Modeling Noisy Annotations for Point-wise Supervision

Jia Wan, Qiangqiang Wu, and Antoni B. Chan

Abstract—Point-wise supervision is widely adopted in computer vision tasks such as crowd counting and human pose estimation. In practice, the noise in point annotations may affect the performance and robustness of algorithm significantly. In this paper, we investigate the effect of annotation noise in point-wise supervision and propose a series of robust loss functions for different tasks. In particular, the point annotation noise includes spatial-shift noise, missing-point noise, and duplicate-point noise. The spatial-shift noise is the most common one, and exists in crowd counting, pose estimation, visual tracking, *etc*, while the missing-point and duplicate-point noises usually appear in dense annotations, such as crowd counting. In this paper, we first consider the shift noise by modeling the real locations as random variables and the annotated points as noisy observations. The probability density function of the intermediate representation (a smooth heat map generated from dot annotations) is derived and the negative log likelihood is used as the loss function to naturally model the shift uncertainty in the intermediate representation. The missing and duplicate noise are further modeled by an empirical way with the assumption that the noise appears at high density region with a high probability. We apply the method to crowd counting, human pose estimation and visual tracking, propose robust loss functions for those tasks, and achieve superior performance and robustness on widely used datasets.

Index Terms—Noisy point annotations, crowd counting, object counting, tracking, pose estimation, deep learning

1 Introduction

Point annotations are widely used in computer vision tasks such as crowd counting [1, 2] and human pose estimation [3, 4]. In crowd counting, a coordinate is annotated to roughly indicate the location of a person. Although we care more about the total number of persons in an image instead of their precise locations, the rough location provides important information for the distribution of the crowd. Then, the model is trained to predict the total count in a given image, typically through predicting an intermediate representation (crowd density map) based on the point annotations [5]. In human pose estimation, the human joints are annotated by a set of points and the model is trained to locate those joints. Then, human pose can be inferred based on the locations of the detected joints locations [4]. For other applications such as visual tracking and object detection, an object is represented by a location and a scale. Therefore, the location of an object/part plays central role in computer vision.

Unfortunately, the location of an object/part is ambiguous due to a variety of reasons, such as occlusion and human labeling error. Therefore, noise commonly exists in point annotations and may affect the performance and robustness of algorithms significantly. In crowd counting, there are potentially thousands of people to be annotated in one image, which makes it easy to make a mistake during labeling. Moreover, people in the crowd are occluded by each other, which makes it even harder to locate a person during labeling. Thus, annotation noise is common for all crowd counting datasets. In human pose estimation, the annotation noise mainly comes from the definition of the joints and the occlusion by clothes (e.g., loose clothing will obscure the actual joint location). For visual tracking and object detection, the center location of an object is usually defined by the center of the bounding box. However, it is not always accurate since the object shape varies.

Since the definition and annotation of locations are ambiguous and noisy, typical methods predict an intermediate result instead of the annotation coordinates directly, which provides better supervision for neural networks. In those methods, an intermediate representation is first generated by convolving the dot map with a unit ball/Gaussian kernel, resulting in a *density map* or *heat map*. Then, the model is learned to predict the intermediate representation which is smoother and easier to predict. Typical loss function used to train the model is Mean Squared Error (MSE) which assumes isotropic Gaussian per-pixel noise in the density/heat intermediate maps. However, this assumption is erroneous since the pixels in the intermediate map are correlated due to the convolution operation. Therefore, traditional loss functions suffer from the problems caused by mismatch between the assumed per-pixel noise and the actual annotation noise. First, the model is easy to overfit without a proper representation of annotation noise. Second, in the presence of large amounts of annotation noise, the model tends to predict smooth maps which are not suitable for localization.

To address this issue, we first propose to explicitly model the point-wise shift noise and derive the distribution of the intermediate representation. By using the negative log likelihood as the loss function, the uncertainty of annotation noise is naturally modeled. In particular, the real location is considered as a random variable and the annotation is treated as the noisy observation of the true location. By assuming the spatial annotation noise as Gaussian, the probability density function (pdf) of the pixels in the intermediate representation is derived, but lacks a closed-form solution. Hence, we approximate the pdf as a multivariate Gaussian by deriving the mean, variance and covariance between pixels in the intermediate

• *Jia Wan, Qiangqiang Wu, and Antoni B. Chan (corresponding author) are with the Department of Computer Science, City University of Hong Kong. E-mail: jiawan1998@gmail.com, qiangqwu2-c@my.cityu.edu.hk, abchan@cityu.edu.hk.*

representations. To decrease the computation cost, a low-rank approximation to the covariance matrix is proposed. Once the pdf of the intermediate representation is obtained, the negative log likelihood is used as the loss function which decomposes into a weighted MSE term and a correlation term. The weighted MSE term pays less attention to the uncertain regions, and the uncertainty is correlated to the k nearest distances to the annotation points. Motivated by this observation, we further model the missing-point and duplicate-point noise by a combination of a weighted pixel and a weighted point loss function, where the weight is determined by the k nearest distances to the annotation points. The weighted point loss assumes that the summation of density around a person is a distribution. If several points are missing in a region, the pixel prediction and the point prediction in this region should be larger. The proposed loss function is effective in modeling the three types of noise in point annotations.

