Optimal Transport Minimization: Crowd Localization on Density Maps for Semi-Supervised Counting

Wei Lin   Antoni B. Chan
Department of Computer Science, City University of Hong Kong
elonlin24@gmail.com, abchan@cityu.edu.hk

Abstract

The accuracy of crowd counting in images has improved greatly in recent years due to the development of deep neural networks for predicting crowd density maps. However, most methods do not further explore the ability to localize people in the density map, with those few works adopting simple methods, like finding the local peaks in the density map. In this paper, we propose the optimal transport minimization (OT-M) algorithm for crowd localization with density maps. The objective of OT-M is to find a target point map that has the minimal Sinkhorn distance with the input density map, and we propose an iterative algorithm to compute the solution. We then apply OT-M to generate hard pseudo-labels (point maps) for semi-supervised counting, rather than the soft pseudo-labels (density maps) used in previous methods. Our hard pseudo-labels provide stronger supervision, and also enable the use of recent density-to-point loss functions for training. We also propose a confidence weighting strategy to give higher weight to the more reliable unlabeled data. Extensive experiments show that our methods achieve outstanding performance on both crowd localization and semi-supervised counting. Code is available at https://github.com/Elin24/OT-M.

1. Introduction

Crowd understanding gains much attention due to its wide applications in surveillance [33, 61] and crowd disaster prevention. Most studies in this area concentrate on crowd counting, whose objective is to provide the total number and distribution of crowds in a scene automatically. Due to the development of deep learning, recent methods [5,58,59,62,67] have achieved success on a variety of counting benchmarks [50, 61, 62, 66]. Counting methods can be extended to other applications, such as traffic management [63], animal protection [2], and health care [32].

Although crowd counting has been greatly developed, most methods do not explore further applications of the estimated density maps after obtaining the count. Specifically, there is limited research on crowd localization, tracking, or group analysis with predicted density maps. Taking crowd localization as an example, only a few methods, such as local maximum [13, 58, 65], integer programming [36], or Gaussian mixture models (GMM) [14], have been proposed to locate pedestrians or tiny objects from density maps. Moreover, recent localization methods ignore the counting density map, and instead are based on point detection [40, 54], blob segmentation [1, 10, 17], or inverse distance maps [24]. However, this will increase the inefficiency of a crowd understanding system, since separate networks are required for counting and localization.

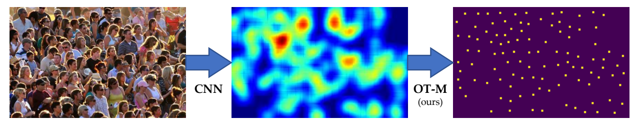


Figure 1. The relationship between crowd counting CNNs and OT-M algorithm. CNNs are trained to generate density maps (soft labels). OT-M produces point maps (hard labels) from the predicted density maps, without needing training.

To broaden the application of density maps for localization, in this paper we propose a parameter-free algorithm, *Optimal Transport Minimization* (OT-M), to estimate the point map indicating locations of objects from a counting density map (see Fig. 1). OT-M minimizes the Sinkhorn distance [7] between the density map (source) and point localization map (target), through an alternating scheme that estimates the optimal transport plan from the current point map (the OT-step), and updates the point map by minimizing their transport cost (the M-step). OT-M is parameter-free and requires no training, and thus can be applied to any crowd-counting method using density maps.

To demonstrate the applicability of density-map based localization, we apply OT-M to semi-supervised counting. In previous work, [38] builds a baseline for semi-supervised counting based on the mean-teacher framework [55], but finds it ineffective. Looking closely, we note that the baseline in [38] uses a *soft* pseudo-label (density map) to supervise the student model, whereas successful semi-supervised classification [52] or segmentation [6, 55] methods usually are based on *hard* pseudo-labels (e.g., class labels or binary segmentation masks). In the context of semi-supervised crowd counting, the hard label is the *point map* in which each person is pseudo-annotated with a point. Thus, in this paper we generate hard pseudo-labels using OT-M for the unlabeled crowd images for semi-supervised crowd counting. As an additional benefit, the hard pseudo-labels allow training CNNs under semi-supervised learning using recent density-to-point loss functions (e.g., Bayesian loss (BL) [34] or generalized loss (GL) [58]), which are more effective than traditional losses, e.g. L2.

Similar to other semi-supervised tasks, some estimated pseudo labels may be inaccurate due to limitations of the current trained model. To reduce the effect of these noisy pseudo-labels, we propose a *confidence-weighted* GL (C-GL) for semi-supervised counting. Specifically, we compute the unbalanced optimal transport plan between the student's predicted density map (source) and the hard pseudo-labels from the teacher (target), and then define the pixel-wise and point-wise confidences based on the consistencies between source, target and the plan. Experiments show that the trained model is more robust with our C-GL.

In summary, the contributions of this paper are 3-fold:

- We propose an OT-M algorithm to estimate the locations of objects from density maps, which is based on minimizing the Sinkhorn distance between the density map and the target point map. Since OT-M is parameter-free, it can be applied to any density map without training.
- We use OT-M to produce *hard pseudo-labels* for semi-supervised counting, which conforms with schemes in other semi-supervised tasks. The hard label also allows applying density-to-point loss like GL to unlabeled data for more effective training.
- To mitigate risks brought by inaccurate pseudo-labels, we propose a confidence-weighted Generalized Loss to reduce the influence of inconsistencies between the teacher's and student's predictions. Experiments show that our loss improves semi-supervised counting performance.

2. Related works

Crowd Counting. Before deep learning became popular, detection-based methods for counting pedestrians were based on detecting human body parts [19, 20], but do not work well in the dense crowds due to partial occlusions. Instead, regression-based methods overcome these obstacles by directly predicting the final count based on low-level features [4, 5, 12]. Recent methods use a convolutional neural network (CNN) to estimate a density map from a crowd image, and the corresponding count is obtained by summing over the density map [21]. Various networks are proposed to address scale variations in crowd scene [3, 11, 22, 67], by using multiple columns [3, 67], multi-task learning [11], or dilated convolution [22]. However, all these methods are trained by pixel-wise L2 loss, which is ineffective since the original ground-truth point map is blurred by a hand-crafted Gaussian kernel and loses localization information. To address this, [25, 34, 35, 58, 60] directly compare the predicted density map and the ground-truth point map. [35] designed an efficient algorithm to optimize counting models based on UOT's semi-dual regularized formulation, while [25] derived a semi-balanced form of Sinkhorn divergence to satisfy the identity of indiscernibles. [58] proves that L2 loss and Bayesian loss [34] are special cases of generalized loss.

Localization with Object Density Maps. The goal of most crowd counting methods is to estimate a density map representing the distribution of pedestrians, and then take its sum as the final count result. However, as shown in Fig. 1, localizing pedestrians according to the density map is tricky since the predicted density map is blurry. [36] recovers the locations of objects by applying integer programming to windowed observations of the density map. [36] also considers clustering methods, like K-means or mean-shift, on each connected component to localize partially-occluded instances. Differently, [14] proposes localization by learning a Gaussian mixture model (GMM) to fit the density map, where the centers of the estimated Gaussian components correspond to the people locations. Finally, [13, 58] aim to estimate sparse density maps, and then define every local maximum pixel whose value is greater than a threshold as the location of a person. In contrast to these methods, our OT-M is based finding the point map with minimal Sinkhorn distance to the density map. Empirical results show that OT-M is more accurate and robust.

