Point-to-Region Loss for Semi-Supervised Point-Based Crowd Counting

Wei Lin, Chenyang Zhao, and Antoni B. Chan
Department of Computer Science, City University of Hong Kong
elonlin24@gmail.com, chenyzhao9-c@my.cityu.edu.hk, abchan@cityu.edu.hk

Abstract

Point detection has been developed to locate pedestrians in crowded scenes by training a counter through a point-to-point (P2P) supervision scheme. Despite its excellent localization and counting performance, training a point-based counter still faces challenges concerning annotation labor: hundreds to thousands of points are required to annotate a single sample capturing a dense crowd. In this paper, we integrate point-based methods into a semi-supervised counting framework based on pseudo-labeling, enabling the training of a counter with only a few annotated samples supplemented by a large volume of pseudo-labeled data. However, during implementation, the training encounters issues as the confidence for pseudo-labels fails to be propagated to background pixels via the P2P. To tackle this challenge, we devise a point-specific activation map (PSAM) to visually interpret the phenomena occurring during the ill-posed training. Observations from the PSAM suggest that the feature map is excessively activated by the loss for unlabeled data, causing the decoder to misinterpret these over-activations as pedestrians. To mitigate this issue, we propose a point-to-region (P2R) scheme to substitute P2P, which segments out local regions rather than detects a point corresponding to a pedestrian for supervision. Consequently, pixels in the local region can share the same confidence with the corresponding pseudo points. Experimental results in both semi-supervised counting and unsupervised domain adaptation highlight the advantages of our method, illustrating P2R can resolve issues identified in PSAM. The code is available at https://github.com/Elin24/P2RLoss.

1. Introduction

Crowd scene analysis has been investigated for many years due to its significant applications in smart cities and urban safety [2, 7, 29, 41, 46, 62, 75]. Crowd counting and localization in extremely dense crowd scenes, as the fundamental task in this field, have attracted considerable attention from researchers in computer vision [4, 5, 37, 61, 66, 68]. Although most supervised crowd counting methods follow the scheme of density regression [10, 24, 38, 61, 73], there are also relevant studies implementing point detection for crowd localization and counting [21, 31, 57], due to its straightforwardness in indicating pedestrians' positions. These approaches use the Hungarian algorithm to perform point-to-point (P2P) matching between predicted point maps and ground-truth (GT) pedestrian annotations during training. Therefore, these matched pixels are considered as pedestrians' positions, while the others are designated as background for loss computation. This scheme successfully trains a point detector for crowd analysis, yielding both good localization and counting performance.

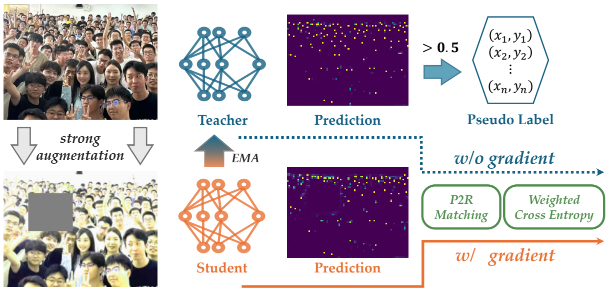


Figure 1. The workflow of semi-supervised point-based counting methods. The teacher model generates pseudo labels by extracting the foreground pixels, while the student model takes the corresponding strongly augmented image as input to construct the computation graph. The training loss between the pseudo label and the student's prediction involves two steps: the proposed P2R matching and the weighted cross-entropy computation.

Great progress has been made in supervised counting, but the labor-intensive nature of annotating data has hindered the development of crowd understanding for a long time [26, 66]. Addressing this issue, semi-supervised counting is gradually being developed, aiming to train a model using large amounts of unlabeled data alongside only a few annotated samples. Similar to supervised counting, most semi-supervised methods also follow the procedure of density regression. Notably, the method proposed in OT-M [26] stands out by employing a simple self-training scheme based on mean-teacher [59] and FixMatch [56]. However, there is no study exploring semi-supervised counting and localization using a point-detection method. To fill this gap, this paper explores an approach to train a point detector with limited annotated data.

As shown in Fig. 1, typical semi-supervised counting methods rely on pseudo-labeling, which generates pseudo labels to serve as learning targets for unlabeled samples using a model pre-trained on a small amount of annotated data. During training, there is a confidence score to determine whether a concerned prediction should be retained as a pseudo label. However, if the P2P matching strategy is applied to compute loss within the pipeline presented in Fig. 1, there is no reasonable way to define the confidence for *unmatched background* pixels, as P2P matching only cares about *matched foreground* pixels. Specifically, a foreground pixel can be retained for training if its matched pseudo-label's foreground probability exceeds a predefined threshold, similar to the approach used in semi-supervised classification tasks [15, 56]. However, background pixels lack confidence scores since no pseudo-labels are assigned to them after matching. Experimental results also demonstrate that ignoring the background leads to the breakdown of semi-supervised training.

To address the aforementioned issue, we propose a gradient-based visual explanation method (akin to [48, 74, 76, 78]), termed point-specific activation map (PSAM) , to visualize regions of interest and the activation levels of detected points. Using PSAM, we observe that the activation values and coverage areas of foreground pixels continuously grow during semi-supervised training, causing the model to misclassify neighboring pixels as foreground.

Inspecting PSAMs reveals that each point in GT is implicitly associated with a local region rather than a single pixel in the prediction. Accordingly, we propose to replace the P2P with a point-to-region (P2R) strategy, which matches local regions to points in the pseudo-label. This allows all pixels within a matched region, including background pixels, to share the confidence of the corresponding pseudo-point. Experimental results and visualizations demonstrate that P2R can effectively train a semi-supervised point-based counter and achieve expected performance. Experimental results show that P2R effectively trains semi-supervised point-based counters with comparable performance to P2P but at significantly reduced computational cost. Unlike P2P [57], P2R eliminates the need for the Hungarian algorithm by assigning each ground-truth or pseudo-point a local "dominated zone" and selecting a representative foreground pixel based on a predefined cost matrix.

The contributions of our paper are as follows:

- We formulate a pipeline of semi-supervised counting with a point-detection method, but find a problem leading the training breakdown.
- We propose Point-Specific Activation Map (PSAM) to visualize the regions of interest and the activation values of each point detected by the point-based crowd counter. Through PSAM, we observe that the surrounding pixels of each expected foreground pixel are over-activated, causing the decoder to recognizes these pixels as instances different from the concerned one.
- Based on insights from PSAM, we propose replacing the P2P with a P2R matching strategy for semi-supervised learning in point-based crowd counting. This not only makes the training process work as expected, but also brings additional benefits of reducing computational requirements.
- The proposed P2R achieves outstanding performance in several experiments, including semi-supervised counting and unsupervised domain adaptation. Ablation study shows that P2R excellently tackles the problem caused by P2P in semi-supervised counting.

2. Related Works

Object counting in dense scenes. The development of deep learning has significantly influenced crowd counting. Prior to this, detecting body parts [16, 17] was the common pipeline for crowd understanding in computer vision, although some methods implemented counting models without explicit object segmentation or tracking information [4–6]. Beyond specific crowd scenarios, class-agnostic counting [14, 27, 30, 45, 49, 71] has also been developed to estimate the distribution of objects in the same category. Most recent counting methods implement neural networks following the density map regression scheme [3, 8, 20, 50, 60, 65, 67, 69, 73]. In addition to continuous innovations in model architecture design [3, 20, 24], new loss functions have also been developed, including Bayesian Loss [38], loss functions based on balanced [63] and unbalanced optimal transport [22, 39, 61], and characteristic function loss computed in the frequency domain [51, 52]. A study analyzes the relationship between current loss functions from the perspective of proximal mapping [28]. Some methods also perform crowd analysis via instance segmentation [1, 9] and point detection [21, 31, 47, 57]. The latter is the scheme adopted in our semi-supervised model.

Semi-supervised counting. Semi-supervised counting has attracted broad attention recently [18, 19, 23, 25, 26, 32–34, 36, 42, 54]. L2R [32, 33] introduces a ranking rule for unlabeled crowd images based on the principle that cropped images should have smaller counts than full images. GP [54] iteratively updates pseudo-labels using Gaussian Processes. IRAST [34] and SUA [42] enhance models by incorporating a segmentation branch. DAC [23] uses learnable density agents to capture varying crowd densities. OT-M [26] transforms density maps into point maps as pseudo-labels, achieving strong performance with a density-to-point loss function [61]. Our method also employs pseudo-labeling but uses a point detector for counting, avoiding the need for density map transformations as in [26]. Additionally, uncertainty is directly measured by the scores of extracted points, without any additional learnable structures like those in [42].