The contributions of this paper are summarized as follows:

1) We propose to model the true ground-truth point locations as random variables and point annotations as noisy observations of the true locations. The uncertainty of the spatial noise is effectively considered during training, which improves the robustness of algorithms.
2) We derive a parametric-based robust loss function that is a generalization of the popular MSE to model the spatial shift noise. The correlation between pixels in the intermediate representation is naturally considered.
3) We further model the missing-point and duplicate-point noise by an empirical robust loss function based on the analysis of the parametric-based robust loss.
4) We analyze the effectiveness of the proposed loss function for different tasks under different noise levels. The proposed robust loss significantly outperforms the traditional loss function when learning from noisy annotations.

A preliminary version of our work appears in our conference paper [1]. The main differences between the preliminary conference version and this paper are three-fold. First, we introduce a new model for the missing-point and duplicate noise-point, which works in conjunction with the model for shift noise from [1]. Second, we propose a new empirical approximation to our loss function to accelerate its efficiency. Finally, we include new experiments on visual tracking and human pose estimation, as well as crowd counting with missing/duplicate noise.

The remainder of the paper is organized as follows. The relevant works are reviewed in Section 2. Then, the proposed methods are described in Section 3. Afterwards, the experimental results are presented and discussed in Section 4, we conclude the paper in Section 5.

2 Related Works

In this section, we briefly review the relevant works of crowd counting, human pose estimation, visual tracking, and methods for modeling label noise.

2.1 Crowd Counting

Traditional crowd counting algorithms detect human bodies [6] or body parts [7], and count the number of the bounding boxes. However, these methods are not capable of handing severe occlusions in crowd scenes. Therefore, direct regression methods are proposed to predict the count directly from low-level features [8], such as texture [9] and color [10]. The performance is still limited due to the large variation of scale and scenes.

In recent years, deep learning based approaches are proposed and achieve superior performance. Different network architectures [5, 11] are proposed to extract multi-scale features due to the scale variation caused by perspective transformation in crowd images. Kang and Chan [12] propose to use a image pyramid to handle scale variations. To further improve the quality of the prediction, refinement based methods are proposed. Context information are exploited by Sindagi and Patel [13], Xiong et al. [14] to improve the counting performance. To improve the generalization ability of the trained model, Zhang et al. [15] propose a cross-scene fine-tuning method, while Wang et al. [16] propose a synthetic dataset. The correlation information is also utilized to improve the generalization ability [17, 18]. However, those methods rely on the quality of intermediate representation ground-truths.

To learn better intermediate representation, Wan and Chan [19] propose an end-to-end learning algorithm to adaptively learn density map ground-truth from dot annotations. An individual kernel learning method is then proposed by Wan et al. [20]. Wang et al. [2], Wan et al. [21] propose to directly utilize point annotations as the supervision. Song et al. [22] propose a purely point-based framework that measures the difference between the predicted points and the ground-truth for both crowd counting and localization. However, most of these methods do not explicitly consider the annotation noise while our loss can model the fundamental annotation noise according to a generative process.

Semi-supervised crowd counting methods are proposed to relieve the annotation burden. Loy et al. [23] propose to utilize the abundant unlabeled data in videos, while Meng et al. [24] propose to exploit the spatial uncertainty for semi-supervised counting. Xu et al. [25] propose to use partial annotations in images for training. These methods mainly consider the uncertainty of the prediction, while here we focus on the annotation uncertainty.

Finally, more recently, the transformer-based models are proposed to improve the performance of crowd counting [26] and localization [27]. Yang et al. [28] propose an overlapped patching method, while Liang et al. [27] propose an end-to-end crowd localization method, which directly matches the predicted and the ground-truth points. Shu et al. [29] propose a new loss function derived in the frequency domain, while Liu et al. [30] leverage self-supervised learning to improve the performance of cross-domain crowd counting. Zhang and Chan [31] propose a calibration free multi-view crowd counting algorithm.

2.2 Human Pose Estimation

Human pose estimation algorithms predict different heat maps for the corresponding joints, and the maximum response in a map indicates the location of the joint [3]. These methods can be divided into to two categories: top-down and bottom-up.

Top-down algorithms first detect each individual in an image and then predict the joint locations for the individual. A simple baseline method is proposed by Xiao et al. [4]. Newell et al. [32] propose stacked hourglass networks for human pose estimation. These methods rely on the quality of detection results, and thus Fang et al. [33] propose a regional-based framework to handle inaccurate bounding boxes of the detector.

Bottom-up algorithms first detect all joints in the image and then group those joints into multiple persons [34]. Pishchulin et al. [35] propose to solve the detection and association jointly using

integer linear programming. Geng et al. [36] propose to use a pixel-wise spatial transformer to regress joint locations. The annotation noise is seldom considered in human pose estimation, while here we investigate the performance of different loss functions for noisy data.

2.3 Visual Tracking

In visual object tracking, deep learning-based trackers are dominant methods that achieve competitive performance on existing tracking benchmarks [37, 38, 39]. The deep correlation filter (CF) trackers [40, 41, 42, 43, 44, 45] are the first to use deep features for deep tracking. Inspired by the great potential of deep learning, many Siamese network-based trackers [46, 47] are proposed in recent years. The improvements of these methods include tracking network design [48, 49, 50, 51] and online target appearance modeling [52, 53, 54, 55]. However, the noisy bounding box annotations in existing large-scale training datasets [38, 39] may severely degrade the learning of these end-to-end trainable deep trackers. To handle noisy annotated bounding boxes, PUL [56] proposes a noisy robust binary-cross entropy (BCE) loss by integrating out the label noise in the likelihood estimation. PrDiMP [57] trains a tracking model to minimize the KL divergence between the generated target response and the ground-truth Gaussian conditional distribution derived from label noise. Different from these methods, our method explicitly models annotation noise as a random variable with Gaussian distributions, and we confirm its effectiveness in learning robust tracking representations from noisy annotations.