Density maps are also used to improve detectors and trackers in crowded scenes [14]. In [45], density estimation is jointly optimized with standard detectors to reduce false positive errors and improve recall. In [43, 44], the tracking-by-counting paradigm is proposed to overcome occlusions and appearance variations during tracking in crowd scene.

Semi-supervised Counting. As labeling very dense crowd images can be expensive, leveraging unlabeled crowd images with semi-supervised counting has seen increased interest in recent years. L2R [29, 30] introduces a rank rule for unlabeled data, inspired by the observation that cropped image contains the same or fewer objects than the original image. GP [49] proposes an iterative method based on the Gaussian process to assign soft pseudo-labels to unlabeled images. IRAST [31] uses density segmentation as a surrogate task to detect conflicting predictions and correct them. The segmentation task is also considered by SUA [38], which follows the popular teacher-student scheme [16], and segmentation results are used to model unlabeled data's uncertainty. DACount [26] designs a structure similar to the multiple columns and switching module in Switch-CNN [3], and uses multiple learnable density agents to learn features of crowd in different density levels.

Unlike these methods based on surrogate tasks, our semi-supervised framework is based on density maps predicted from the unlabeled data. Our OT-M is utilized to generate hard pseudo-labels (point maps) for training with unlabeled data, and we further propose a confidence-weighted GL to reduce the influence of inaccurate pseudo-labels.

3. OT-M Algorithm

In object counting, the ground-truth (GT) density map is obtained by convolving a Gaussian kernel with the GT point map, *i.e.*, converting the hard-label into a soft-label. However, there is little research to address the inverse problem, converting the *soft-label* density map into a *hard-label* point map. In this section, we introduce a parameter-free algorithm to estimate the hard label from a soft density map by minimizing the entropic optimal transport cost (*i.e.*, Sinkhorn distance [7][1]) between them.

Let the soft-label density map predicted by a CNN be represented as $\mathcal{A} = \{(a_i, \boldsymbol{x}_i)\}_{i=1}^n$, where $a_i \geq 0$ and $\boldsymbol{x}_i \in \mathbb{R}^2$ are the density value and coordinate of the i-th pixel, and n is the number of pixels. Given the density map $\mathcal{A}$, our goal is to estimate a hard label $\mathcal{B} = \{(b_j, \boldsymbol{y}_j)\}_{j=1}^m$, where $b_j = 1$ and $\boldsymbol{y}_j \in \mathbb{R}^2$ represents the j-th point (person location). The number of points m is the count obtained from the density map $\mathcal{A}$, rounded to the nearest integer, *i.e.*, $m = \lfloor \sum_{i=1}^n a_i \rceil$. Since $b_j = 1$ is fixed for all points, we will use the shorthand $\mathcal{B} = \{\boldsymbol{y}_j\}_{j=1}^m$ to reduce clutter.

We estimate the hard labels by minimizing the Sinkhorn distance between the points $\mathcal{B}$ and the density map $\mathcal{A}$,

$$\hat{\mathcal{B}} = \mathop{\arg\min}_{\mathcal{B}=\{\boldsymbol{y}_j\}_{j=1}^m} \mathcal{L}^\varepsilon(\mathcal{A}, \mathcal{B}), \tag{1}$$

where $\mathcal{L}^\varepsilon(\mathcal{A}, \mathcal{B})$ is the Sinkhorn distance between $\mathcal{A}$ and $\mathcal{B}$, and ε is a near-zero weight for the entropic term:

$$\mathcal{L}^\varepsilon(\mathcal{A}, \mathcal{B}) = \min_{\mathbf{P} \in \mathbf{U}(\boldsymbol{a}, \boldsymbol{b})} \langle \mathbf{C}, \mathbf{P} \rangle - \varepsilon \mathcal{H}(\mathbf{P}), \tag{2}$$

$$= \sum_{i,j} C_{ij} P_{ij} + \varepsilon \sum_{i,j} P_{ij} \log(P_{ij}) \tag{3}$$

where $\mathbf{C} = [C_{ij}]$ is the cost matrix, $\mathbf{P} = [P_{ij}]$ is the transport plan, and $\mathcal{H}(\mathbf{P}) = -\sum_{i,j} P_{ij} \log(P_{ij})$ is the entropy of $\mathbf{P}$. Here, the cost matrix element $C_{ij} = C(\boldsymbol{x}_i, \boldsymbol{y}_j)$ measures the cost when moving a unit mass from $\boldsymbol{x}_i$ to $\boldsymbol{y}_j$. We use the squared Euclidean distance as the cost function:

$$C(\boldsymbol{x}_i, \boldsymbol{y}_j) = \|\boldsymbol{x}_i - \boldsymbol{y}_j\|^2. \tag{4}$$

[1]Sinkhorn *distance* [7] is different from Sinkhorn *divergence* [8]. The latter removes the entropic bias from the former to build a positive-definite loss function. Here we use the basic one [7] to estimate hard labels.

The Sinkhorn distance in (2) finds the optimal transport plan $\mathbf{P}$, whose element P_{ij} is the mass quantity (*i.e.*, density) transported from $\boldsymbol{x}_i$ to $\boldsymbol{y}_j$, that minimizes the total transport cost. In balanced optimal transport, $\mathbf{P}$ is constrained to admissible couplings that preserves the total mass from each $\boldsymbol{x}_i$ and to each $\boldsymbol{y}_j$, $\mathbf{U}(\boldsymbol{a}, \boldsymbol{b}) \stackrel{\text{def.}}{=} \{\mathbf{P} \in \mathbb{R}_+^{n \times m} : \mathbf{P}\mathbf{1}_m = \boldsymbol{a}, \mathbf{P}^\top \mathbf{1}_n = \boldsymbol{b}\}$, where $\boldsymbol{a} = [a_i]$, $\boldsymbol{b} = [b_j]$, and $\mathbf{1}_n$ is the vector of n ones.

To find the solution of (1), we propose the OT-M algorithm that iteratively computes: 1) the optimal transport plan for Sinkhorn distance in (2) while holding the cost matrix fixed; and 2) the optimal cost matrix, parametrized by the points $\mathcal{B}$, while holding the transport plan fixed. Formally, after initialization of the points $\mathcal{B}^{(0)} = \{\boldsymbol{y}_j^{(0)}\}_{j=1}^m$, the k-th iteration of the OT-M algorithm is:

$$\text{OT-step: } \mathbf{P}^{(k)} = \mathop{\text{argmin}}_{\mathbf{P} \in \mathbf{U}(\boldsymbol{a}, \boldsymbol{b})} \langle \mathbf{C}(\mathcal{B}^{(k-1)}), \mathbf{P} \rangle - \varepsilon \mathcal{H}(\mathbf{P}), \tag{5}$$

$$\text{M-step: } \mathcal{B}^{(k)} = \mathop{\text{argmin}}_{\mathcal{B}=\{\boldsymbol{y}_j\}_{j=1}^m} \langle \mathbf{C}(\mathcal{B}), \mathbf{P}^{(k)} \rangle - \varepsilon \mathcal{H}(\mathbf{P}^{(\mathbf{k})}), \tag{6}$$

where $\mathbf{C}(\mathcal{B})$ is the cost matrix between $\mathcal{A}$ and $\mathcal{B}$. The details of each step and convergence proof are presented next.