Visual Explanation. Visualizing the importance of extracted image features is a straightforward way to interpret a model. The mainstream of visual explanation focuses on methods producing activation maps [13, 44, 48, 64, 74, 76–78], which are the products of feature maps from some intermediate layer and weight maps indicating the corresponding feature's importance. For example, Grad-CAM [13, 48] defines the weights using the corresponding gradient maps, generated via backpropagation. Beyond the focus on classification tasks, ODAM [74, 76] extends gradient-based CAM to object detection, visualizing the local impact of features on the detector's decisions. In this paper, we propose PSAM to visualize how the activation map changes for each predicted point in an ill-posed semi-supervised counting framework. Based on the observations from PSAM, we design a P2R matching strategy to avoid the over-activation of pixels in local regions, which effectively resolves the drawbacks of P2P and enables the successful training of a semi-supervised point detector.

3. Point Counter in Semi-Supervised Counting

This section formulates a simplified point-based counter by removing the regression branch in P2PNet [57], and then presents the training scheme and challenges when training with labeled and unlabeled data, respectively.

3.1. Preliminary on A Simplified Point Counter

Common point-based crowd counters like P2PNet [57] contain a classification and a regression branch. However, we find a simplification can be achieved by removing the offset regression branch. On one hand, the crowd count can be estimated by counting foreground pixels with scores greater than 0.5, a process that does not involve the regression branch. On the other hand, pixel coordinates alone are sufficient to locate pedestrians, as demonstrated in STEERER [10] and GL [61], which use the coordinates of local maxima to indicate pedestrians locations.

As a result, P2PNet can be simplified to:

$$\mathcal{F}(I, \Theta_f) \rightarrow F \in \mathbb{R}^{c\times h\times w}, \tag{1}$$

$$\mathcal{D}(F, \Theta_d) \rightarrow D \in \mathbb{R}^{h\times w} \tag{2}$$

$$\rightarrow \mathcal{P} = \left\{\boldsymbol{p} \in \mathbb{R}^n, \boldsymbol{x} \in \mathbb{R}^{n\times 2}\right\}, \tag{3}$$

where $\mathcal{F}$ is the image encoder with learnable parameters Θ_f, and $\mathcal{D}$ denotes the density decoder with learnable parameter Θ_d. Notably, the offset regression branch is excluded in this simplified formulation. The pixel set $\boldsymbol{p} \in \mathbb{R}^n$ represents the flattened values in D, and its corresponding coordinate set is denoted as $\boldsymbol{x} \in \mathbb{R}^{n\times 2}$, where $n = h \times w$.

3.2. Training with Labeled Data via P2P Matching

The GT with m points can be denoted as $\boldsymbol{x}' \in \mathbb{R}^{m\times 2}$, where $\boldsymbol{x}'_{[j]} \in \mathbb{R}^2$ represents the location of the j-th pedestrian. Using $\mathcal{P}$ and $\boldsymbol{x}'$, the loss computation involves two parts: P2P matching and binary cross entropy (BCE) calculation.

The former aims to obtain a matrix-based solution $\mathbf{M} \in \{0,1\}^{n\times m}$ that minimize the bipartite-graph matching cost between $\boldsymbol{x}$ and $\boldsymbol{x}'$ within the cost matrix $\mathbf{C} \in \mathbb{R}^{n\times m}$:

$$\mathbf{C}_{[ij]} = \tau\|\boldsymbol{x}_{[i]} - \boldsymbol{x}'_{[j]}\|_2 - \mathcal{S}(\boldsymbol{p}_{[i]}), \tag{4}$$

where $\mathcal{S}(\cdot)$ is defined as the inverse sigmoid function $\mathcal{S}(p) = -\log(1/p - 1)$, rather than the identity operator used in the vanilla P2PNet [57], for improved performance.

To compute BCE loss, the learning objective of $\boldsymbol{p}$ is formulated as $\hat{\boldsymbol{p}} = \mathbf{M}\mathbf{1}_m$, and the BCE loss is calculated for optimization as follows:

$$\mathcal{L}_l = -\lambda\hat{\boldsymbol{p}}^\top \log \boldsymbol{p} - (\mathbf{1}_n - \hat{\boldsymbol{p}})^\top \log(\mathbf{1}_n - \boldsymbol{p}) \tag{5}$$

where λ is a re-weighting factor for matched pixels [57].

3.3. Problem in Training with Unlabeled Image

Here we consider training with unlabeled data following the standard pseudo-labeling procedure [15, 56]. As illustrated in Fig. 1, it requires the cooperation of a student and teacher model that share the same structure described in (1)∼(3). The difference lies in that the teacher model is updated via exponential moving average (EMA) [59], while the student is optimized via gradient descending. During training, the teacher and student process the weakly and strongly data-augmented versions of an unlabeled image, producing $\mathcal{P}_t = \{\boldsymbol{p}_t, \boldsymbol{x}_t\}$ and $\mathcal{P}_s = \{\boldsymbol{p}_s, \boldsymbol{x}_s\}$, respectively. After that, foreground points are extracted from $\mathcal{P}_t$ to construct pseudo-labels:

$$\mathcal{P}'_t = \{\boldsymbol{p}'_t,\ \boldsymbol{x}'_t\} = \left\{\boldsymbol{p}_{t[j]},\ \boldsymbol{x}_{t[j]} \,\middle|\, \boldsymbol{p}_{t[j]} > 0.5\right\}, \tag{6}$$

where $\boldsymbol{x}'_t$ is treated as the pseudo hard label to estimate the matching matrix $\mathbf{M}_{st}$ via Hungarian algorithm, and the loss is obtained by substituting $\hat{\boldsymbol{p}}_t = \mathbf{M}_{st}\mathbf{1}_m$ and $\boldsymbol{p}_s$ into (5).

However, some pseudo-labels may not be reliable. A common solution is to only consider points in $\mathcal{P}'_t$ with high scores [15, 56], *i.e.*, using a confidence vector $\boldsymbol{\zeta}$ to mask out elements whose score is below a threshold η:

$$\boldsymbol{\zeta} = \mathbb{1}\left(\boldsymbol{p}'_t > \eta\right) \xrightarrow{\text{map to } \hat{\boldsymbol{p}}_t \text{ by } \mathbf{M}_{st}} \boldsymbol{z} = \mathbf{M}_{st}\boldsymbol{\zeta}. \tag{7}$$

Note η is greater than 0.5 and (7) only let pseudo points with score greater than η be reliable. Subsequently, $\mathbf{Z} = \mathrm{diag}(\boldsymbol{z})$ is incorporated into the BCE loss for supervision:

$$\mathcal{L}_u = -\lambda\hat{\boldsymbol{p}}_t^\top \mathbf{Z} \log \boldsymbol{p}_s - (\mathbf{1}_n - \hat{\boldsymbol{p}}_t)^\top \mathbf{Z} \log(\mathbf{1}_n - \boldsymbol{p}_s). \tag{8}$$

However, the second item of (8) in P2P is identically zero:

$$\mathcal{L}_{u\,[\texttt{P2P}]} = -\lambda \hat{\boldsymbol{p}}_t^\top \mathbf{Z} \log \boldsymbol{p}_s - 0, \tag{9}$$

regardless of $\boldsymbol{\zeta}$ (embedded in $\mathbf{Z}$). Consequently, no gradients are back-propagated from the losses associated with background pixels, which is ill-posed. Utilizing (9) as the loss function is similar to training a two-class classifier with only samples from the category "1", which will lead the trained model to output "1" regardless of the input.

4. Visual Explanation with PSAM

In addition to the theoretical analysis, we implement a framework that supervises a model using labeled and unlabeled data via (5) and (9), respectively, to empirically investigate the behavior of P2P-based semi-supervised counting. Subsequently, PSAM is designed to visualize how the prediction is influenced by (9) during training. Thus, we can design an effective confidence scheme accordingly to counteract the effects revealed by PSAM, thereby enabling semi-supervised learning to function as expected.

