two independent subsets in ShanghaiTech dataset: PartA and PartB. The PartA contains highly congested images collecting from the Internet. While the PartB is collected from a busy street and represents relatively sparse scenes. Our P2PNet achieves the best performance on both PartA and PartB. In particular, on the PartA, the P2PNet reduces the MAE by 4.8% and MSE by 12.9% respectively, compared with the second best method ADSCNet. For sparse scenes in PartB, the P2PNet could also bring a reduction of 2.3% in MAE.

UCF_CC_50. UCF_CC_50 has only 50 images collecting from the Internet, but contains complicated scenes with large variation of crowd numbers. As shown in Table 2, our P2PNet surpasses all the other methods, reducing the MAE by 2.1 compared with the second best performance.

UCF-QNRF. UCF-QNRF is a challenging dataset due to

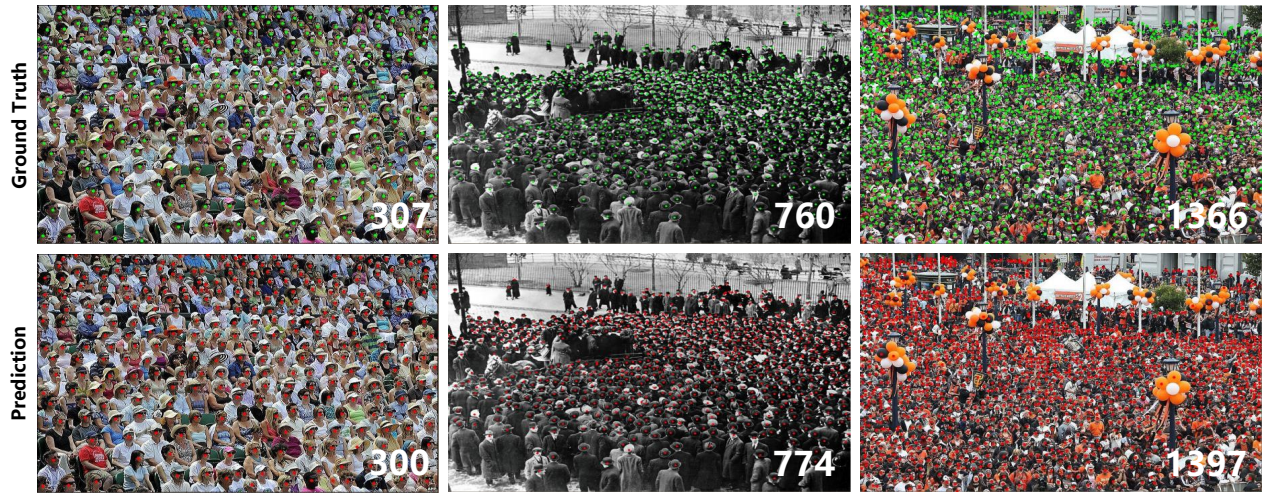

Figure 6. Some qualitative results for the predicted individuals of our P2PNet. The white numbers denote to the ground truth or prediction counts. The visualizations demonstrate the superiority of our method under various densities in terms of both localization and counting.

the much wider range of counts. As shown in Table 2, our P2PNet achieves an MAE of 85.32, which is much better than the Neural Architecture Search based method AMSNet. Compared with the previous best method ADSCNet, although the accuracy of our method is not so competitive, it is still much higher than ADSCNet on all other datasets. Besides, among all the methods in Table 2, only ours could provide exact individual locations.

NWPU-Crowd. The NWPU-Crowd dataset is a large-scale congested dataset recently introduced in [35]. As shown in Table 3, our P2PNet achieves the best overall MAE, with a reduction of 12.4% compared with the second best method DM-Count. Since our predictions are only based on a single scale feature map for simplicity, the result is slightly lower than those best performance on MAE[S]. MAE[S] is the average MAE of different scale levels, please refer to [35].

4.3. Ablation Studies

Layout	MAE	MSE	nAP$_\delta$
Center	53.7	89.61	61.7
Grid	**52.74**	**85.06**	**64.4**

Table 4. The effect of the layout for reference points. For an overall comparison, we use $\delta = \{0.05 : 0.05 : 0.50\}$.

Layout of reference points. We firstly evaluate the effect from the layouts of the reference points. As shown in Table 4, we compare two layouts in the Figure 4. Generally speaking, both the two layouts achieve state-of-the-art performance with minor difference, proving that the target association matters more than the layout of reference points. The Grid layout performs slightly better due to its dense arrangement of reference points, which is beneficial for the congested regions.

Effect of feature levels. We exhibit the effect of different feature levels used for prediction. For fair comparison, we keep the total reference points the same when using feature levels with different strides. As shown in Table 5, the

Method		MAE	MSE	nAP$_\delta$
P2PNet	$s = 4$	53.51	85.77	**66.8**
	$s = 8$	**52.74**	**85.06**	64.4
	$s = 16$	54.3	85.18	52.4

Table 5. The ablation study on SHTech PartA. For an overall comparison, we use $\delta = \{0.05 : 0.05 : 0.50\}$.