3.1. Optimal Transport Step (OT-Step)

The goal of OT-step is to compute the optimal transport plan $\mathbf{P}^{(k)}$ between $\mathcal{A}$ and $\mathcal{B}^{(k-1)}$. The solution can be formulated as [41]:

$$\mathbf{P} = \text{diag}(\mathbf{u}) \mathbf{K} \text{diag}(\mathbf{v}), \quad \mathbf{K} = \exp(-\mathbf{C}/\varepsilon), \tag{7}$$

where $\mathbf{u} \in \mathbb{R}_+^n$ and $\mathbf{v} \in \mathbb{R}_+^m$ are two unknown scaling variables. Then the minimization of (5) can be solved efficiently through the Sinkhorn algorithm – an alternate minimization scheme [46, 51]. Specifically, the following iterations are repeated after initializing $\mathbf{v}$ with an arbitrary positive vector $\mathbf{v}^{(0)}$ ($\mathbf{v}^{(0)} = \mathbf{1}_m$ by default):

$$\mathbf{u}^{(l+1)} = \frac{\boldsymbol{a}}{\mathbf{K}\mathbf{v}^{(l)}}, \quad \mathbf{v}^{(l+1)} = \frac{\boldsymbol{b}}{\mathbf{K}^\top \mathbf{u}^{(l+1)}}, \tag{8}$$

where the division is element-wise.

In summary, the cost matrix $\mathbf{C}^{(k)} = \mathbf{C}(\mathcal{B}^{(k-1)})$ is computed from the current points $\mathcal{B}^{(k-1)}$, and the Gibbs kernel matrix $\mathbf{K}^{(k)} = \exp(-\mathbf{C}^{(k)}/\varepsilon)$ is calculated. Next, the iterations in (8) are run until convergence, and the transport plan $\mathbf{P}^{(k)}$ is calculated from (7). Note that $\boldsymbol{a}$ and $\boldsymbol{b}$ are normalized to make their sums equal to perform balanced OT.

3.2. Minimization step (M-Step)

The M-step computes the new set of points $\mathcal{B}^{(k)} = \{\boldsymbol{y}_j^{(k)}\}$ by minimizing (6) while keeping $\mathbf{P}^{(k)}$ fixed. Specifically, we rewrite (6) by plugging in the cost function,

$$\mathcal{B}^{(k)} = \mathop{\text{argmin}}_{\{\boldsymbol{y}_j\}_{j=1}^m} \sum_{i,j} P_{ij}^{(k)} C(\boldsymbol{x}_i, \boldsymbol{y}_j), \tag{9}$$

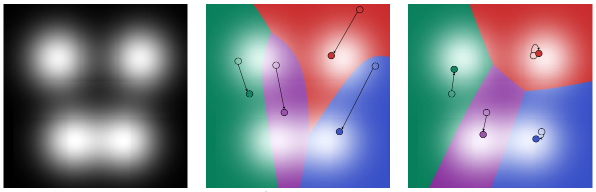


Figure 2. A demonstration of OT-M algorithm. (a) a density map $\mathcal{A}$ with four objects; (b, c) two iterations in OT-M. The color indicates the pixels that are assigned to the same point using the current transport plan $\mathbf{P}^{(k)}$ in the OT-step. Filled-in markers are the new set of points $\mathcal{B}^{(k)}$, while open markers are the previous set of points $\mathcal{B}^{(k-1)}$. Solid arrows represent movements of corresponding points in the M-step.

and noting that each $\boldsymbol{y}_j$ can be optimized independently,

$$\boldsymbol{y}_j^{(k)} = \arg\min_{\boldsymbol{y}_j} \sum_{i=1}^{n} P_{ij}^{(k)} \|\boldsymbol{x}_i - \boldsymbol{y}_j\|^2. \tag{10}$$

Setting the derivative of (10) equal to zero, the solution is:

$$\frac{\partial}{\partial \boldsymbol{y}_j} \sum_{i=1}^{n} P_{ij}^{(k)} \|\boldsymbol{x}_i - \boldsymbol{y}_j\|^2 = 0 \Rightarrow \boldsymbol{y}_j^{(k)} = \frac{\sum_{i=1}^{n} P_{ij}^{(k)} \boldsymbol{x}_i}{\sum_{i=1}^{n} P_{ij}^{(k)}}. \tag{11}$$

In (11), $\boldsymbol{y}_j^{(k)}$ is the *barycenter* of masses assigned to $\boldsymbol{y}_j$ in the transport plan $\mathbf{P}^{(k)}$. The algorithm is summarized in the Supp., and two iterations are visualized in in Fig. 2.

3.3. Convergence of the OT-M Algorithm

We next prove that our OT-M algorithm converges – after each iteration the estimated $\mathcal{B}^{(k)}$ decreases the Sinkhorn distance in (1) until a local minimum is achieved, at which point it cannot decrease (but will not increase) [37, 39]. Denote the cost matrix in the k-th iteration as $\mathbf{C}^{(k)} = \mathbf{C}(\mathcal{B}^{(k-1)})$. After computing the optimal transport plan $\mathbf{P}^{(k)}$ for cost matrix $\mathbf{C}^{(k)}$ in the OT-step in (5), we have

$$\langle \mathbf{C}^{(k)}, \mathbf{P}^{(k)} \rangle - \varepsilon \mathcal{H}(\mathbf{P}^{(k)}) \leq \langle \mathbf{C}^{(k)}, \mathbf{P}^{(k-1)} \rangle - \varepsilon \mathcal{H}(\mathbf{P}^{(k-1)}), \tag{12}$$

since $\mathbf{P}^{(k)}$ is the minimizer over all admissible transport plans. Next, in the M-step in (6), we obtain the optimal $\mathcal{B}^{(k)}$ for fixed transport plan $\mathbf{P}^{(k)}$, and thus

$$\langle \mathbf{C}(\mathcal{B}^{(k)}), \mathbf{P}^{(k)} \rangle \leq \langle \mathbf{C}^{(k)}, \mathbf{P}^{(k)} \rangle, \tag{13}$$

since the cost matrix $\mathbf{C}(\mathcal{B}^{(k)})$ is the minimizer. Noting that $\mathbf{C}(\mathcal{B}^{(k)}) = \mathbf{C}^{(k+1)}$, we thus obtain

$$\langle \mathbf{C}^{(k+1)}, \mathbf{P}^{(k)} \rangle \leq \langle \mathbf{C}^{(k)}, \mathbf{P}^{(k)} \rangle, \tag{14}$$

$$\Rightarrow \langle \mathbf{C}^{(k)}, \mathbf{P}^{(k-1)} \rangle \leq \langle \mathbf{C}^{(k-1)}, \mathbf{P}^{(k-1)} \rangle. \tag{15}$$

Finally, substituting (15) into (12), we obtain the convergence condition:

$$\langle \mathbf{C}^{(k)}, \mathbf{P}^{(k)} \rangle - \varepsilon \mathcal{H}(\mathbf{P}^{(k)}) \leq \langle \mathbf{C}^{(k-1)}, \mathbf{P}^{(k-1)} \rangle - \varepsilon \mathcal{H}(\mathbf{P}^{(k-1)}), \tag{16}$$

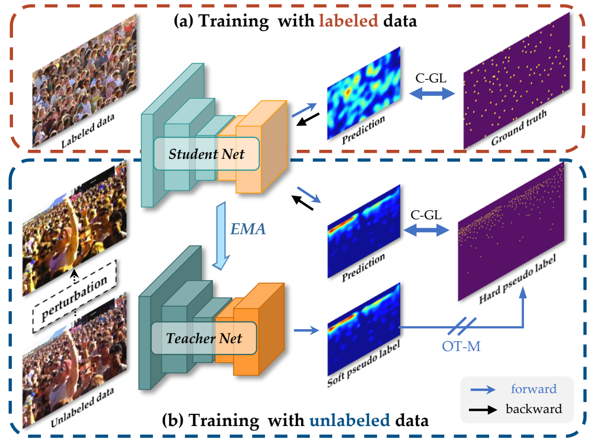


Figure 3. The pipeline of our semi-supervised counting framework. The teacher is updated with the EMA of the student. On the unlabeled data, the teacher predicts a soft pseudo label (density map), from which OT-M generates a hard pseudo label (point map). The student is trained on the labeled data with GT point maps, and the unlabeled data with pseudo point maps. C-GL is our proposed confidence-weighted generalized loss.

where the LHS is the Sinkhorn objective for iteration k and the RHS is for $k-1$. Thus, in each iteration the objective in (1) is non-increasing, and the algorithm converges.