4.1. PSAM: Point-Specific Activation Map

An activation map is a heat map with the same spatial resolution as the image features, where local regions with relatively high values are interpreted as being discriminative (informative) for the current output [48, 74, 78]. Most techniques obtain activation maps by linearly aggregating features with their corresponding gradients, as [53, 58] have shown that the partial derivative with respect to a specific element can reflect its contribution to the final prediction.

Unlike [48, 78], which compute gradients at the image level, crowd counting requires a point-specific activation map (PSAM) to visually explain the regions of interest for every element in $\boldsymbol{p}$ of (3). Thus, the Jacobian matrix of $\boldsymbol{p}$ with respect to image feature F is required:

$$\mathbf{J}_{\mathcal{D}} = \tfrac{\partial \boldsymbol{p}}{\partial F} = \left[\nabla \boldsymbol{p}_{[1]}, \nabla \boldsymbol{p}_{[2]}, \cdots, \nabla \boldsymbol{p}_{[n]}\right]^\top. \tag{10}$$

Afterwards, the q-th heat map $H_{[q]}$ in PSAM is derived by filtering out negative influences on the aggregated feature:

$$H_{[q]} = \max\left(\sum\nolimits_{k=1}^{c} \nabla \boldsymbol{p}_{[q]} \odot F, 0\right) \in \mathbb{R}^{h \times w}, \tag{11}$$

where the summation operator is applied along the channel dimension, and $\odot$ denotes the Hadamard product.

In a naïve implementation, the computation of $\mathbf{J}_{\mathcal{D}} \in \mathbb{R}^{n \times hwc}$ requires a large memory and involves n back-propagations through the computational graph, posing a challenge for visual explanation in pixel-level tasks. However, this issue can be resolved by computing gradients only between $\boldsymbol{p}_{[q]}$ and features within its receptive field. Gradients in other areas are zero, as there is no path connecting them in the computational graph. Thus, an efficient approach can be implemented, as illustrated in Fig. 2.

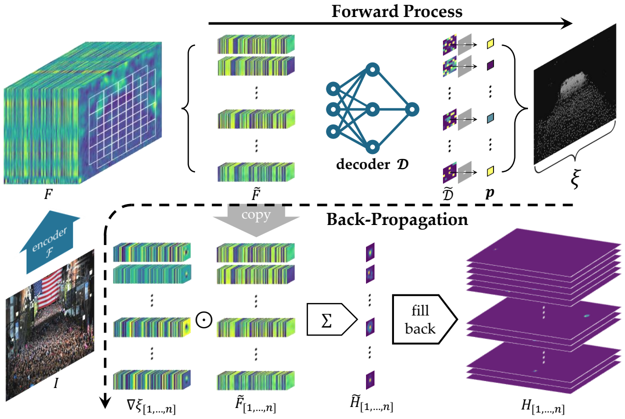


Figure 2. The generation process of PSAM.

In this implementation, the forward process begins with extracting sliding local blocks $\tilde{F}$ according to the receptive field r of $\mathcal{D}$, and then decoding $\tilde{F}$ through (2) in parallel. The final prediction $\boldsymbol{p}$ is generated by stacking pixels located at the center of $r \times r$ patches denoted as in $\tilde{D}$:

$$F \in \mathbb{R}^{h \times w \times c} \rightarrow \tilde{F} \in \mathbb{R}^{(hw) \times (r \times r \times c)}, \tag{12}$$

$$\mathcal{D}(\tilde{F}, \Theta_d) \rightarrow \tilde{D} \in \mathbb{R}^{n \times (r \times r)} \xrightarrow{\text{center of patches}} \boldsymbol{p} \in \mathbb{R}^{n}. \tag{13}$$

The remaining $r^2 - 1$ pixels in each patch are discarded.

Denoting the sum of the output as $\xi = \mathbf{1}_n^\top \boldsymbol{p}$, the gradient $\nabla \xi = \partial \xi / \partial \tilde{F}$ can be obtained via just one back-propagation, and the q-th gradient block, $\nabla \xi_{[q]}$, is equal to the gradient of $\boldsymbol{p}_{[q]}$ with respect to $\tilde{F}_{[q]}$:

$$\nabla \xi_{[q]} = \frac{\partial \xi}{\partial \tilde{F}_{[q]}} = \frac{\partial \xi}{\partial \boldsymbol{p}_{[q]}} \cdot \frac{\partial \boldsymbol{p}_{[q]}}{\partial \tilde{F}_{[q]}} = \frac{\partial \boldsymbol{p}_{[q]}}{\partial \tilde{F}_{[q]}}, \tag{14}$$

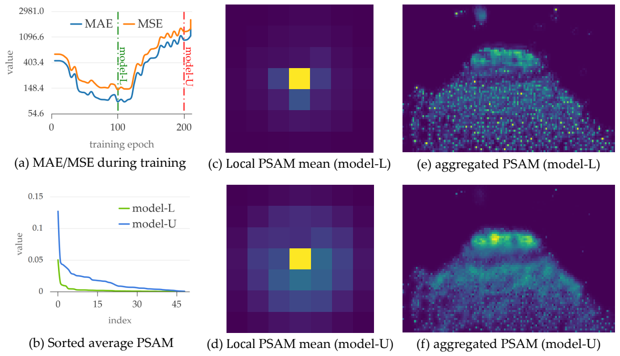
since $\partial \xi / \partial \boldsymbol{p}_{[q]} = 1$. After that, we can parallel compute the PSAM (denoted as $\tilde{H}_{[q]}$) within the receptive field of $\boldsymbol{p}_{[q]}$, by substituting (14) into the formulation presented in (11):

$$\tilde{H}_{[q]} = \max\left(\sum\nolimits_{k=1}^{c} \nabla \xi_{[q]} \odot \tilde{F}_{[q]}, 0\right). \tag{15}$$

As illustrated in the bottom of Fig. 2, the complete PSAM $H_{[1,\dots,n]}$ in (11) can be obtained by filling $\tilde{H}_{[1,\dots,n]}$ back into the corresponding regions of a fully zero matrix:

$$H_{[q,\boldsymbol{t}]} = \begin{cases} \tilde{H}_{[q,\boldsymbol{t}']}, & \text{if } \boldsymbol{t} \in \Omega_q \\ 0, & \text{otherwise} \end{cases}, \tag{16}$$

where Ω_q represents the index set of $\boldsymbol{p}_{[q]}$'s receptive field, and $\boldsymbol{t}' \in \mathbb{R}^2$ is the transformed coordinate of $\boldsymbol{t} \in \mathbb{R}^2$ in H, ensuring $F_{[\boldsymbol{t}]} \equiv \tilde{F}_{[q,\boldsymbol{t}']}$. In this way, the matrix memory required to obtain all $H_{[q]}$ is reduced from $\mathbb{R}^{n \times hwc}$ to $\mathbb{R}^{n \times r^2 c}$ ($r^2 \ll hw$), and only one back-propagation is needed to obtain PSAMs for all pixels.



(a) MAE/MSE during training (c) Local PSAM mean (model-L) (e) aggregated PSAM (model-L)

(b) Sorted average PSAM (d) Local PSAM mean (model-U) (f) aggregated PSAM (model-U)

Figure 3. Observations in PSAM. (a) The training process, where model-L and model-U are extracted from the 100th and 200th epochs, respectively. (b) Comparing sorted values of PSAMs in the two models. (c) & (d) Visualizing the average of local PSAM, and (e) & (f) the aggregated PSAM to compare model-L and model-U from a global perspective.

4.2. Observation in PSAM

We next apply PSAM to visualize the model during semi-supervised counting. The semi-supervised loss function combines (5) and (8):

$$\mathcal{L} = \alpha \mathcal{L}_u + (1 - \alpha)\mathcal{L}_l, \tag{17}$$

where α is a hyperparameter to balance the training of labeled and unlabeled data. Following [15], we set $\alpha = 0$ during the first 100 epochs to generate valid pseudo labels. After that, α is gradually increased to 1. In Fig. 3(a), we plot how the MAE/MSE changes on the validation set during training. The training failed as evinced by both metrics increasing significantly when unlabeled data were involved in training (after epoch 100). We denote 'model-L' as the model at the 100-th epoch that is trained only on labeled data, and 'model-U' as the model at the 200-th epoch that was trained with both labeled and unlabeled data.