2.4 Modeling Label Noise

Previous works mainly focus on classification task with class label noise. Robust loss functions are proposed by Wang et al. [58], while label cleaning [59] and sample selection [60] are proposed to filter noisy annotations in classification. Yang et al. [61] propose to deal with the noisy segmentation boundary. Natarajan et al. [62] theoretically study binary classification with random classification noise. A few recent works consider filtering out noisy samples; Wang et al. [63] propose a statistical sample selection framework to identify noisy data, while FINE [64] is proposed to filter noisy samples based on their eigenvectors, and Bai et al. [65] exploit early stopping with noisy labels. Other recent approaches are proposed in specific domain areas: Liang et al. [66] propose a few-shot learning method with noisy labels by leveraging an attention mechanism; Fu et al. [67] propose a large-scale pre-training method with noisy labels for person re-identification; Liu et al. [68] propose a correction mechanism for segmentation with noisy annotations.

In crowd counting, the annotation noise is yet not fully-exploited. Bai et al. [69] propose the correct the annotation noise from the prediction, but may be prone to degenerate solutions. In contrast, we consider the annotation noise in a probabilistic way and model the transformation of noise from annotation points to the intermediate representation, which is able to handle large annotation noise as shown in our experiments. Similar to crowd counting, the annotation noise in human pose estimation is seldom considered. Kato et al. [70] propose to correct the missing annotations caused by occlusion, but the spatial shift noise is not considered. In this paper, we investigate the shift noise in the point annotations in a probabilistic fashion, which is demonstrated to be more robust to large noise. We also take the missing and duplicate noise into consideration.

3 METHODOLOGY

In this section, we first review the traditional method for generating the intermediate representations. Then, we propose a parametric modeling for shift noise in point annotations and an efficient approximation is proposed for practical training. Finally, the missing-point and duplicate-point noise are modeled based on an empirical approach.

3.1 Intermediate Representation Generation

In general, the point annotations are not directly used as the ground-truth during training because they are noisy and easy to overfit. An intermediate representation (i.e. a smooth heat map) is generated by convolving the dot map with a Gaussian kernel, which is equivalent to placing a Gaussian kernel at each annotation point. Given an input image $\mathcal{I}$, there are N annotation points $\{\tilde{D}_i\}_{i=1}^N$, where each point indicates the location of a person in the image. The value y at the 2D location $\mathbf{x}$ in the corresponding intermediate representation is defined as:

$$y(\mathbf{x}) = \sum_{i=1}^{N} \mathcal{N}(\mathbf{x}|\tilde{\mathbf{D}}_i, \beta\mathbf{I}) = \sum_i \frac{1}{2\pi\beta} e^{-\frac{1}{2}||\mathbf{x}-\tilde{\mathbf{D}}_i||^2_{\beta\mathbf{I}}}, \quad (1)$$

where β is the squared bandwidth of the Gaussian kernel and $\mathcal{N}(\mathbf{x}|\boldsymbol{\mu}, \boldsymbol{\Sigma})$ is the pdf of a multivariate Gaussian with $\boldsymbol{\mu}$ as the mean and $\boldsymbol{\Sigma}$ as the covariance matrix. The values of $y(\mathbf{x})$ for all locations $\mathbf{x}$ in the image form an intermediate representation. This representation is usually called a *density map* in crowd counting [5], or a *heat map* or *response map* in human pose estimation [3] and visual tracking [56].

After the intermediate representation y is generated, a regressor $f(\mathcal{I})$ is learned to predict y from the input image $\mathcal{I}$ with the L2 loss function, $\mathcal{L}(\mathbf{y}, f(\mathcal{I})) = ||\mathbf{y} - f(\mathcal{I})||^2$, where $\mathbf{y}$ is the intermediate map evaluated at the output locations of $f(\mathcal{I})$. This traditional framework is illustrated by the orange and green arrows in Fig. 1. A standard result [71] shows that L2 loss assumes i.i.d Gaussian noise between the observations $\mathbf{y}$ and the underlying function output $f(\mathcal{I})$, which is erroneous here since the observation noise (in the intermediate representation) is induced from the uncertainty in the annotation via a non-linear transformation as in (1).

3.2 Modeling Shift Noise

We consider the annotations $\{\tilde{D}_i\}_{i=1}^N$ as the noisy observation of people's true locations in image $\mathcal{I}$ as shown in Fig 1. Let the true location of the i-th person be a random variable (r.v.) D_i, where $D_i = \tilde{D}_i + \boldsymbol{\epsilon}_i$ and $\boldsymbol{\epsilon}_i$ is the spatial annotation noise, and we assume $\boldsymbol{\epsilon}_i$ is i.i.d multivariate Gaussian noise, $\boldsymbol{\epsilon}_i \sim \mathcal{N}(0, \alpha\mathbf{I})$, where α is the variance of the Gaussian. Then, the intermediate representation can be generated using the true locations. Specifically, at location $\mathbf{x}$, the density value $\Phi(\mathbf{x})$ is