4. OT-M Based Semi-Supervised Counting

Using pseudo-labels [18] is an effective method for semi-supervised learning. Most related works on classification [18,53,64] and segmentation [6,9,68] empirically show that hard labels are more valuable than soft labels. In this section we show how to effectively take advantage of hard pseudo-labels, *i.e.*, point maps, generated through OT-M algorithm for semi-supervised counting. As shown in Fig. 3, we use the mean-teacher framework [55], where an exponential moving average (EMA) is used to update the parameters in the teacher net. For labeled images, the student net is trained with fully-supervised learning on the GT point maps. For unlabeled images, we use the teacher net to generate a soft pseudo-label (density map), and OT-M is applied to produce a hard pseudo-label (point map). Meanwhile, these unlabeled images are perturbed and input into the student net to generate a prediction, which is supervised by the hard pseudo labels. For effective training, we propose a *confidence-weighted generalized loss* (C-GL) to reduce the effect of inconsistent (noisy) pseudo-labels.

4.1. Generalized Loss with Gating

The Generalized Loss (GL) [58] is based on the unbalanced optimal transport (UOT) problem,

$$L_{gl}^{\varepsilon,\tau} = \min_{\mathbf{P} \in \mathbb{R}_{+}^{n \times m}} \langle \mathbf{C}, \mathbf{P} \rangle - \varepsilon \mathcal{H}(\mathbf{P}) + \tau D(\mathbf{P}, \mathbf{a}, \mathbf{b}), \tag{17}$$

where $\mathbf{a} \in \mathbb{R}^n$ and $\mathbf{b} = \mathbf{1}_m$ are the predicted density values and ground truth point values. D is a divergence penalizing marginal deviation, with τ controls the degree of imbalance allowed. In GL, $D(\mathbf{P}, \mathbf{a}, \mathbf{b})$ is defined as

$$D_{gl}(\mathbf{P}, \mathbf{a}, \mathbf{b}) = \|\mathbf{P}\mathbf{1}_m - \mathbf{a}\|_2^2 + \|\mathbf{P}^\top \mathbf{1}_n - \mathbf{b}\|_1. \quad (18)$$

There is no efficient algorithm to directly implement (17) with D_{gl} as the divergence, so [58] firstly approximates the optimal $\mathbf{P}$ by solving UOT with KL divergence (KL-UOT), and then plugs $\widehat{\mathbf{P}}$ back into (17) to compute the GL. Specifically, applying the Sinkhorn algorithm [41] to KL-UOT yields both the optimal transport plan $\widehat{\mathbf{P}}$ and $(\mathbf{f}^*, \mathbf{g}^*)$, the gradients of $(\mathbf{a}, \mathbf{b})$. Thus the generalized loss is rewritten:

$$\begin{aligned} L_{gl}^{\varepsilon,\tau} &= \mathbf{a}^\top \mathbf{f}^* + \mathbf{b}^\top \mathbf{g}^* - \varepsilon \mathcal{H}(\widehat{\mathbf{P}}) \\ &\quad + \tau_2 \|\widehat{\mathbf{P}}\mathbf{1}_m - \mathbf{a}\|_2^2 + \tau_1 \|\widehat{\mathbf{P}}^\top \mathbf{1}_n - \mathbf{b}\|_1. \end{aligned} \quad (19)$$

Note that $\widehat{\mathbf{P}}$ is a function of both $\mathbf{a}$ and $\mathbf{b}$.

In (19) we introduce separate hyperparameters (τ_1, τ_2) on the L1/L2 loss terms to "gate" them to improve training. Let $m_{\widehat{\mathbf{P}}}$, $m_{\mathbf{a}}$, and $m_{\mathbf{b}}$ be the sum of $\widehat{\mathbf{P}}$, $\mathbf{a}$, and $\mathbf{b}$ respectively, which correspond to the total transported density, the count of the predicted density map, and the GT count. In practice we find that the Sinkhorn algorithm sometimes estimates a $\widehat{\mathbf{P}}$ whose sum $m_{\widehat{\mathbf{P}}}$ is larger than both $m_{\mathbf{a}}$ and $m_{\mathbf{b}}$, which is harmful to training. For example, suppose $m_{\mathbf{a}} < m_{\mathbf{b}}$, then the predicted count is smaller than the GT count, and we hope to increase $m_{\mathbf{a}}$ to match $m_{\mathbf{b}}$. However if $m_{\mathbf{b}} < m_{\widehat{\mathbf{P}}}$, then the L1 loss term will encourage $m_{\widehat{\mathbf{P}}}$ to decrease, which also decreases $m_{\mathbf{a}}$, but this is in conflict to the goal of increasing $m_{\mathbf{a}}$. Thus, we can set $\tau_1 = 0$ to ignore the L1 loss term when $m_{\mathbf{a}} < m_{\mathbf{b}} < m_{\widehat{\mathbf{P}}}$. Other cases can be handled analogously, resulting in the following "gating" of the L1/L2 loss terms through setting of (τ_1, τ_2),

$$\tau_1 = \begin{cases} 0, & m_{\mathbf{a}} < m_{\mathbf{b}} < m_{\widehat{\mathbf{P}}}, \\ 0, & m_{\widehat{\mathbf{P}}} < m_{\mathbf{b}} < m_{\mathbf{a}}, \\ \tau, & \text{otherwise}. \end{cases} \quad \tau_2 = \begin{cases} 0, & m_{\mathbf{b}} < m_{\mathbf{a}} < m_{\widehat{\mathbf{P}}}, \\ 0, & m_{\widehat{\mathbf{P}}} < m_{\mathbf{a}} < m_{\mathbf{b}}, \\ \tau, & \text{otherwise}. \end{cases}$$

We set hyperparameter $\tau = 0.1$ following [58].

4.2. Confidence Strategy

Semi-supervised learning has a common drawback: predictions for unlabeled data are usually noisy, which leads to *confirmation bias* [15] towards these errors and consequently learns defective models. To overcome this issue, we build a confidence strategy based on the consistency between the teacher's hard label and the student's prediction for semi-supervised counting model trained with GL.