We next compare explanations of model-L and model-U using PSAM. We visualize the concerned regions of the crowd by extracting the PSAMs corresponding to the foreground pixels ($\boldsymbol{p}_{[q]} > 0.5$). In Fig. 3(c & d) the element-wise mean of the PSAM patch ($\sum_q \tilde{H}_{[q]} \cdot (\boldsymbol{p}_{[q]} > 0.5)$) is displayed to illustrate how the area of the concerned regions changes between model-L and model-U, while Fig. 3(b) presents a detailed value comparison using a line chart. On one hand, Fig. 3(b) shows that the values in model-U's PSAM are larger than those of model-L; on the other hand, Fig. 3(c & d) shows that model-L's PSAM is more concentrated when compared to model-U. These observations reveal how (8) in P2P scheme affects the trained model: $\mathcal{L}_u$ *leads to over-activation in the PSAM of foreground pixels, causing the decoder $\mathcal{D}$ to falsely identify surrounded pixels as foreground pixels as well.*

Next, we visualize the global PSAM of both models by compressing $H_{[q]}$ corresponding to foreground pixels: $\bar{H} = \sum_{q=1}^{n} (\boldsymbol{p}_{[q]} > 0.5) \cdot H_{[q]}$. As displayed in Fig. 3(e & f), the aggregated PSAM of model-L is sparser than that of model-U, which is consistent with the observation for individual points presented in Fig. 3(c & d). Another observation is that the background regions do not change significantly, indicating that almost all false positives occur in the neighborhood of the initial foreground pixels. This suggests that $\mathcal{L}_u$ *does not alter the hyperplane separating crowd and non-crowd features in the high-dimensional space.*

5. P2R Matching

The observations described in last section reveal that $\mathcal{L}_u$ over-activates the values and enlarges the area of the PSAM corresponding to foreground pixels, which should be suppressed by the supervision of background part in (5), but is actually zero in (8) ($\mathcal{L}_{u\,\texttt{[P2P]}}$ in (9)). This over-activation also confirms two important functions of the background item in (5): it not only guides the model to classify the crowd and non-crowd regions in the image, but also *suppresses the surrounding person region from being activated*. The latter is important since the neighboring pixels of a detected foreground pixel have similar features to the foreground pixel due to the CNN architecture, and thus may become activated if suppression is not applied. This interpretation makes more sense because each person in a crowd image is represented by tens to thousands of pixels, not just one. Supervising the region around the positive pixel with labels of 0 is crucial to a point-based counter. Therefore, during training, it would be more effective to adopt a point-to-region (P2R) matching strategy instead of a point-to-point (P2P) approach. Moreover, as suggested by the above interpretation, P2R is not limited in training with unlabeled data, but can also work in supervised learning with point supervision.

5.1. Training with Labeled Data via P2R

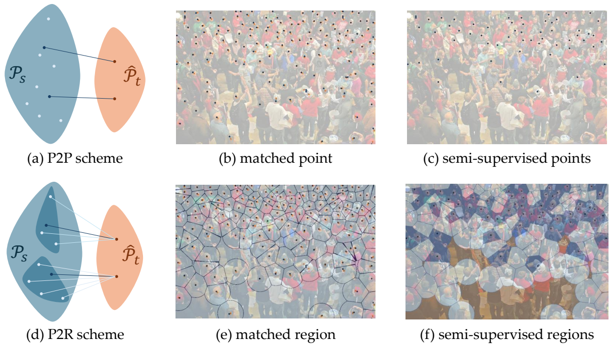
Recalling the notation of the prediction $\mathcal{P} = \{\boldsymbol{p}, \boldsymbol{x}\}$ in (3) and ground truth $\boldsymbol{x}'$, the matching matrix $\mathbf{M}$ in our P2R matching is the Hadamard product of two items:

$$\mathbf{M} = \mathbf{M}_f \odot (\boldsymbol{\beta}^\top \mathbf{1}_m) \quad \in \mathbf{M}^{n \times m}. \tag{18}$$

The first item $\mathbf{M}_f \in \{0,1\}^{n \times m}$ is a many-to-one matrix that assigns each pixel in $\mathcal{P}$ to its nearest point in $\boldsymbol{x}'$:

$$\mathbf{M}_{f\,[ij]} = \begin{cases} 1, & \text{if } \; l_2(i,j) < l_2(i,k) \;\; \forall k \neq j \\ 0, & \text{otherwise} \end{cases}, \tag{19}$$

where $l_2(i,j) = \|\boldsymbol{x}_{[i]} - \boldsymbol{x}'_{[j]}\|_2$ is the l_2-distance between the i-th pixel's and the j-th point's coordinates. $\boldsymbol{\beta} \in \{0,1\}^n$ in the second item of (18) is a vector marking foreground



(a) P2P scheme (b) matched point (c) semi-supervised points

(d) P2R scheme (e) matched region (f) semi-supervised regions

Figure 4. Difference between P2P and P2R matching. (a) and (d) demonstrate an overall difference. P2P focuses only on foreground pixels in $\mathbf{P}_s$, while P2R considers background pixels as well. (b) and (e) show how the matching is performed in P2P and P2R, respectively. P2R segments out local regions for each pseudo-label, whereas P2P only detects one point. (c) and (f) illustrate how untrusted predictions are filtered. P2P retains only foreground pixels for loss computation, while P2R also keeps pixels in the neighborhoods of their corresponding pseudo points.

regions, by setting those pixels in $\mathcal{P}$ far from any points in $\boldsymbol{x}'$ as zero:

$$\boldsymbol{\beta}_{[i]} = \begin{cases} 1, & \text{if } \min_j l_2(i,j) < \mu \\ 0, & \text{otherwise} \end{cases}, \tag{20}$$

in which μ is a hyper parameter indicating the radius if the neighborhood of each GT point is considered as a circle. As displayed in Fig. 4(e), $\mathbf{M}$ segments out a coarse local region that is consistent with the location of the GT point. After that, the learning objective $\hat{\boldsymbol{p}}$ is derived by defining the cost matrix $\mathbf{C} \in \mathbb{R}^{n \times m}$:

$$\mathbf{C}_{[ij]} = \begin{cases} \tau l_2(i,j) - \mathcal{S}(\boldsymbol{p}_{[i]}), & \text{if } \mathbf{M}_{[ij]} = 1 \\ \infty, & \text{otherwise} \end{cases}. \tag{21}$$

Then, the pixel with the minimum cost in each column of $\mathbf{C}$ is marked as the potential foreground pixel:

$$\hat{\mathbf{M}}_{[ij]} = \begin{cases} 1, & \text{if } \mathbf{C}_{[ij]} < \mathbf{C}_{[kj]} \ \forall k \neq i \\ 0, & \text{otherwise} \end{cases}. \tag{22}$$

Finally, the learning objective $\hat{\boldsymbol{p}}$ is derived as $\hat{\boldsymbol{p}} = \hat{\mathbf{M}}\mathbf{1}$, and the loss $\mathcal{L}_l$ is computed between $\boldsymbol{p}$ and $\hat{\boldsymbol{p}}$ according to BCE loss formulated as (5) during training with labeled data.

An extra benefit is that computing $\mathbf{M}$ and $\hat{\mathbf{M}}$ in P2R requires only finding the minimum of each row or column in the matrices, which is significantly faster than executing the Hungarian algorithm — a time-consuming component of the P2P matching strategy [57].

5.2. Training with Unlabeled Data via P2R

Recalling the pseudo-label $\mathcal{P}_t' = \{\boldsymbol{p}_t', \boldsymbol{x}_t'\}$ from the teacher and the supervised student output as $\mathcal{P}_s = \{\boldsymbol{p}_s, \boldsymbol{x}_s\}$, the matching matrix $\mathbf{M}_{st}$ between $\mathcal{P}_s$ and $\mathcal{P}_t'$ is computed via (18) in P2R, and then combined with the confidence in (7) to mask unreliable pseudo labels. Additionally, pixels far from any pseudo points can be directly marked as reliable background pixels, as suggested by observations from the aggregated PSAM in Fig. 3(e & f). Thus, the confidence diagonal matrix $\mathbf{Z}$ is formulated as follows with $\boldsymbol{\zeta} = \mathbb{1}(\hat{\boldsymbol{p}}_t' > \eta)$:

$$\mathbf{Z} = \operatorname{diag}\left[\mathbf{M}_{st}\boldsymbol{\zeta} + (\mathbf{1}_n - \boldsymbol{\beta})\right] \tag{23}$$

The final loss supervising unlabeled data $\mathcal{P}_s$ with pseudo labels is derived by substituting $\mathbf{Z}$ in (23) into (8).