$$\begin{aligned}\Phi(\mathbf{x}) &= \sum_{i=1}^{N} \mathcal{N}(\mathbf{x}|\mathbf{D}_i, \beta\mathbf{I}) = \sum_{i=1}^{N} \mathcal{N}(\mathbf{x}|\tilde{\mathbf{D}}_i + \boldsymbol{\epsilon}_i, \beta\mathbf{I}) \\ &= \sum_{i=1}^{N} \mathcal{N}(\mathbf{q}_i^{\mathbf{x}}|\boldsymbol{\epsilon}_i, \beta\mathbf{I}) \triangleq \sum_i \phi_i(\mathbf{x}), \end{aligned} \quad (2)$$

where $\phi_i(\mathbf{x})$ indicates the term for each individual annotation, and $\mathbf{q}_i^{\mathbf{x}} = \mathbf{x} - \tilde{\mathbf{D}}_i$ is the difference between the i-th annotation location and $\mathbf{x}$. Since $\boldsymbol{\epsilon}_i$ is a random variable, the density value $\Phi(\mathbf{x})$ at a location $\mathbf{x}$ is also a random variable.

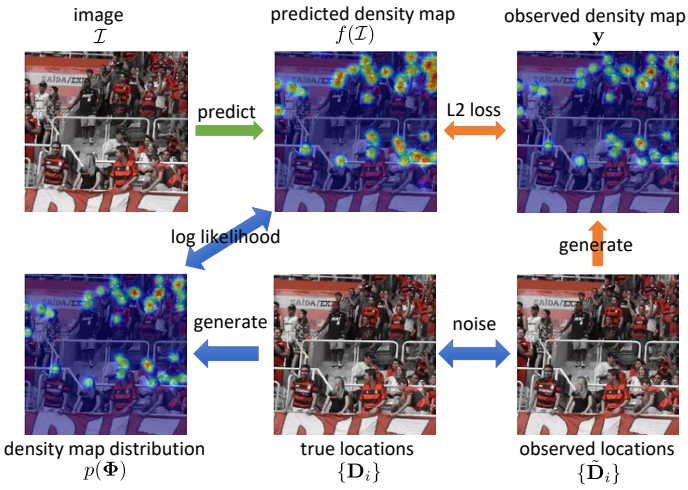

Fig. 1. Our framework for modeling noisy annotations. Blue arrows represent our model, and orange arrows are the traditional intermediate representation-based method. Note that we use crowd counting as an example.

Let $\{\mathbf{x}^{(\eta)}\}_{\eta=1}^{P}$ be the P locations in the image, and $\Phi^{(\eta)} \triangleq \Phi(\mathbf{x}^{(\eta)})$ be their corresponding density value random variables. Then the vectorized density map $\mathbf{\Psi} = [\Phi^{(1)}, \cdots, \Phi^{(P)}]$ is a multivariate r.v. whose individual entries are from (2). Note that the density values $\mathbf{\Psi}$ are related through a spatial convolution operation, and thus the values in neighboring locations are correlated.

As the multivariate r.v. $\mathbf{\Psi}$ is complex, we first derive its marginal distribution $\Phi(\mathbf{x})$ and an efficient Gaussian approximation of Φ. Then, we propose to approximate the joint distribution of $\mathbf{\Psi}$ with a m.v. Gaussian.

3.2.1 Probability distribution of $\Phi(\mathbf{x})$

We now consider the marginal of $\mathbf{\Psi}$, which corresponds to the pdf of $\Phi(\mathbf{x})$ at locations $\mathbf{x}$. First, the pdf of $\Phi(\mathbf{x})$ can be derived by passing the r.v.s $\{\boldsymbol{\epsilon}_i\}_i$ through the non-linear transformation defined in (2). The individual term $\phi_i(\mathbf{x}) = \frac{1}{2\pi\beta}\exp(-\frac{1}{2}||\mathbf{q}_i^{\mathbf{x}} - \epsilon_i||^2_{\beta\mathbf{I}})$ consists of a series of transformations: squared L2 norm, negative exponential, and scaling. Note that the squared L2 norm of a m.v. Gaussian random variable with non-zero mean is a non-central χ^2 distribution. Then, the density of $\phi_i(\mathbf{x})$ is obtained by applying formulas for the transformation of a random variable (see Supp. for derivation) as

$$\phi_i(\mathbf{x}) \sim p_i(\phi_i|\mathbf{x}) = \tfrac{\delta}{h} e^{-\frac{\lambda_i^{\mathbf{x}}}{2}} \left(\tfrac{\phi_i}{h}\right)^{\delta-1} I_0\Big(\sqrt{-2\delta\lambda_i^{\mathbf{x}} \log \tfrac{\phi_i}{h}}\Big), \tag{3}$$

where $h = (2\pi\beta)^{-1}$ denotes the maximum value of the Gaussian kernel, $\delta = \beta/\alpha$, and $I_0(x)$ indicates a modified Bessel function of the 1st kind of order 0. $\lambda_i^{\mathbf{x}} = \frac{1}{\alpha}||\mathbf{q}_i^{\mathbf{x}}||^2$ is the non-centrality parameter of a non-central χ^2 r.v., which depends on $\mathbf{x}$. Second, since we assume the noise $\boldsymbol{\epsilon}_i$ for each annotation are independent, the resulting individual terms $\phi_i(\mathbf{x})$ are also independent r.v.s. Thus, the pdf of the sum $\Phi(\mathbf{x}) = \sum_i \phi_i(\mathbf{x})$ is the convolution of the pdfs of the individual terms,

$$\Phi(\mathbf{x}) \sim p(\Phi|\mathbf{x}) = p_1(\Phi|\mathbf{x}) * p_2(\Phi|\mathbf{x}) * \cdots * p_N(\Phi|\mathbf{x}), \tag{4}$$

where $*$ is the convolution operation. Unfortunately, this convolution is intractable to compute in closed form.