Assume the density map predicted by the student model is $\mathbf{a}$, and the hard pseudo-label predicted by the teacher model (via OT-M) is $\mathbf{b}$, and the KL-UOT transport plan between them is $\widehat{\mathbf{P}}$. We calculate the consistency via the point-wise distance between the transport plan and point target $\mathbf{b}$:

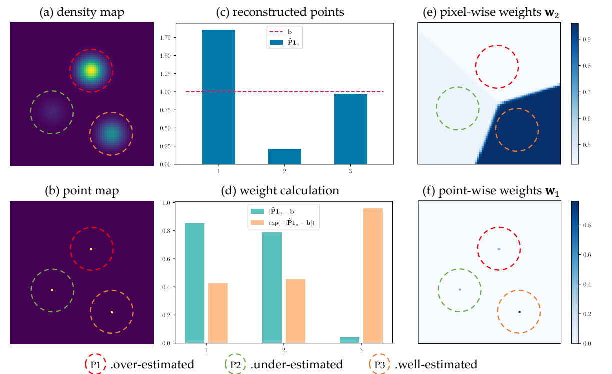


Figure 4. The visualization of confidence weights. There are three points: P1 is over-estimated, P2 is under-estimated, and P3 is well-esimated. (a) predicted density map. (b) point map. (c) target points $\mathbf{b}$ and its reconstruction $\widehat{\mathbf{P}}\mathbf{1}_n$ using KL-UOT. (d) The absolute difference between $\mathbf{b}$ and $\widehat{\mathbf{P}}\mathbf{1}_n$, and $\mathbf{w}_1$ calculated by (20). (e, f) visualization of pixel-wise $\mathbf{w}_2$ and point-wise $\mathbf{w}_1$ weights on the original map. Our confidence weights assign large weight to point P3 and its surrounding pixels, and small weights to P1 and P2 and their surrounding pixels.

$$\mathbf{w}_1 = \exp\left[-\gamma\big(\mathrm{diag}(\mathbf{b})^{-1}|\widehat{\mathbf{P}}^\top \mathbf{1}_n - \mathbf{b}|\big)\right], \quad (20)$$

in which $\gamma > 0$ is a hyperparameter to decrease the confidence, and $|\cdot|$ is the element-wise absolute value. Note that $\mathbf{w}_1$ is close to 1 as long as the sum of each column of $\widehat{\mathbf{P}}$'s is close to corresponding element $b_j = 1$.

Next, we propagate $\mathbf{w}_1$ to the pixels to compute confidence values for elements in $\mathbf{a}$. Specifically, pixel-wise confidence $\mathbf{w}_2$ is a weighted sum of elements in $\mathbf{w}_1$, and the weight is computed by normalizing each row in $\widehat{\mathbf{P}}$:

$$\mathbf{w}_2 = \mathrm{diag}(\widehat{\mathbf{P}}\mathbf{1}_m)^{-1}\widehat{\mathbf{P}}\mathbf{w}_1. \quad (21)$$

Embedding $\mathbf{w}_1$ and $\mathbf{w}_2$ into GL, the final formulation is:

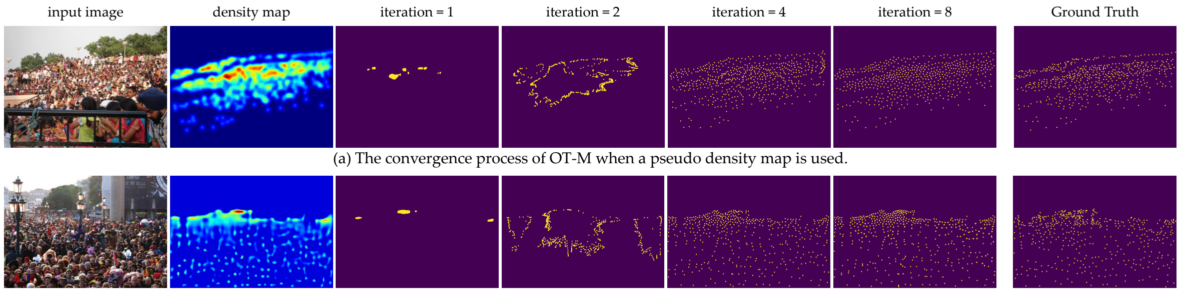
$$\begin{aligned} L_{c\text{-}gl}^{\varepsilon,\tau,\gamma} &= \mathbf{a}^\top \mathbf{W}_2 \mathbf{f}^* + \mathbf{b}^\top \mathbf{W}_1 \mathbf{g}^* - \varepsilon \mathcal{H}(\widehat{\mathbf{P}}) \\ &\quad + \tau_2 \|\mathbf{W}_2(\widehat{\mathbf{P}}\mathbf{1}_m - \mathbf{a})\|_2^2 + \tau_1 \|\mathbf{W}_1(\widehat{\mathbf{P}}^\top \mathbf{1}_n - \mathbf{b})\|_1, \end{aligned} \quad (22)$$

where $\mathbf{W}_1 = \mathrm{diag}(\mathbf{w}_1)$ and $\mathbf{W}_2 = \mathrm{diag}(\mathbf{w}_2)$. Fig. 4 visualizes $\mathbf{w}_1$ and $\mathbf{w}_2$ in a simple example.

Note that the original GL in (19) is a special case of the C-GL in (22). Some counting methods [26, 27, 57] report that there may be annotation noise in labeled data, so (22) can also be applied to labeled data to depress noise if a suitable γ is given. In our experiments, we set $\gamma = 0.5$ for both labeled and unlabeled data.

5. Experiments

In this section, we conduct experiments to demonstrate the efficacy of our OT-M algorithm and its use in semi-supervised counting. In the first part, we use synthetic and real data to empirically show the OT-M algorithm's



(a) The convergence process of OT-M when a pseudo density map is used.

(b) The convergence process of OT-M when the density map is predicted by CNN in GL[53].

Figure 5. Estimated point maps in different iterations of OT-M algorithm (Top-k initialization is used here for better visualization).

convergence process. The localization performance is also compared with previous density-map based algorithms. In the second part, we compare our framework with previous semi-supervised counting approaches. After that, the ablation study is conducted to show whether each component of our framework works as expected.

5.1. Experiment setup

Localization experiments are conducted on two public datasets, UCF-QNRF [13] and NWPU-Crowd [61]. Semi-supervised counting is tested on four datasets: ShanghaiTech-A and B (ST-A, ST-B) [67], UCF-QNRF [13], and JHU++ [50]. In each dataset, 5%, 10%, 40% of training samples are selected as labeled data. We follow 2 protocols, the single trial version from [26], and a new version based on averaging over 5 random trials for each percentage.

For the OT-M algorithm, we consider three initialization methods of $\mathcal{B}^{(0)}$: 1) *Top-k* selects the m pixels with the largest density values as the initial points; 2) *Uniform* selects the initial m points uniformly at random; 3) *adaptive* initialization normalizes the density map into a probability distribution, from which the m initial points are sampled. We set the maximum number of iterations in OT-M as 16, and also use the following early stopping criteria:

$$\frac{1}{m}\sum_{j=1}^{m}\|\boldsymbol{y}_j^{(k)}-\boldsymbol{y}_j^{(k-1)}\|_2 < 1 \text{ and } \max_j \|\boldsymbol{y}_j^{(k)}-\boldsymbol{y}_j^{(k-1)}\|_2 < \frac{1}{r},$$

where r is the down-sampling ratio of CNN ($r = \frac{1}{8}$ in our experiments). When the average distance moved of points in $\mathcal{B}^{(k)}$ is smaller than 1 pixel, and the maximum moved distance is smaller than $1/r$, then the algorithm stops.