As displayed in Fig. 4(f), the confidence for each pseudo point is propagated to it corresponding region. On one hand, reliable foreground parts (blue) contain a pixel whose pseudo-label is 1, while the pseudo-labels of its surrounding pixels are marked as 0. On the other hand, all pixels in the reliable background parts (khaki) are assigned pseudo-labels of 0 to train the counter with unlabeled data. The uncolored regions are ignored because their confidence is 0.

The pseudo code and flow chart of the proposed P2R loss can be found in the supplemental material.

6. Experiments

This section explores how the proposed P2R training approach enhances a point-based counting model using both labeled and unlabeled data across three parts: semi-supervised crowd counting, unsupervised domain adaptation (UDA) for crowd counting, and ablation studies.

In the semi-supervised experiments, our method is implemented on four crowd counting datasets: ShTech A/B [73], UCF-QNRF [12], and JHU++ [55]. Following DAC [67] and OT-M [26], three protocols are applied: 5%, 10%, and 40% of labeled data, with the remaining crowd images involved in training without annotations. In the UDA part, labeled data from one domain, *e.g.*, ShTech A, is used to initialize a counting model, while unlabeled data from another domain is utilized to capture the domain characteristics of the application scenes. In the ablation study, we test different hyperparameters to demonstrate their impact on the performance of P2R.

6.1. Semi-Supervised Counting

Tab. 1 presents the experimental results of semi-supervised counting. Our method achieves the best performance in nearly all protocols across all crowd datasets. Besides, using 5% data with P2R surpasses the fully-supervised learning with 10% data and is even equivalent to other semi-supervised learning methods using 10% data. Similar performances are also observed between the semi-supervised training with 10% and 40% labeled data. These results demonstrate that P2R is label-efficient and advantageous in semi-supervised learning. In Fig. 5, we visualize predictions of DAC [67], OTM [26], and our P2R. The 3rd row is

Label Pct.	Methods	ShTech A [73]		ShTech B [73]		UCF-QNRF [12]		JHU++ [55]	
		MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE
5%	label-only	93.7	155.2	13.1	24.0	132.7	231.7	104.1	383.5
	MT [59]	104.7	156.9	19.3	33.2	172.4	284.9	101.5	363.5
	L2R [32]	103.0	155.4	20.3	27.6	160.1	272.3	101.4	338.8
	GP [54]	102.0	172.0	15.7	27.9	160.0	275.0	98.9	355.7
	DAC [23]	85.4	134.5	12.6	22.8	120.2	209.3	82.2	294.9
	OT-M [26]	83.7	133.3	12.6	21.5	118.4	195.4	82.7	304.5
	P2R (ours)	**69.9**	**119.5**	**9.1**	**16.6**	**100.1**	**182.5**	**77.8**	**293.5**
10%	label-only	84.0	138.3	10.7	19.2	112.4	186.8	78.7	305.5
	MT [59]	319.3	94.5	15.6	24.5	156.1	245.5	250.3	90.2
	L2R [32]	90.3	153.5	15.6	24.4	148.9	249.8	87.5	315.3
	IRAST [34]	86.9	148.9	14.7	22.9	135.6	233.4	86.7	303.4
	DAC [23]	74.9	115.5	11.1	19.1	109.0	187.2	75.9	282.3
	OT-M [26]	80.1	118.5	10.8	18.2	113.1	186.7	73.0	280.6
	CU [18]	70.7	116.6	9.7	17.7	104.0	1644.2	74.8	281.6
	P2R (ours)	**65.2**	**114.6**	**8.4**	**14.5**	**94.9**	**167.2**	**68.7**	**272.3**
40%	label-only	64.5	105.6	8.1	14.0	99.2	174.7	68.8	283.5
	MT [59]	88.2	151.1	15.9	25.7	147.2	249.6	121.5	388.9
	L2R [32]	86.5	148.2	16.8	25.1	145.1	256.1	123.6	376.1
	SUA [42]	68.5	121.9	14.1	20.6	130.3	226.3	80.7	290.8
	DAC [23]	67.5	110.7	9.6	14.6	91.1	153.4	65.1	**260.0**
	OT-M [26]	70.7	114.5	8.1	13.1	100.6	167.6	72.1	272.0
	P2R (ours)	**55.6**	**95.0**	**6.8**	**11.0**	**86.0**	**144.3**	**63.3**	271.1
100%	P2PNet [57]	52.74	85.06	6.25	9.90	85.32	154.50	61.25	258.65
	P2R (ours)	**51.02**	**79.68**	**6.17**	**9.84**	**83.26**	**138.11**	**58.83**	**253.10**

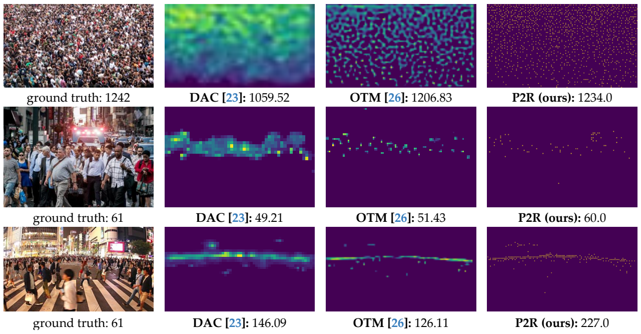
Table 1. Comparison with other recent methods on four benchmark datasets under different labeled protocols.

a failure case of P2R due to crowd motion blur. However, the overall performance of P2R is best.

The last row of Tab. 1 also compares P2P and P2R under a fully-supervised learning scheme (100% labels); P2R performs better than P2PNet on all crowd counting datasets. Additionally, as described in Section 5.1, P2R is faster than P2P due to the absence of the Hungarian algorithm. We compared the efficiency using an image with a resolution of 576 × 960 and 775 annotated points — P2P requires an average of 0.4307 seconds for loss computation, while P2R only needs 0.0064 seconds, which is nearly 68 times faster than P2P. These comparisons demonstrate that P2R surpasses P2P in both effectiveness and efficiency.

6.2. Unsupervised Domain Adaptation (UDA)

UDA aims to transfer knowledge learned from a source domain to a target domain by designing modules or learning objectives that capture domain-agnostic features [11, 29, 43, 65, 70, 72]. Unlike semi-supervised learning, where labeled and unlabeled data originate from the same domain, UDA typically involves a domain gap between the labeled (source domain) and unlabeled crowd data (target domain). This gap often manifests as differences in crowd distribution, density levels, and perspectives. Our P2R can also be applied to UDA. When an unlabeled image from the target domain is provided, the threshold automatically partitions its corresponding pseudo points into two categories: higher-score ($> \eta$) and lower-score ($< \eta$). The higher-score class is considered to be closer to the source domain, while the lower-score class is considered to be farther from the source domain. Thus, the higher-score pseudo points can participate in the training to gradually guide the model to learn to count in the target domain.


ground truth: 1242 | **DAC [23]:** 1059.52 | **OTM [26]:** 1206.83 | **P2R (ours):** 1234.0

ground truth: 61 | **DAC [23]:** 49.21 | **OTM [26]:** 51.43 | **P2R (ours):** 60.0

ground truth: 61 | **DAC [23]:** 146.09 | **OTM [26]:** 126.11 | **P2R (ours):** 227.0

Figure 5. qualitative comparison with other models.