3.2.2 Gaussian approximation to $\Phi(\mathbf{x})$

Since (4) is intractable, we approximate the distribution of $\Phi(\mathbf{x})$ using a Gaussian, $\hat{p}(\Phi|\mathbf{x}) = \mathcal{N}(\Phi|\mu^{\mathbf{x}}, \upsilon^{\mathbf{x}})$, where $\mu^{\mathbf{x}}$ and $\upsilon^{\mathbf{x}}$ are mean and variance of the distribution for location $\mathbf{x}$. The mean of Φ is calculated as (see detailed derivation in Supp.)

$$\begin{aligned}\mu^{\mathbf{x}} &= \mathbb{E}[\Phi|\mathbf{x}] = \mathbb{E}\Big[\sum_i \mathcal{N}(\mathbf{q}_i^{\mathbf{x}}|\boldsymbol{\epsilon}_i, \beta\mathbf{I})\Big|\mathbf{x}\Big] \\ &= \sum_i \mathcal{N}(\mathbf{q}_i^{\mathbf{x}}|\mathbf{0}, (\alpha+\beta)\mathbf{I}) \triangleq \sum_i \mu_i^{\mathbf{x}},\end{aligned} \tag{5}$$

where $\mu_i^{\mathbf{x}}$ indicates the mean for the individual term $\phi_i(\mathbf{x})$, and the variance is

$$\begin{aligned}\upsilon^{\mathbf{x}} &= \mathrm{var}(\Phi|\mathbf{x}) = \mathbb{E}[\Phi^2|\mathbf{x}] - \mathbb{E}[\Phi|\mathbf{x}]^2 \\ &= \sum_i \frac{1}{4\pi\beta}\mathcal{N}(\mathbf{q}_i^{\mathbf{x}}|\mathbf{0}, (\beta/2+\alpha)\mathbf{I}) - \sum_i (\mu_i^{\mathbf{x}})^2.\end{aligned} \tag{6}$$

Figure 2 (a-c) shows an example with three annotation points and the corresponding marginal distributions of $\Phi(\mathbf{x})$ for two spatial locations.

We use a Gaussian approximation because it is tractable and can be estimated from the 1st and 2nd moments of $\Phi(\mathbf{x})$. Extensions of the central limit theorem prove that sums of independent non-identical r.v.s converge to Gaussian. As shown in Fig. 2 (c), the distribution is tending to Gaussian with just 3 annotations, and we observe that this tendency becomes stronger with more annotations. We have also tried Gamma distributions for the approximation, but the results are worse (MAE is 89.7 on UCF-QNRF compared to 85.8 for the Gaussian approximation).

To further demonstrate the suitability of the Gaussian approximation for high-density regions, we run a simulation experiment. First, we randomly select an image and its corresponding GT dot map from the training set, and generate samples from $\Phi(\mathbf{x})$ for each location $\mathbf{x}$. Then, we run D'Agostino and Pearson's normality-test [72] on the samples at each location $\mathbf{x}$. Fig. 3 (left) shows an example density map and the corresponding regions where the normality test indicates a Gaussian distribution of $\Phi(\mathbf{x})$.[1] The locations with high density values (i.e., closer to the ground-truth dots) are more likely to follow a Gaussian distribution. Fig. 3 (right) shows a histogram of the percentage of Gaussian-distributed locations versus average density value bins. Over 80% of the locations with average density values over 0.31 are Gaussian distributed. Since these locations most affect the density map and count, it is better to use a Gaussian approximation in these regions. Using Gamma distribution will only potentially fit better to the sparse regions (with small count), at the expense of poorly fitting the important regions with larger density.

3.2.3 Gaussian approximation to joint likelihood of $\mathbf{\Psi}$

The previous derivation independently approximates each spatial location $\mathbf{x}$. We next consider the correlation between locations via a m.v. Gaussian approximation to the joint likelihood $\mathbf{\Psi}$. Given the spatial locations $\mathbf{x}^{(\eta)}$, where $\eta \in \{1, \cdots, P\}$, let $\mathbf{q}_i^{(\eta)} = \mathbf{x}^{(\eta)} - \tilde{\mathbf{D}}_i$ be the difference between the i-th annotation and the pixel location $\mathbf{x}^{(\eta)}$. From (2), the density value r.v. $\Phi^{(\eta)}$ at pixel location $\mathbf{x}^{(\eta)}$ is

$$\Phi^{(\eta)} = \sum_{i=1}^{N} \mathcal{N}(\mathbf{q}_i^{(\eta)}|\boldsymbol{\epsilon}_i, \beta\mathbf{I}) \triangleq \sum_i \phi_i^{(\eta)}. \tag{7}$$

1. Specifically, the null hypothesis that the samples are from a Gaussian distribution cannot be rejected, $p > 0.05$.

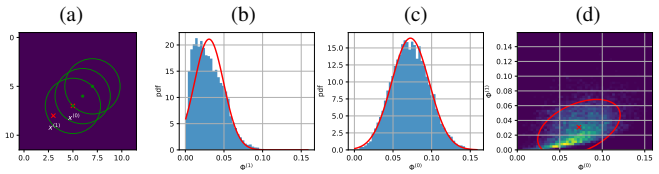

Fig. 2. Example of probability distributions of density values at two locations: (a) three annotations are shown in green, where the circles represent 2 standard deviations of the noise; (b) marginal distribution of density values at $\mathbf{x}^{(1)}$ and (c) $\mathbf{x}^{(0)}$; (d) the joint distribution of density values $(\Phi^{(1)}, \Phi^{(0)})$. The histograms in (b-d) are obtained using sampling, and the red lines represent the Gaussian approximations.