5.2. Experiments on OT-M Convergence

We first show the effectiveness and convergence of OT-M on some examples from ST-A. The accuracy of OT-M relies on the precision of estimated density maps. When the density map is perfect, the estimated point map should be extremely similar to the GT. To show this, we generate synthetic density maps by applying a Gaussian kernel (with variance 8) to the GT point map, and then recover the point maps through OT-M. In Fig. 6(a), we present the Sinkhorn

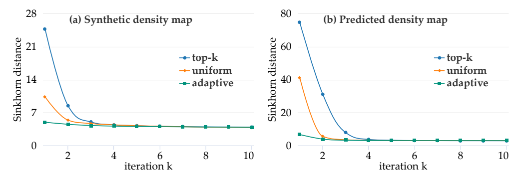


Figure 6. The convergence of OT-M during the iterative process on (a) synthetic density maps; (b) predicted density maps by CNN.

distance objective between the predicted point map and the given density map during the iterations, when using different initializations. Using the adaptive initialization yields faster convergence, compared to other initialization methods. In Fig. 5(a), we visualize the iterative convergence process on an example. To better show the effectiveness of OT-M, we use the worst initialization method, top-k, where most of the initial points are in a small area. After initialization, OT-M is able to gradually move the points closer to the targets, and finally yields a point map that is close to the ground-truth.

We next evaluate how well the recovered point maps compare to the GT point maps. We use the GT density maps from UCF-QNRF, downsample them by 1/8 to make the problem more difficult, and then apply OT-M to recover the point maps. The results are presented in the first row of Table 1. OT-M obtains high precision, recall, and F1 (> 0.91), showing its superior accuracy compared to Local Maximum (LM) [58] and GMM [14]. Furthermore, we find that the OT-M algorithm is robust to initialization – the three different initialization methods yield the same results, although they have different convergence speed.

Finally, we visualize a example on density maps predicted from a CNN in Fig. 5(b). Compared with synthetic density maps, CNN predictions are more ambiguous and noisy. The OT-M algorithm works as expected: decreasing the Sinkhorn distance between the estimated point map and the given density map, as displayed in Fig. 6(b). We further evaluate the localization performance of OT-M on predicted density maps in the next section.

5.3. Localization Performance on Density Maps

In this section, we compare OT-M algorithm with two other localization methods based on density maps: Lo-

Density Map	Localization	Precision	Recall	F-measure
ground-truth density map	LM [58]	0.892	0.736	0.807
	GMM [14]	0.842	0.838	0.840
	OT-M (ours)	**0.914**	**0.910**	**0.912**
GL [58] cvpr'21	LM [58]	0.782	0.748	0.765
	GMM [14]	0.750	0.728	0.739
	OT-M (ours)	**0.804**	**0.783**	**0.793**
MAN [27] cvpr'22	LM [58]	0.624	0.483	0.544
	GMM [14]	0.749	0.732	0.736
	OT-M (ours)	**0.772**	**0.755**	**0.760**
ChfL [47] cvpr'22	LM [58]	0.812	0.571	0.671
	GMM [14]	0.755	0.740	0.747
	OT-M (ours)	**0.780**	**0.765**	**0.772**

Table 1. Comparison of different localization methods on UCF-QNRF dataset for different density maps (ground-truth and predicted). Note that the ground-truth density map is downsampled by 1/8 to match the output size of GL and ChfL.

Method			Prec.	Rec.	F-meas.
box	Faster RCNN [42]	cvpr'15	0.958	0.035	0.068
density map	RAZNet [28]	cvpr'19	0.666	0.543	0.599
	GL+LM [58]	cvpr'21	**0.800**	0.562	0.660
	GL+OT-M(ours)		0.710	**0.658**	**0.683**
point	P2PNet [54]	iccv'21	0.729	0.695	0.712
	CLTR [23]	eccv'22	0.694	0.676	0.685

Table 2. Localization from density maps on NWPU-Crowd.

cal Maximum (LM) [58] and GMM [14]. The evaluation is based on precision, recall, and F-measure, following [13, 61]. Tab. 1 presents results on UCF-QNRF using different density-map-based localization and recent density map counting models, including GL [58], MAN [27], and ChfL [47]. Note that GL uses a simple VGG-19 [48] as backbone, while MAN uses a transformer-based [56] structure. Thus, their down-sampling rates are $\frac{1}{8}$ and $\frac{1}{16}$ respectively, and the predicted density map from GL have higher resolution than MAN, which is why GL's localization ability is better than MAN's. Overall, OT-M is the better localization algorithm with substantially better precision, recall, and F-measure, regardless of the density map model used. In contrast LM and GMM are sensitive to the type of density maps; LM performs better than GMM on GL density maps because they have higher resolution, and thus are more sparse, while in contrast GMM performs better than LM on MAN density maps because they are smoother. Finally, the result on GT density maps are the upper-bound performance of localization with 1/8-downsampled density maps, showing there is still room for improvement.

We also test the localization performance on the NWPU-Crowd test set and compare with other density-map based methods, as presented in Tab. 2. Faster RCNN [42] is based on box detection, and it has the best precision (0.958), but its recall is very small (0.035). Among the density map methods, our OT-M gives the highest recall (0.658) and F-measure (0.683). It outperforms the baseline Local Maximum (LM) in terms of recall and F-measure. Besides, OT-M also performs similarly to the recent point-based method CLTR [23] in terms of F-measure.

Label Pct.	Methods	ST-A MAE	ST-A MSE	ST-B MAE	ST-B MSE	UCF-QNRF MAE	UCF-QNRF MSE	JHU++ MAE	JHU++ MSE
5%	MT [55]	104.7	156.9	19.3	33.2	172.4	284.9	101.5	363.5
	L2R [29]	103.0	155.4	20.3	27.6	160.1	272.3	101.4	338.8
	GP [49]	102.0	172.0	15.7	27.9	160.0	275.0	98.9	355.7
	DAC [26]	85.2	135.0	**12.5**	22.1	123.5	207.3	83.9	308.8
	OT-M (ours)	**83.7**	**133.3**	12.6	**21.5**	**118.4**	**195.4**	**82.7**	**304.5**
10%	MT [55]	94.5	115.5	15.6	24.5	145.5	250.3	90.2	319.3
	L2R [29]	90.3	115.5	15.6	24.4	148.9	249.8	87.5	315.3
	IRAST [31]	86.9	148.9	14.7	22.9	135.6	233.4	86.7	303.4
	DAC [26]	82.5	123.2	10.9	19.1	115.1	193.5	74.0	297.1
	OT-M (ours)	**80.1**	**118.5**	**10.8**	**18.2**	**113.1**	**186.7**	**73.0**	**280.6**
40%	MT [55]	88.2	151.1	15.9	25.7	147.2	249.6	121.5	388.9
	L2R [29]	86.5	148.2	16.8	25.1	145.1	256.1	123.6	376.1
	SUA [38]	**68.5**	121.9	14.1	20.6	130.3	226.3	80.7	290.8
	DAC [26]	71.1	119.7	8.1	13.6	**96.8**	168.2	**66.3**	276.6
	OT-M (ours)	70.7	**114.5**	**8.1**	**13.1**	100.6	**167.6**	72.1	**272.0**

Table 3. Comparison of semi-supervised counting on the single trial experiment from [26].

5.4. Semi-Supervised Counting

We next present the results for semi-supervised counting. Tab. 3 compares ours with previous methods using the same backbone and experiment protocol (*i.e.*, same set of labeled data) from [26], consisting of one trial for each label percentage (5%, 10%, 40%). The DAC results are reproduced to ensure the same experiment design. It shows that OT-M outperforms most semi-supervised counting approaches, especially when there are fewer labeled data (5% and 10%). When the label percentage is increased to 40%, DAC [26] achieves lower MAE and MSE on UCF-QNRF and JHU++, while SUA [38]'s MAE is the lowest on ST-A. However, our framework has the smallest MSE on all these datasets.