P2PNet consistently achieves competitive results using different feature levels, which demonstrates the effectiveness of our point based solution. In particular, the feature level with a stride of 8 provides a trade-off for the various densities, thus yields better performance.

In terms of the localization accuracy, we observe an obvious trend of improvement on nAP when we increase the feature map resolution, as shown in Table 5. It implies that the finest feature map is beneficial for localization, which is also in accord with the consensus on other tasks. Besides, based on our baseline method, it would be interesting to introduce existing multi-scale feature fusion techniques such as [18], which are discarded in our P2PNet for simplicity.

5. Conclusion

In this work, we go beyond crowd counting and propose a purely point-based framework to directly predict locations for crowd individuals. This new framework could better satisfy the practical demands of downstream tasks in crowd analysis. In conjunction with it, we advocate to use a new metric nAP for a more comprehensive accuracy evaluation on both localization and counting. Moreover, as an intuitive solution following this framework, we propose a novel network P2PNet, which is capable of directly taking point annotations as supervision whilst predicting the point locations during inference. P2PNet's key component is the one-to-one matching during the ground truth targets association, which is beneficial to the improvement of the nAP metric. This conceptually simple framework yields state-of-the-art counting performance and promising localization accuracy.

References

[1] Shuai Bai, Zhiqun He, Yu Qiao, Hanzhe Hu, Wei Wu, and Junjie Yan. Adaptive dilated network with self-correction supervision for counting. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2020. 2, 7

[2] Zhaowei Cai and Nuno Vasconcelos. Cascade r-cnn: Delving into high quality object detection. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2018. 7

[3] Xiao-Han Chen and Jian-Huang Lai. Detecting abnormal crowd behaviors based on the div-curl characteristics of flow fields. *Pattern Recognition*, 2019. 3

[4] Yilun Chen, Zhicheng Wang, Yuxiang Peng, Zhiqiang Zhang, Gang Yu, and Jian Sun. Cascaded pyramid network for multi-person pose estimation. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2018. 4

[5] Zhi-Qi Cheng, Jun-Xiu Li, Qi Dai, Xiao Wu, and Alexander G Hauptmann. Learning spatial awareness to improve crowd counting. In *IEEE International Conference on Computer Vision*, 2019. 7

[6] Camille Dupont, Luis Tobias, and Bertrand Luvison. Crowd-11: A dataset for fine grained crowd behaviour analysis. In *IEEE Conference on Computer Vision and Pattern Recognition Workshops*, 2017. 3

[7] Ricardo Guerrero-Gómez-Olmedo, Beatriz Torre-Jiménez, Roberto López-Sastre, Saturnino Maldonado-Bascón, and Daniel Onoro-Rubio. Extremely overlapping vehicle counting. In *Iberian Conference on Pattern Recognition and Image Analysis*, 2015. 2, 3

[8] Yutao Hu, Xiaolong Jiang, Xuhui Liu, Baochang Zhang, Jungong Han, Xianbin Cao, and David Doermann. Nas-count: Counting-by-density with neural architecture search. In *European Conference on Computer Vision*, 2020. 2, 7

[9] Haroon Idrees, Imran Saleemi, Cody Seibert, and Mubarak Shah. Multi-source multi-scale counting in extremely dense crowd images. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2013. 6

[10] Haroon Idrees, Muhmmad Tayyab, Kishan Athrey, Dong Zhang, Somaya Al-Maadeed, Nasir Rajpoot, and Mubarak Shah. Composition loss for counting, density map estimation and localization in dense crowds. In *European Conference on Computer Vision*, 2018. 3, 6

[11] Xiaoheng Jiang, Li Zhang, Mingliang Xu, Tianzhu Zhang, Pei Lv, Bing Zhou, Xin Yang, and Yanwei Pang. Attention scaling for crowd counting. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2020. 2, 7

[12] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *International Conference for Learning Representations*, 2014. 6

[13] Harold W Kuhn. The hungarian method for the assignment problem. *Naval Research Logistics Quarterly*, 1955. 5

[14] Issam H Laradji, Negar Rostamzadeh, Pedro O Pinheiro, David Vazquez, and Mark Schmidt. Where are the blobs: Counting by localization with point supervision. In *European Conference on Computer Vision*, 2018. 2

[15] Victor Lempitsky and Andrew Zisserman. Learning to count objects in images. In *Advances in Neural Information Processing Systems*, 2010. 2

[16] Yuhong Li, Xiaofan Zhang, and Deming Chen. Csrnet: Dilated convolutional neural networks for understanding the highly congested scenes. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2018. 2, 7

[17] Dongze Lian, Jing Li, Jia Zheng, Weixin Luo, and Shenghua Gao. Density map regression guided detection network for rgb-d crowd counting and localization. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2019. 2