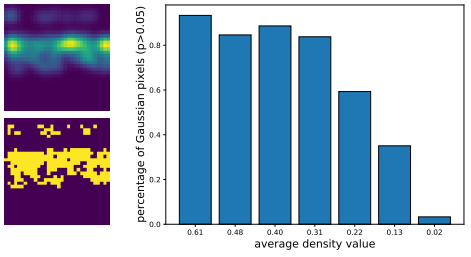

Fig. 3. Suitability of the Gaussian approximation in high-density regions: (left-top) Density map and (left-bottom) yellow regions show pixels that are Gaussian distributed according to a normality test. (right) Normality test results vs. average density value bins. In the normality test, the null hypothesis is that the pixel is Gaussian distributed.

Here the superscript (η) indicates evaluating/conditioning on the location $\mathbf{x}^{(\eta)}$. Note that $\boldsymbol{\epsilon}_i$ is the same r.v. across all $\Phi^{(\eta)}$.

We propose a Gaussian approximation to the distribution of $\boldsymbol{\Psi}$, i.e., $\hat{p}(\boldsymbol{\Psi}) = \mathcal{N}(\boldsymbol{\Psi}|\boldsymbol{\mu}, \boldsymbol{\Sigma})$, where $\boldsymbol{\mu} \in \mathbb{R}^P$ is the mean vector and $\boldsymbol{\Sigma} \in \mathbb{R}^{P\times P}$ is the covariance matrix. From the previous derivation, the entries in $\boldsymbol{\mu}$ are $\mu^{(\eta)} = \mathbb{E}[\Phi^{(\eta)}] = \sum_i \mu_i^{(\eta)}$, which can be computed with (5). The diagonal of the covariance matrix is $\boldsymbol{\Sigma}_{\eta,\eta} = \text{var}(\Phi^{(\eta)})$, which is computed from (6). The covariance terms are derived as (see Supp. for derivation),

$$\boldsymbol{\Sigma}_{\eta,\rho} = \text{cov}(\Phi^{(\eta)}, \Phi^{(\rho)}) = \sum_i \omega_i^{(\eta,\rho)} - \sum_i \mu_i^{(\eta)}\mu_i^{(\rho)}, \quad (8)$$

where

$$\begin{aligned}\omega_i^{(\eta,\rho)} &= \mathbb{E}[\phi_i^{(\eta)}\phi_i^{(\rho)}] \quad (9)\\ &= \mathcal{N}(\mathbf{x}^{(\eta)}|\mathbf{x}^{(\rho)}, 2\beta\mathbf{I})\mathcal{N}(\frac{1}{2}(\mathbf{q}_i^{(\eta)} + \mathbf{q}_i^{(\rho)})|\mathbf{0}, (\beta/2+\alpha)\mathbf{I}).\end{aligned}$$

Fig. 2 (d) shows an example of the joint distribution $\boldsymbol{\Psi}$ and its Gaussian approximation for two spatial locations, while Fig. 4 (top) shows an example on a small image.

Given the $(\boldsymbol{\mu}, \boldsymbol{\Sigma})$ calculated from the annotations for an image $\mathcal{I}$, the negative log-likelihood function is used as the loss function for supervising the prediction $\boldsymbol{\Psi} = f(\mathcal{I})$ of the density-map regressor,

$$\begin{aligned}\mathcal{L} &= -\log p(\boldsymbol{\Psi}) = -\log \mathcal{N}(\boldsymbol{\Psi}|\boldsymbol{\mu}, \boldsymbol{\Psi}) \quad (10)\\ &\propto ||\boldsymbol{\Psi} - \boldsymbol{\mu}||^2_{\boldsymbol{\Sigma}} = (\boldsymbol{\Psi} - \boldsymbol{\mu})^T\boldsymbol{\Sigma}^{-1}(\boldsymbol{\Psi} - \boldsymbol{\mu}). \quad (11)\end{aligned}$$

Note that $\boldsymbol{\mu}$ has entries from (5), which is equivalent to a density map generated with squared-bandwidth $\alpha + \beta$, i.e., sum of the spatial noise variance and the original squared-bandwidth. Hence the loss in (11) is a generalization of the standard L2 (MSE) loss, except now the correlation between pixel locations is considered

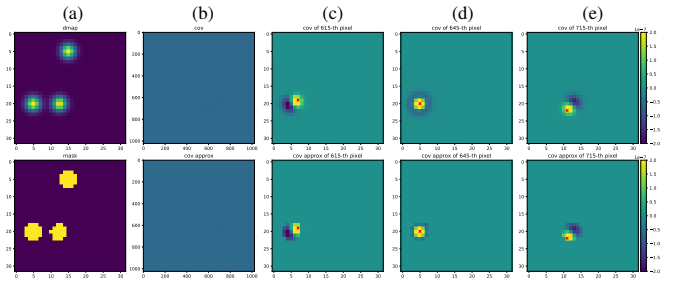

Fig. 4. (top) Example of Gaussian approximation to $\boldsymbol{\Psi}$: (a) mean vector (reshaped to 32×32); (b) covariance matrix; (c-e) covariance map for the spatial locations (marked as red x), corresponding to one row in the covariance matrix in (b) reshaped to 32×32. (bottom) low-rank approximation using the indices in the mask (bottom-a).

via the covariance matrix $\boldsymbol{\Sigma}$. As seen in (8) and (9), this correlation still exists, even if there is no spatial noise, i.e., $\alpha = 0$, since the pixels are correlated through the spatial convolution operation.