In the above experiments, only one trial is used for each label percentage, which is inadequate because the random selection of labeled data strongly influences the counting performance and stability. To investigate this issue, we test DAC [26] and our method in another experiment with multiple trials, where each trial uses different randomly selected labeled data. Here the averaged MAE/MSE over multiple trials is more representative of the algorithm's performance, compared to using a single trial, especially when the number of labeled samples is small. The experiment results are presented in Tab. 4. For 5% and 10% labeled data, our framework outperforms DAC [26] on all four datasets. The average and standard deviation of MAE/MSE are much smaller than DAC. For 40% labeled data, DAC [26] has lower MAEs than ours on three datasets, while our model has lower MSE on all datasets. In summary, the combination of OT-M and the proposed confidence strategy can achieve outstanding performance using a simple mean-teacher framework, especially for smaller percentages of labeled data (5% & 10%).

5.5. Ablation Study on Semi-Supervised Counting

We next conduct ablation studies using the 5% labeled data setting on UCF-QNRF [13].

Confidence-weighted GL on labeled data. The top half of Tab. 5 presents the effect of the (τ_1, τ_2) gating scheme

Label Percentage	Methods	ST-A MAE	ST-A MSE	ST-B MAE	ST-B MSE	UCF-QNRF MAE	UCF-QNRF MSE	JHU++ MAE	JHU++ MSE
5%	DAC [26]	92.9±3.4	148.6±10.3	13.4±2.2	24.6±6.7	122.7±7.8	218.9±14.0	81.2±2.4	313.7±12.2
	OT-M (ours)	**86.0±2.2**	**132.7±3.3**	**12.8±1.4**	**22.0±4.5**	**120.1±7.3**	**208.9±11.7**	**80.9±3.1**	**303.1±9.5**
10%	DAC [26]	84.8±4.5	140.9±11.3	11.1±0.5	18.9±1.9	110.5±5.9	196.0±16.3	76.0±2.0	293.8±10.4
	OT-M (ours)	**81.6±2.6**	**127.1±3.8**	**10.9±0.5**	**18.1±1.4**	**107.9±4.1**	**180.6±7.8**	**75.5±1.6**	**287.9±11.1**
40%	DAC [26]	71.6±2.0	120.8±5.6	**9.0±0.3**	14.6±0.5	**91.8±4.7**	161.4±12.4	**64.1±3.0**	270.6±9.3
	OT-M (ours)	**70.0±2.2**	**113.0±6.9**	9.0±0.4	**14.2±0.7**	93.4±5.4	**157.5±7.8**	66.5±3.1	**268.2±9.5**

Table 4. Comparison with DAC [26] averaged over 5 trials (mean±std), where each trial uses different randomly sampled labeled data.

Data	gate	confidence	MAE	MSE
label only			145.59	257.31
	✓		144.48	255.33
	✓	✓	**138.52**	**242.26**
Data	**loss for unlabeled data**		**MAE**	**MSE**
label+unlabel	L2 loss		137.17	239.52
	L2 w/ confidence		135.88	233.19
	GL		125.32	214.96
	GL w/ confidence (C-GL)		**120.13**	**208.87**

Table 5. Ablation study on 5% label data of UCF-QNRF.

Method	MAE	MSE
Label only	138.52±10.65	242.26±16.62
LM [58]	148.53±9.53	270.25±23.67
GMM [14]	126.67±7.41	217.00±16.17
OT-M (ours)	**120.13±7.34**	**208.87±11.65**

Table 6. Ablation study on semi-supervised counting when using different density-map localization methods. (mean±std).

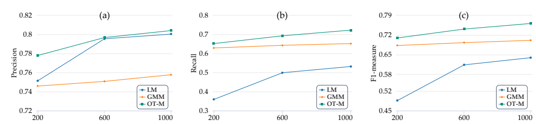


Figure 7. The localization performance on unlabeled data improves during training. (x-axis is the training epoch.)

and confidence-weights on *labeled* data. With gating, MAE is reduced from 145.59 to 144.48. The gap is small since most loss is from the transport term, $\langle \mathbf{C}, \mathbf{P} \rangle$, but the MAE and MSE still decrease by 1.11 and 1.94. As mentioned in Sec. 4.2, confidence weights can also be applied to labeled data to suppress annotation noise. Relevant experimental results demonstrate its effectiveness – it helps the counting model achieve better performance (MAE: 138.52).

Hard labels vs. soft labels. Next, we compare the performance while unlabeled data is used during training. We compare our framework with the soft pseudo-labels (predicted density maps) using L2 loss. We also design a confidence strategy for L2 loss:

$$\mathbf{w}' = \exp[-\gamma' (\mathrm{diag}(\mathbf{a}_t)^{-1} |\mathbf{a}_t - \mathbf{a}_s|)], \tag{23}$$

where $\mathbf{a}_t$ and $\mathbf{a}_s$ represent density maps predicted by the teacher and student, and γ' is similar to γ in (20). The results are shown in the bottom half of Tab. 5 – the counting model can predict more accurately under the guidance of confidence strategy during training, regardless of using soft or hard labels. However, using hard pseudo-labels for unlabeled samples reduces estimation errors dramatically, compared to soft labels. Using the confidence-weights with GL also greatly improves the MAE and MSE (e.g., MAE 125.3 drops to 120.13).

Localization method. Finally, we consider different density-map localization methods for generating hard pseudo-labels in our semi-supervised counting framework, as presented in Tab. 6. LM [58] performs even worse than training with only labeled data, which is because the number of points generated by LM could be different from the count in the teacher's density map, *i.e.*, the number of local maxima in the density map is not guaranteed to be the sum of the density map. In contrast, both GMM [14] and OT-M sets the number of pseudo-points as the sum of the predicted density map, and then estimates their locations. Compared with GMM, OT-M obtains more accurate pseudo hard-label, which leads to counting models that can better capture information from the unlabeled, and thus improves the semi-supervised learning performance. Specifically, Fig. 7 shows the localization performance on the unlabeled data during semi-supervised training. Generally, the localization improves during training, while OT-M obtains the best localization accuracy, i.e., the most accurate hard pseudo-labels.

6. Conclusion

This paper presents a parameter-free crowd localization method on density map, the OT-M algorithm. OT-M alternates between two steps: in the OT-step, the transport plan between the current point map and the input density map is estimated; in the M-step, the point map is updated using the transport plan computed in the OT-step. The convergence of OT-M is analyzed both in theory and practice. Experiments also show that OT-M outperforms previous localization methods based on density maps, as well as recent point detection methods. Furthermore, we apply OT-M to semi-supervised counting to produce hard pseudo-labels, and we propose a confidence-weighted generalized loss for this task, which assigns lower confidence to unlabeled data with inconsistency between teacher's labels and student's predictions. Empirical results demonstrate that efficacy of our framework on several crowd counting datasets.

Acknowledgements. This work was supported by a Strategic Research Grant from City University of Hong Kong (Project No. 7005665).