	METHODS	A → B		A → Q		B → A		B → Q	
		MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE
DG	DCCUS [40]	12.6	24.6	119.4	216.6	121.8	203.1	179.1	316.2
	MPCount [43]	11.4	19.7	115.7	199.8	99.6	182.9	165.6	290.4
UDA	BL [38] (w/o DA)	15.9	25.8	166.7	287.6	138.1	228.1	226.4	411.0
	RBT [35]	13.4	29.3	175.0	294.8	112.2	218.2	211.3	381.9
	C^2MoT [70]	12.4	21.1	125.7	218.3	120.7	192.0	198.9	368.0
	FGFD [79]	12.7	23.3	124.1	242.0	123.5	210.7	209.7	384.7
	FSIM [80]	11.1	19.3	105.3	**191.1**	120.3	202.6	194.9	324.5
UDA	P2R (w/o DA)	25.6	35.7	155.1	286.8	130.2	229.1	173.3	329.8
	P2R (w/ DA)	**10.6**	**18.7**	**105.3**	194.8	**89.3**	**176.0**	**139.5**	**243.1**

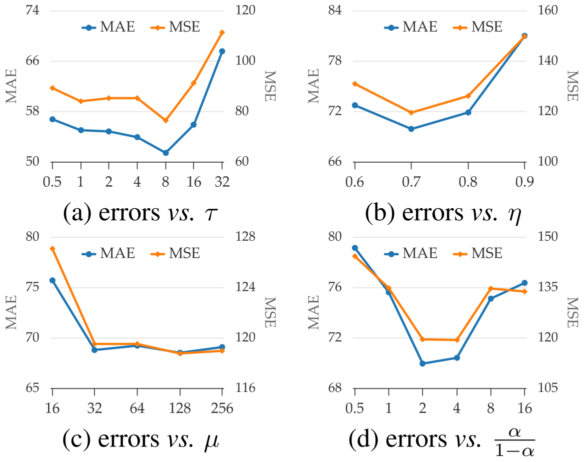
Table 2. Unsupervised domain adaptation for crowd counting. The LHS and RHS of "→" represent adaptation direction. Q, A, and B indicate UCF-QNRF [12], ShTech A, and ShTech B [73]. *DG* indicates domain generalization. The training of UDA requires samples from the target domain, whereas DG does not.

We conduct experiments on four protocols following [70]. The results are presented in Tab. 2. The last two rows demonstrate the results of the proposed P2R. The poor performance obtained without domain adaptation highlights the domain gap between the source data and target data. However, when the unlabeled data from the target domain is involved in training, the estimation errors are significantly reduced, and the results surpass previous methods with complex network structures for domain adaptation. These advanced experimental results demonstrate that the threshold works as expected to capture samples close to the source domain and gradually teaches the model to count in the target domain.

6.3. Ablation Study

We next conduct ablation studies to investigate the influence of hyperparameters in P2R, including τ in (21), the threshold η, μ in (20), and α in (17). The results are in Fig. 6

The impact of τ in (21). τ is a hyperparameter to balance the distance and foreground score when selecting the foreground pixel from the matched region. When $\tau \to \infty$, the predicted score is ignored, which is equivalent to computing cross-entropy between the prediction and the ground truth point map. If $\tau = 0$, the pixel with the highest score is chosen as the foreground in $\hat{\boldsymbol{p}}$. However, Fig. 6(a) shows that neither extreme is optimal. The best performance is achieved when $\tau = 8$.



(a) errors *vs.* τ

(b) errors *vs.* η

(c) errors *vs.* μ

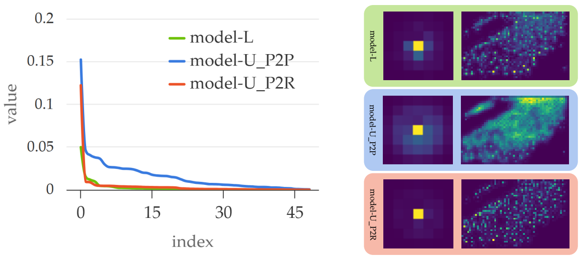
(d) errors *vs.* $\frac{\alpha}{1-\alpha}$

Figure 6. The estimation errors vs. various hyperparameters. (a) is conducted on the fully-supervised ShTech A since it is irrelevant to semi-supervised counting, while (b)-(d) are conducted with 5% labeled data in ShTech A.

The effect of threshold η. η defines the threshold to distinguish reliable and unreliable regions. Since all pseudo points are selected by picking pixels whose score is greater than 0.5 in the prediction, the threshold should also be greater than 0.5. Moreover, a lower η introduces unreliable pseudo-labels into training, while a higher η cannot fully utilize the unlabeled data as most parts cannot participate in training. As shown in Fig. 6(b), the estimation errors exhibit a trend of first decreasing and then increasing within the range of 0.6 to 0.9, with the lowest errors occurring when the threshold is set to 0.7.

The radius of each point μ. The vector $\boldsymbol{\beta}$ in (20) derived from μ marks the foreground points' neighborhood and background regions. It not only restricts the positions where foreground pixels appear by substituting $\boldsymbol{\beta}$ into (18), but also directly identifies reliable background parts via the second item in (23) based on the observation of PSAM in Section 4.2. Fig. 6(c) presents how μ affects the counting performance. It shows the MAE remains around 68 when μ increases from 32 to 256, indicating that a larger μ does not significantly affect the results. This is consistent with the observation in PSAM since pixels far from the foreground points are less likely to be activated during training.

The weight of unlabeled loss. Fig. 6(d) presents how α in (17) affects the performance. A large α focuses more on unlabeled data, leading to training failure, while a small α may cause the model to overfit on labeled data. In the experiments with ShTech A and 5% labeled data, $\frac{\alpha}{1-\alpha} = 2$, *i.e.*, $\alpha = 2/3$, results in the best performance. Additionally, as long as α is not 0, the point-based counter can benefit from unlabeled data, as the maximum MAE/MSE in Fig. 6(d) are smaller than 80/150, which is much better than the model trained with only 5% labeled data (MAE: 93.7, MSE: 155.2).



(a) sorted average PSAM

(b) Local & aggregated PSAM

Figure 7. The comparison of PSAMs among different training schemes. Model-L (green) is trained with only labeled data; model-U_P2P (blue) is trained with both labeled data and unlabeled data using the P2P scheme, while model-U_P2R (orange) is trained with the proposed P2R scheme.

PSAM comparison. Fig. 7 visualizes the PSAMs of models trained with the P2P and P2R schemes. Comparing model-L (green) and model-U_P2R (blue), the latter effectively restricts the PSAM to the neighborhood of the concerned foreground pixel, and its aggregated PSAM is also sparser than that of model-L, demonstrating the effectiveness of P2R. Comparing model-U_P2P and model-U_P2R in Fig. 7(a), the values of the PSAM are similar at the non-peak indexes, but the latter has a higher activation at the peak position compared to the former, indicating that semi-supervised learning further enhances the identification ability of the point-based counter.

7. Conclusion

This paper presents a P2R scheme to train a point-based counter for semi-supervised crowd counting. At the beginning, we establish a pseudo-labeling framework based on P2P matching, but its formulation is ill-posed since the confidence is only applicable to foreground pixels and cannot be propagated to backgrounds. To observe how the foreground and background feature are learned, we propose PSAM to visually interpret the concerned region and activation value of all foreground pixels. From the visualized activation map, it is observed that the neighborhood of each foreground pixel is over-activated and falsely recognized as instances by the decoder. Based on this observation, we replace P2P with P2R by segmenting the prediction into multiple regions containing the corresponding instances. Thus, pixels in the neighborhood of the concerned pseudo points can share the weights with the corresponding pixels. Additionally, the time-consuming Hungarian algorithm is no longer necessary in P2R. Several outstanding experimental results in semi-supervised counting and unsupervised domain adaptation demonstrate the advantages of our P2R matching strategy.

Acknowledgements. This work was supported by a grant from City University of Hong Kong (Project No. 7030010).

References

[1] Shahira Abousamra, Minh Hoai, Dimitris Samaras, and Chao Chen. Localization in the crowd with topological constraints. In *Proceedings of the AAAI Conference on Artificial Intelligence*, pages 872–881, 2021. 2

[2] Areej Alhothali, Amal Balabid, Reem Alharthi, Bander Alzahrani, Reem Alotaibi, and Ahmed Barnawi. Anomalous event detection and localization in dense crowd scenes. *Multimedia Tools and Applications*, 82(10):15673–15694, 2023. 1

[3] Deepak Babu Sam, Shiv Surya, and R Venkatesh Babu. Switching convolutional neural network for crowd counting. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 5744–5752, 2017. 2

[4] Antoni B Chan and Nuno Vasconcelos. Bayesian poisson regression for crowd counting. In *2009 IEEE 12th international conference on computer vision*, pages 545–551. IEEE, 2009. 1, 2

[5] Antoni B Chan and Nuno Vasconcelos. Counting people with low-level features and bayesian regression. *IEEE Transactions on image processing*, 21(4):2160–2177, 2011. 1

[6] Antoni B Chan, Zhang-Sheng John Liang, and Nuno Vasconcelos. Privacy preserving crowd monitoring: Counting people without people models or tracking. In *2008 IEEE conference on computer vision and pattern recognition*, pages 1–7. IEEE, 2008. 2