3.2.4 *Low-rank approximation to covariance matrix*

The covariance matrix $\boldsymbol{\Sigma} \in \mathbb{R}^{P\times P}$ does not scale well in computation and storage for large images. However, in $\boldsymbol{\Sigma}$, most of the off-diagonal elements in a column or row are close to zero if that spatial location is far from annotations. Therefore, $\boldsymbol{\Sigma}$ can be approximated by those rows/columns with significant covariance values (see full derivation in Supp.).

Let $\mathcal{M} = \{m_1, \cdots, m_M\}$ be the set of indices of spatial locations $\mathbf{x}^{(m_i)}$ that we select for the approximate covariance matrix. The approximation to $\boldsymbol{\Sigma}$ only uses the off-diagonal elements corresponding to $\mathcal{M}$,

$$\boldsymbol{\Sigma} \approx \hat{\boldsymbol{\Sigma}} = \mathbf{V} + \mathbf{M}\mathbf{A}_{\mathcal{M}}\mathbf{M}^T, \quad (12)$$

where $\mathbf{V} = \text{diag}(\text{diag}(\boldsymbol{\Sigma}))$ is the diagonal matrix of the diagonal of $\boldsymbol{\Sigma}$, $\mathbf{M}$ is a permutation matrix with i-th column $[\mathbf{M}]_i = \mathbf{e}_{m_i}$, and the selected off-diagonal entries are

$$[\mathbf{A}_{\mathcal{M}}]_{ij} = \begin{cases} 0, & i = j,\\ \text{cov}(\Phi^{(m_i)}, \Phi^{(m_j)}), & i \neq j.\end{cases} \quad (13)$$

Using the matrix inversion lemma, we obtain the approximate inverse covariance matrix,

$$\hat{\boldsymbol{\Sigma}}^{-1} = \mathbf{V}^{-1} - \mathbf{M}\mathbf{B}_{\mathcal{M}}\mathbf{M}^T, \quad (14)$$

$$\mathbf{B}_{\mathcal{M}} = (\mathbf{V}_{\mathcal{M}}\mathbf{A}_{\mathcal{M}}^{-1}\mathbf{V}_{\mathcal{M}} + \mathbf{V}_{\mathcal{M}})^{-1}, \quad \mathbf{V}_{\mathcal{M}} = \mathbf{M}^T\mathbf{V}\mathbf{M}. \quad (15)$$

Finally, the approximate loss function is the negative log-likelihood function using the approximate covariance,

$$\begin{aligned}\hat{\mathcal{L}} &= -\log \hat{p}(\boldsymbol{\Psi}) = -\log \mathcal{N}(\boldsymbol{\Psi}|\boldsymbol{\mu}, \hat{\boldsymbol{\Sigma}}) \propto ||\boldsymbol{\Psi} - \boldsymbol{\mu}||^2_{\hat{\boldsymbol{\Sigma}}}\\ &= \bar{\boldsymbol{\Psi}}^T\mathbf{V}^{-1}\bar{\boldsymbol{\Psi}} - \bar{\boldsymbol{\Psi}}^T\mathbf{M}\mathbf{B}_{\mathcal{M}}\mathbf{M}^T\bar{\boldsymbol{\Psi}}, \quad (16)\end{aligned}$$

where $\bar{\boldsymbol{\Psi}} = \boldsymbol{\Psi} - \boldsymbol{\mu}$. Since $\mathbf{V}$ is diagonal, the first term in (16) is equivalent to the sum over the negative log-marginals of Φ (i.e., a diagonal covariance matrix). The second term is the correlation term, based on the M selected entries $\mathbf{M}^T\bar{\boldsymbol{\Psi}}$. The storage/computational complexity for one training example using the low-rank approximation is $O(M^2 + N)$ compared to $O(N^2)$ for the full covariance.

3.2.5 Point-wise regularizer

We further use a point-wise regularization, inspired by Ma et al. [73], to encourage that the predicted density map near to each annotation will sum to 1. For the i-th annotation point, we first define the *point-wise density* $\hat{\gamma}_i$ as the total density in the predicted density map that is "assigned" to the i-th annotation. Specifically, $\hat{\gamma}_i$ is the weighted sum over all density values weighted by the posterior probability that the location $\mathbf{x}^{(\eta)}$ is "assigned" to the i-th annotation,

$$\hat{\gamma}_i = \sum_{\eta} \Psi^{(\eta)} \frac{\mathcal{N}(\mathbf{x}^{(\eta)}|\tilde{\mathbf{D}}_i, \beta\mathbf{I})}{\sum_{k=1}^{N} \mathcal{N}(\mathbf{x}^{(\eta)}|\tilde{\mathbf{D}}_k, \beta\mathbf{I})}, \tag{17}$$

where $\Psi^{(\eta)}$ is the η-th entry of the predicted density map $\mathbf{\Psi}$. The total density for each annotation should be 1, and thus the point-wise regularizer is defined as $\mathcal{L}_i^r = |\hat{\gamma}_i - 1|$.