[18] Tsung-Yi Lin, Piotr Dollár, Ross Girshick, Kaiming He, Bharath Hariharan, and Serge Belongie. Feature pyramid networks for object detection. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2017. 7, 8

[19] Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. Microsoft coco: Common objects in context. In *European Conference on Computer Vision*, 2014. 3

[20] Chenchen Liu, Xinyu Weng, and Yadong Mu. Recurrent attentive zooming for joint crowd counting and precise localization. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2019. 2, 3

[21] Liang Liu, Hao Lu, Haipeng Xiong, Ke Xian, Zhiguo Cao, and Chunhua Shen. Counting objects by blockwise classification. *IEEE Transactions on Circuits and Systems for Video Technology*, 2019. 2

[22] Liang Liu, Hao Lu, Hongwei Zou, Haipeng Xiong, Zhiguo Cao, and Chunhua Shen. Weighing counts: Sequential crowd counting by reinforcement learning. In *European Conference on Computer Vision*, 2020. 2

[23] Weizhe Liu, Krzysztof Lis, Mathieu Salzmann, and Pascal Fua. Geometric and physical constraints for drone-based head plane crowd density estimation. In *International Conference on Intelligent Robots and Systems*, 2019. 3

[24] Weizhe Liu, Mathieu Salzmann, and Pascal Fua. Context-aware crowd counting. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2019. 7

[25] Xiyang Liu, Jie Yang, and Wenrui Ding. Adaptive mixture regression network with local counting map for crowd counting. In *European Conference on Computer Vision*, 2020. 2, 7

[26] Yuting Liu, Miaojing Shi, Qijun Zhao, and Xiaofang Wang. Point in, box out: Beyond counting persons in crowds. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2019. 1, 2, 3

[27] Zhiheng Ma, Xing Wei, Xiaopeng Hong, and Yihong Gong. Bayesian loss for crowd count estimation with point supervision. In *IEEE International Conference on Computer Vision*, 2019. 2, 7

[28] Yunqi Miao, Zijia Lin, Guiguang Ding, and Jungong Han. Shallow feature based dense attention network for crowd counting. In *Association for the Advancement of Artificial Intelligence*, 2020. 2, 7

[29] Stewart Russell, Andriluka Mykhaylo, and Andrew Y. Ng. End-to-end people detection in crowded scenes. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2016. 5

[30] Deepak Babu Sam, Skand Vishwanath Peri, Mukuntha Narayanan Sundararaman, Amogh Kamath, and

Venkatesh Babu Radhakrishnan. Locate, size and count: Accurately resolving people in dense crowds via detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2020. 1, 2, 3

[31] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*, 2014. 5

[32] Yukun Tian, Yiming Lei, Junping Zhang, and James Z Wang. Padnet: Pan-density crowd counting. *IEEE Transactions on Image Processing*, 2019. 2, 3

[33] Alexander Toshev and Christian Szegedy. Deeppose: Human pose estimation via deep neural networks. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2014. 4

[34] Boyu Wang, Huidong Liu, Dimitris Samaras, and Minh Hoai Nguyen. Distribution matching for crowd counting. *Advances in Neural Information Processing Systems*, 2020. 7

[35] Qi Wang, Junyu Gao, Wei Lin, and Xuelong Li. Nwpu-crowd: A large-scale benchmark for crowd counting and localization. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2020. 2, 6, 7, 8

[36] Zijun Wei, Wang Boyu, Hoai Minh, Zhang Jianming, Shen Xiaohui, Lin Zhe, Mech Radomir, and Samaras Dimitris. Sequence-to-segments networks for detecting segments in videos. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2019. 5

[37] Bin Xiao, Haiping Wu, and Yichen Wei. Simple baselines for human pose estimation and tracking. In *European Conference on Computer Vision*, 2018. 4

[38] Fu Xiong, Boshen Zhang, Yang Xiao, Zhiguo Cao, Taidong Yu, Joey Tianyi Zhou, and Junsong Yuan. A2j: Anchor-to-joint regression network for 3d articulated pose estimation from a single depth image. In *IEEE International Conference on Computer Vision*, 2019. 4

[39] Haipeng Xiong, Hao Lu, Chengxin Liu, Liang Liu, Zhiguo Cao, and Chunhua Shen. From open set to closed set: Counting objects by spatial divide-and-conquer. In *IEEE International Conference on Computer Vision*, 2019. 2, 4, 7

[40] Shifeng Zhang, Longyin Wen, Xiao Bian, Zhen Lei, and Stan Z Li. Single-shot refinement neural network for object detection. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2018. 7

[41] Yingying Zhang, Desen Zhou, Siqin Chen, Shenghua Gao, and Yi Ma. Single-image crowd counting via multi-column convolutional neural network. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2016. 3, 6, 7

[42] Feng Zhu, Xiaogang Wang, and Nenghai Yu. Crowd tracking by group structure evolution. *IEEE Transactions on Circuits and Systems for Video Technology*, 2016. 3