[7] Mulin Chen, Qi Wang, and Xuelong Li. Anchor-based group detection in crowd scenes. In *2017 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 1378–1382. IEEE, 2017. 1

[8] Zhi-Qi Cheng, Qi Dai, Hong Li, Jingkuan Song, Xiao Wu, and Alexander G Hauptmann. Rethinking spatial invariance of convolutional networks for object counting. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 19638–19648, 2022. 2

[9] Junyu Gao, Tao Han, Qi Wang, Yuan Yuan, and Xuelong Li. Learning independent instance maps for crowd localization. *arXiv preprint arXiv:2012.04164*, 2020. 2

[10] Tao Han, Lei Bai, Lingbo Liu, and Wanli Ouyang. Steerer: Resolving scale variations for counting and localization via selective inheritance learning. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 21848–21859, 2023. 1, 3

[11] Yuhang He, Zhiheng Ma, Xing Wei, Xiaopeng Hong, Wei Ke, and Yihong Gong. Error-aware density isomorphism reconstruction for unsupervised cross-domain crowd counting. In *Proceedings of the AAAI conference on artificial intelligence*, pages 1540–1548, 2021. 7

[12] 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 *Proceedings of the European conference on computer vision (ECCV)*, pages 532–546, 2018. 6, 7

[13] Brian Kenji Iwana, Ryohei Kuroki, and Seiichi Uchida. Explaining convolutional neural networks using softmax gradient layer-wise relevance propagation. In *2019 IEEE/CVF International Conference on Computer Vision Workshop (ICCVW)*, pages 4176–4185. IEEE, 2019. 3

[14] Ruixiang Jiang, Lingbo Liu, and Changwen Chen. Clipcount: Towards text-guided zero-shot object counting. In *Proceedings of the 31st ACM International Conference on Multimedia*, pages 4535–4545, 2023. 2

[15] Dong-Hyun Lee et al. Pseudo-label: The simple and efficient semi-supervised learning method for deep neural networks. In *Workshop on challenges in representation learning, ICML*, page 896. Atlanta, 2013. 2, 3, 5

[16] Bastian Leibe, Edgar Seemann, and Bernt Schiele. Pedestrian detection in crowded scenes. In *2005 IEEE computer society conference on computer vision and pattern recognition (CVPR'05)*, pages 878–885. IEEE, 2005. 2

[17] Victor Lempitsky and Andrew Zisserman. Learning to count objects in images. *Advances in neural information processing systems*, 23, 2010. 2

[18] Chen Li, Xiaoling Hu, Shahira Abousamra, and Chao Chen. Calibrating uncertainty for semi-supervised crowd counting. In *2023 IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 16685–16695. IEEE, 2023. 2, 7

[19] Hanxiao Li, Yonghong Song, and Tong Geng. Semi-supervised crowd counting based on hard pseudo-labels. In *2024 International Joint Conference on Neural Networks (IJCNN)*, pages 1–8. IEEE, 2024. 2

[20] Yuhong Li, Xiaofan Zhang, and Deming Chen. Csrnet: Dilated convolutional neural networks for understanding the highly congested scenes. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1091–1100, 2018. 2

[21] Dingkang Liang, Wei Xu, and Xiang Bai. An end-to-end transformer model for crowd localization. In *ECCV*, pages 38–54. Springer, 2022. 1, 2

[22] Hui Lin, Xiaopeng Hong, Zhiheng Ma, Xing Wei, Yunfeng Qiu, Yaowei Wang, and Yihong Gong. Direct measure matching for crowd counting. In *Proceedings of the Thirtieth International Joint Conference on Artificial Intelligence, pages = 837–844, publisher = ijcai.org, year = 2021,* . 2

[23] Hui Lin, Zhiheng Ma, Xiaopeng Hong, Yaowei Wang, and Zhou Su. Semi-supervised crowd counting via density agency. In *ACM Multimedia*, 2022. 2, 7

[24] Hui Lin, Zhiheng Ma, Rongrong Ji, Yaowei Wang, and Xiaopeng Hong. Boosting crowd counting via multifaceted attention. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 19628–19637, 2022. 1, 2

[25] Hui Lin, Zhiheng Ma, Rongrong Ji, Yaowei Wang, Zhou Su, Xiaopeng Hong, and Deyu Meng. Semi-supervised counting via pixel-by-pixel density distribution modelling. *arXiv preprint arXiv:2402.15297*, 2024. 2

[26] Wei Lin and Antoni B Chan. Optimal transport minimization: Crowd localization on density maps for semi-supervised counting. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 21663–21673, 2023. 1, 2, 3, 6, 7

[27] Wei Lin and Antoni B Chan. A fixed-point approach to unified prompt-based counting. In *Proceedings of the*

AAAI Conference on Artificial Intelligence, pages 3468–3476, 2024. 2

[28] Wei Lin, Jia Wan, and Antoni B Chan. Proximal mapping loss: Understanding loss functions in crowd counting & localization. In *The Thirteenth International Conference on Learning Representations*, . 2

[29] Wei Lin, Junyu Gao, Qi Wang, and Xuelong Li. Learning to detect anomaly events in crowd scenes from synthetic data. *Neurocomputing*, 436:248–259, 2021. 1, 7

[30] Wei Lin, Kunlin Yang, Xinzhu Ma, Junyu Gao, Lingbo Liu, Shinan Liu, Jun Hou, Shuai Yi, and Antoni B Chan. Scale-prior deformable convolution for exemplar-guided class-agnostic counting. In *BMVC*, page 313, 2022. 2

[31] Chengxin Liu, Hao Lu, Zhiguo Cao, and Tongliang Liu. Point-query quadtree for crowd counting, localization, and more. In *CVPR*, pages 1676–1685, 2023. 1, 2

[32] Xialei Liu, Joost Van De Weijer, and Andrew D Bagdanov. Leveraging unlabeled data for crowd counting by learning to rank. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 7661–7669, 2018. 2, 7

[33] Xialei Liu, Joost Van De Weijer, and Andrew D Bagdanov. Exploiting unlabeled data in cnns by self-supervised learning to rank. *IEEE transactions on pattern analysis and machine intelligence*, 41(8):1862–1878, 2019. 2

[34] Yan Liu, Lingqiao Liu, Peng Wang, Pingping Zhang, and Yinjie Lei. Semi-supervised crowd counting via self-training on surrogate tasks. In *European Conference on Computer Vision*, pages 242–259. Springer, 2020. 2, 7

[35] Yuting Liu, Zheng Wang, Miaojing Shi, Shin'ichi Satoh, Qijun Zhao, and Hongyu Yang. Towards unsupervised crowd counting via regression-detection bi-knowledge transfer. In *Proceedings of the 28th ACM International Conference on Multimedia*, pages 129–137, 2020. 7

[36] Yanbo Liu, Yingxiang Hu, Guo Cao, and Yanfeng Shang. Semi-supervised crowd counting via multi-task pseudo-label self-correction strategy. *IEEE Transactions on Circuits and Systems for Video Technology*, 2024. 2

[37] Zheng Ma and Antoni B Chan. Counting people crossing a line using integer programming and local features. *IEEE Transactions on Circuits and Systems for Video Technology*, 26(10):1955–1969, 2015. 1

[38] Zhiheng Ma, Xing Wei, Xiaopeng Hong, and Yihong Gong. Bayesian loss for crowd count estimation with point supervision. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 6142–6151, 2019. 1, 2, 7, 13

[39] Zhiheng Ma, Xing Wei, Xiaopeng Hong, Hui Lin, Yunfeng Qiu, and Yihong Gong. Learning to count via unbalanced optimal transport. In *Proceedings of the AAAI Conference on Artificial Intelligence*, pages 2319–2327, 2021. 2

[40] Lucas Mansilla, Rodrigo Echeveste, Diego H Milone, and Enzo Ferrante. Domain generalization via gradient surgery. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 6630–6638, 2021. 7

[41] Riccardo Mazzon, Fabio Poiesi, and Andrea Cavallaro. Detection and tracking of groups in crowd. In *2013 10th IEEE International conference on advanced video and signal based surveillance*, pages 202–207. IEEE, 2013. 1

[42] Yanda Meng, Hongrun Zhang, Yitian Zhao, Xiaoyun Yang, Xuesheng Qian, Xiaowei Huang, and Yalin Zheng. Spatial uncertainty-aware semi-supervised crowd counting. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 15549–15559, 2021. 2, 3, 7