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*, volume 35, pages 872–881, 2021. 1

[2] Carlos Arteta, Victor Lempitsky, and Andrew Zisserman. Counting in the wild. In *European conference on computer vision*, pages 483–498. Springer, 2016. 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, 3

[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. 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, 2

[6] Xiaokang Chen, Yuhui Yuan, Gang Zeng, and Jingdong Wang. Semi-supervised semantic segmentation with cross pseudo supervision. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2613–2622, 2021. 1, 4

[7] Marco Cuturi. Sinkhorn distances: Lightspeed computation of optimal transport. *Advances in neural information processing systems*, 26, 2013. 1, 3

[8] Jean Feydy, Thibault Séjourné, François-Xavier Vialard, Shun-ichi Amari, Alain Trouvé, and Gabriel Peyré. Interpolating between optimal transport and mmd using sinkhorn divergences. In *The 22nd International Conference on Artificial Intelligence and Statistics*, pages 2681–2690. PMLR, 2019. 3

[9] Geoffrey French, Samuli Laine, Timo Aila, Michal Mackiewicz, and Graham D. Finlayson. Semi-supervised semantic segmentation needs strong, varied perturbations. In *31st British Machine Vision Conference*, 2020. 4

[10] Junyu Gao, Tao Han, Yuan Yuan, and Qi Wang. Learning independent instance maps for crowd localization. *arXiv preprint arXiv:2012.04164*, 2020. 1

[11] Junyu Gao, Qi Wang, and Xuelong Li. Pcc net: Perspective crowd counting via spatial convolutional network. *IEEE Transactions on Circuits and Systems for Video Technology*, 30(10):3486–3498, 2019. 2

[12] Haroon Idrees, Imran Saleemi, Cody Seibert, and Mubarak Shah. Multi-source multi-scale counting in extremely dense crowd images. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2547–2554, 2013. 2

[13] 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. 1, 2, 6, 7

[14] Di Kang, Zheng Ma, and Antoni B Chan. Beyond counting: comparisons of density maps for crowd analysis tasks—counting, detection, and tracking. *IEEE Transactions on Circuits and Systems for Video Technology*, 29(5):1408–1422, 2018. 1, 2, 6, 7, 8

[15] Donghyeon Kwon and Suha Kwak. Semi-supervised semantic segmentation with error localization network. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 9957–9967, 2022. 5

[16] Samuli Laine and Timo Aila. Temporal ensembling for semi-supervised learning. *arXiv preprint arXiv:1610.02242*, 2016. 2

[17] Issam H Laradji, Negar Rostamzadeh, Pedro O Pinheiro, David Vazquez, and Mark Schmidt. Where are the blobs: Counting by localization with point supervision. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 547–562, 2018. 1

[18] 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*, volume 3, page 896, 2013. 4

[19] 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)*, volume 1, pages 878–885. IEEE, 2005. 2

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

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

[22] 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

[23] Dingkang Liang, Wei Xu, and Xiang Bai. An end-to-end transformer model for crowd localization. In *Proceedings of the European conference on computer vision*, volume 13661, pages 38–54, 2022. 7

[24] Dingkang Liang, Wei Xu, Yingying Zhu, and Yu Zhou. Focal inverse distance transform maps for crowd localization. *IEEE Transactions on Multimedia*, pages 1–13, 2022. 1

[25] 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, 8 2021. 2

[26] Hui Lin, Zhiheng Ma, Xiaopeng Hong, Yaowei Wang, and Zhou Su. Semi-supervised crowd counting via density agency. In *ACM Multimedia*, 2022. 2, 5, 6, 7, 8

[27] 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. 5, 7

[28] Chenchen Liu, Xinyu Weng, and Yadong Mu. Recurrent attentive zooming for joint crowd counting and precise localization. In *Proceedings of the IEEE/CVF Conference*

on Computer Vision and Pattern Recognition, pages 1217–1226, 2019. 7

[29] 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

[30] 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

[31] 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

[32] Erika Lu, Weidi Xie, and Andrew Zisserman. Class-agnostic counting. In *Asian conference on computer vision*, pages 669–684. Springer, 2018. 1

[33] 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

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

[35] 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*, volume 35, pages 2319–2327, 2021. 2

[36] Zheng Ma, Lei Yu, and Antoni B Chan. Small instance detection by integer programming on object density maps. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3689–3697, 2015. 1, 2

[37] Gonzalo Mena, Amin Nejatbakhsh, Erdem Varol, and Jonathan Niles-Weed. Sinkhorn em: an expectation-maximization algorithm based on entropic optimal transport. *arXiv preprint arXiv:2006.16548*, 2020. 4

[38] 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. 1, 2, 7

[39] Todd K Moon. The expectation-maximization algorithm. *IEEE Signal processing magazine*, 13(6):47–60, 1996. 4

[40] Lin Niu, Xinggang Wang, Chen Duan, Qiongxia Shen, and Wenyu Liu. Local point matching network for stabilized crowd counting and localization. In *Chinese Conference on Pattern Recognition and Computer Vision (PRCV)*, pages 566–579. Springer, 2022. 1

[41] Gabriel Peyré, Marco Cuturi, et al. Computational optimal transport: With applications to data science. *Foundations and Trends® in Machine Learning*, 11(5-6):355–607, 2019. 3, 5

[42] Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun. Faster r-cnn: Towards real-time object detection with region proposal networks. *Advances in neural information processing systems*, 28, 2015. 7

[43] Weihong Ren, Di Kang, Yandong Tang, and Antoni B Chan. Fusing crowd density maps and visual object trackers for people tracking in crowd scenes. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 5353–5362, 2018. 2

[44] 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. 2

[45] Mikel Rodriguez, Ivan Laptev, Josef Sivic, and Jean-Yves Audibert. Density-aware person detection and tracking in crowds. In *2011 International Conference on Computer Vision*, pages 2423–2430. IEEE, 2011. 2

[46] Bernhard Schmitzer. Stabilized sparse scaling algorithms for entropy regularized transport problems. *SIAM Journal on Scientific Computing*, 41(3):A1443–A1481, 2019. 3

[47] 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. 7

[48] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*, 2014. 7

[49] 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

[50] Vishwanath A Sindagi, Rajeev Yasarla, and Vishal M Patel. Jhu-crowd++: Large-scale crowd counting dataset and a benchmark method. *Technical Report*, 2020. 1, 6

[51] Richard Sinkhorn. A relationship between arbitrary positive matrices and doubly stochastic matrices. *The annals of mathematical statistics*, 35(2):876–879, 1964. 3

[52] 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

[53] 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. 4

[54] 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 *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 3365–3374, 2021. 1, 7

[55] 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, 4, 7

[56] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. *Advances in neural information processing systems*, 30, 2017. 7

[57] Jia Wan and Antoni Chan. Modeling noisy annotations for crowd counting. *Advances in Neural Information Processing Systems*, 33:3386–3396, 2020. 5

[58] 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, 4, 5, 6, 7, 8

[59] 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*, 2020. 1

[60] 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

[61] 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, 6, 7

[62] 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. 1

[63] Qi Wang, Jia Wan, and Yuan Yuan. Locality constraint distance metric learning for traffic congestion detection. *Pattern Recognition*, 75:272–281, 2018. 1

[64] Qizhe Xie, Minh-Thang Luong, Eduard Hovy, and Quoc V Le. Self-training with noisy student improves imagenet classification. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 10687–10698, 2020. 4

[65] Ji Zhang, Zhi-Qi Cheng, Xiao Wu, Wei Li, and Jian-Jun Qiao. Crossnet: Boosting crowd counting with localization. In *Proceedings of the 30th ACM International Conference on Multimedia*, pages 6436–6444, 2022. 1

[66] 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. 1

[67] 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

[68] Yuliang Zou, Zizhao Zhang, Han Zhang, Chun-Liang Li, Xiao Bian, Jia-Bin Huang, and Tomas Pfister. Pseudoseg: Designing pseudo labels for semantic segmentation. In *9th International Conference on Learning Representations*, 2021. 4