[43] Zhuoxuan Peng and S-H Gary Chan. Single domain generalization for crowd counting. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 28025–28034, 2024. 7

[44] Harish Guruprasad Ramaswamy et al. Ablation-cam: Visual explanations for deep convolutional network via gradient-free localization. In *proceedings of the IEEE/CVF winter conference on applications of computer vision*, pages 983–991, 2020. 3

[45] Viresh Ranjan, Udbhav Sharma, Thu Nguyen, and Minh Hoai. Learning to count everything. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 3394–3403, 2021. 2

[46] Weihong Ren, Xinchao Wang, Jiandong Tian, Yandong Tang, and Antoni B Chan. Tracking-by-counting: Using network flows on crowd density maps for tracking multiple targets. *IEEE Transactions on Image Processing*, 30:1439–1452, 2020. 1

[47] Jihye Ryu and Kwangho Song. Crowd counting and individual localization using pseudo square label. *IEEE Access*, 2024. 2

[48] Ramprasaath R Selvaraju, Michael Cogswell, Abhishek Das, Ramakrishna Vedantam, Devi Parikh, and Dhruv Batra. Grad-cam: Visual explanations from deep networks via gradient-based localization. In *Proceedings of the IEEE international conference on computer vision*, pages 618–626, 2017. 2, 3, 4

[49] Zenglin Shi, Ying Sun, and Mengmi Zhang. Training-free object counting with prompts. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 323–331, 2024. 2

[50] Weibo Shu, Jia Wan, Kay Chen Tan, Sam Kwong, and Antoni B Chan. Crowd counting in the frequency domain. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 19618–19627, 2022. 2

[51] Weibo Shu, Jia Wan, Kay Chen Tan, Sam Kwong, and Antoni B Chan. Crowd counting in the frequency domain. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 19618–19627, 2022. 2

[52] Weibo Shu, Jia Wan, and Antoni B Chan. Generalized characteristic function loss for crowd analysis in the frequency domain. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2023. 2

[53] Karen Simonyan. Deep inside convolutional networks: Visualising image classification models and saliency maps. *arXiv preprint arXiv:1312.6034*, 2013. 4

[54] Vishwanath A Sindagi, Rajeev Yasarla, Deepak Sam Babu, R Venkatesh Babu, and Vishal M Patel. Learning to count in the crowd from limited labeled data. In *European Conference on Computer Vision*, pages 212–229. Springer, 2020. 2, 7

[55] Vishwanath A Sindagi, Rajeev Yasarla, and Vishal M Patel. Jhu-crowd++: Large-scale crowd counting dataset and a

benchmark method. *IEEE transactions on pattern analysis and machine intelligence*, 44(5):2594–2609, 2020. 6, 7

[56] Kihyuk Sohn, David Berthelot, Nicholas Carlini, Zizhao Zhang, Han Zhang, Colin A Raffel, Ekin Dogus Cubuk, Alexey Kurakin, and Chun-Liang Li. Fixmatch: Simplifying semi-supervised learning with consistency and confidence. *Advances in neural information processing systems*, 33:596–608, 2020. 1, 2, 3

[57] Qingyu Song, Changan Wang, Zhengkai Jiang, Yabiao Wang, Ying Tai, Chengjie Wang, Jilin Li, Feiyue Huang, and Yang Wu. Rethinking counting and localization in crowds: A purely point-based framework. In *CVPR*, pages 3365–3374, 2021. 1, 2, 3, 6, 7, 13

[58] Jost Tobias Springenberg, Alexey Dosovitskiy, Thomas Brox, and Martin A. Riedmiller. Striving for simplicity: The all convolutional net. In *3rd International Conference on Learning Representations, ICLR 2015, Workshop Track Proceedings*, 2015. 4

[59] Antti Tarvainen and Harri Valpola. Mean teachers are better role models: Weight-averaged consistency targets improve semi-supervised deep learning results. *Advances in neural information processing systems*, 30, 2017. 1, 3, 7

[60] Jia Wan, Qingzhong Wang, and Antoni B Chan. Kernel-based density map generation for dense object counting. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 44(3):1357–1370, 2020. 2

[61] Jia Wan, Ziquan Liu, and Antoni B Chan. A generalized loss function for crowd counting and localization. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 1974–1983, 2021. 1, 2, 3, 13

[62] Jia Wan, Qiangqiang Wu, Wei Lin, and Antoni Chan. Robust zero-shot crowd counting and localization with adaptive resolution sam. In *European Conference on Computer Vision*, pages 478–495. Springer, 2024. 1

[63] Boyu Wang, Huidong Liu, Dimitris Samaras, and Minh Hoai Nguyen. Distribution matching for crowd counting. *Advances in neural information processing systems*, 33:1595–1607, 2020. 2, 13

[64] Haofan Wang, Zifan Wang, Mengnan Du, Fan Yang, Zijian Zhang, Sirui Ding, Piotr Mardziel, and Xia Hu. Score-cam: Score-weighted visual explanations for convolutional neural networks. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition workshops*, pages 24–25, 2020. 3

[65] Qi Wang, Junyu Gao, Wei Lin, and Yuan Yuan. Learning from synthetic data for crowd counting in the wild. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 8198–8207, 2019. 2, 7

[66] 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*, 43(6):2141–2149, 2020. 1

[67] Qi Wang, Wei Lin, Junyu Gao, and Xuelong Li. Density-aware curriculum learning for crowd counting. *IEEE Transactions on Cybernetics*, 52(6):4675–4687, 2020. 2, 6

[68] Qi Wang, Wei Lin, Junyu Gao, and Xuelong Li. Density-aware curriculum learning for crowd counting. *IEEE Transactions on Cybernetics*, 52(6):4675–4687, 2020. 1

[69] Qi Wang, Junyu Gao, Wei Lin, and Yuan Yuan. Pixel-wise crowd understanding via synthetic data. *International Journal of Computer Vision*, 129(1):225–245, 2021. 2

[70] Qiangqiang Wu, Jia Wan, and Antoni B Chan. Dynamic momentum adaptation for zero-shot cross-domain crowd counting. In *Proceedings of the 29th ACM International Conference on Multimedia*, pages 658–666, 2021. 7

[71] Jingyi Xu, Hieu Le, Vu Nguyen, Viresh Ranjan, and Dimitris Samaras. Zero-shot object counting. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 15548–15557, 2023. 2

[72] Qi Zhang, Wei Lin, and Antoni B Chan. Cross-view cross-scene multi-view crowd counting. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 557–567, 2021. 7

[73] Yingying Zhang, Desen Zhou, Siqin Chen, Shenghua Gao, and Yi Ma. Single-image crowd counting via multi-column convolutional neural network. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 589–597, 2016. 1, 2, 6, 7, 13

[74] Chenyang Zhao and Antoni B. Chan. ODAM: gradient-based instance-specific visual explanations for object detection. In *ICLR*, 2023. 2, 3, 4

[75] Chunhui Zhao, Zhiyuan Zhang, Jinwen Hu, Dong Wang, Bin Fan, Quan Pan, and Qiang He. Crowd anomaly event detection in surveillance video based on the evolution of the spatial position relationship feature. In *2018 IEEE 14th International Conference on Control and Automation (ICCA)*, pages 247–252, 2018. 1

[76] Chenyang Zhao, Janet H Hsiao, and Antoni B Chan. Gradient-based instance-specific visual explanations for object specification and object discrimination. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2024. 2, 3

[77] Chenyang Zhao, Kun Wang, Xingyu Zeng, Rui Zhao, and Antoni B Chan. Gradient-based visual explanation for transformer-based clip. In *International Conference on Machine Learning*, pages 61072–61091. PMLR, 2024.

[78] Bolei Zhou, Aditya Khosla, Agata Lapedriza, Aude Oliva, and Antonio Torralba. Learning deep features for discriminative localization. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2921–2929, 2016. 2, 3, 4

[79] Huilin Zhu, Jingling Yuan, Zhengwei Yang, Xian Zhong, and Zheng Wang. Fine-grained fragment diffusion for cross domain crowd counting. In *Proceedings of the 30th ACM International Conference on Multimedia*, pages 5659–5668, 2022. 7

[80] Huilin Zhu, Jingling Yuan, Xian Zhong, Liang Liao, and Zheng Wang. Find gold in sand: Fine-grained similarity mining for domain-adaptive crowd counting. *IEEE Transactions on Multimedia*, 2023. 7