However, this regularizer assumes there are no missing-point or duplicate-point noise in annotations. Hence, we further improve the regularization by also explicitly considering noise caused by missing and duplicate annotations.

3.3 Missing and Duplicate Noise

In this section, we consider the missing and duplicate annotations by assuming the density of each annotation as a random variable instead of one. Our model is based on two assumptions about crowd scenes: 1) *missing noise*: since the missing annotated person is usually partially occluded by an existing person next to it, we assume that the missing annotation will appear next to an existing annotation; 2) *duplicate noise*: each annotation can possibly be a duplicate annotation. Under these assumptions, we derive the distribution of each point-wise prediction γ_i, which is generated from density map according to (17). Finally, the negative log likelihood is used as the point-wise loss function.

3.3.1 Probability distribution of γ_i

We assume that each annotation can be a duplicate annotation, and that one person close to it is potentially missing. If the j-th annotation is a duplicate annotation and no density is predicted over its location, the point-wise density of nearby annotation γ_i will decrease since part of its density will be assigned to this duplicate j-th annotation when computing (17). Similarly, if there is a missing annotation near to the j-th annotation and density is predicted there, then the point-wise density γ_i of the nearby i-th annotation will increase since the extra predicted density will be assigned to it. The visualization is shown in Fig. 5.

Therefore, we define the i-th point-wise density γ_i as:

$$\gamma_i = 1 + \sum_{j=1}^{N} d_{ij} = 1 + \sum_{j=1}^{N} \frac{p_{ij}}{\sum_{k=1}^{N} p_{ik}} \tilde{\gamma}_j, \tag{18}$$

where d_{ij} is the density fluctuation of the i-th annotation caused by missing or duplicate annotations of the j-th annotation. We define $d_{ij} = \frac{p_{ij}}{\sum_{k=1}^{N} p_{ik}} \tilde{\gamma}_j$, where $p_{ij} = \mathcal{N}(\tilde{\mathbf{D}}_j|\tilde{\mathbf{D}}_i, \beta\mathbf{I})$ is the association between the i-th and j-th annotations. $\tilde{\gamma}_j$ is an indicator random variable, which takes a value in $\{1, 0, -1\}$, indicating that the j-th annotation has a nearby missing annotation, is a correct annotation, or duplicate annotation, respectively. The probability distribution of $\tilde{\gamma}_j$ is

$$p(\tilde{\gamma}_j) = \begin{cases} \tilde{\pi}, & \tilde{\gamma}_j = 1, \\ 1 - 2\tilde{\pi}, & \tilde{\gamma}_j = 0, \\ \tilde{\pi}, & \tilde{\gamma}_j = -1. \end{cases} \tag{19}$$

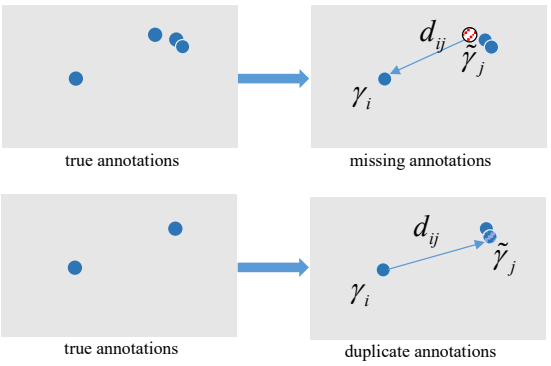

Fig. 5. The illustration of (top) missing and (bottom) duplicate annotations. The annotation points are shown in blue dots. (top) density predictions over a missing annotation will be assigned to other annotations, causing their point-wise density γ_i to increase. (bottom) a duplicate annotation will have density assigned to it, which causes the point-wise density γ_i of other annotations to decrease.

Here $\tilde{\pi}$ is the probability of a missing or duplicate annotation, which is also used as the missing or duplicate noise level in the experiments.

3.3.2 Approximation to γ_i

To understand the effect of our missing/duplicate annotation noise model, we first use sampling to analyze the distribution of γ. The distribution could be well modeled by a Laplace distribution as shown in Fig. 6. The mean of γ_i is

$$a_i = \mathbb{E}[\gamma_i] = 1 + \sum_{j=1}^{N} \frac{p_{ij}}{\sum_{k=1}^{N} p_{ik}} \mathbb{E}[\tilde{\gamma}_j] = 1, \tag{20}$$

since $\mathbb{E}[\tilde{\gamma}_j] = 0$, and its variance (see Supp. for derivation) is

$$\sigma_i^2 = \mathrm{var}(\gamma_i) = \sum_{j=1}^{N} \Big(\frac{p_{ij}}{\sum_{k=1}^{N} p_{ik}} \Big)^2 2\tilde{\pi}. \tag{21}$$

Note that we assume that annotations are independent with each other. Using these statistics, the parameters of the Laplace distribution are the mean a_i and the diversity $b_i = \frac{\sigma_i}{\sqrt{2}}$. Finally, the negative log likelihood of γ_i is used as the loss function for the point-wise densities $\hat{\gamma}_i$ of the predicted density map,

$$\mathcal{L}_i^r = -\log p(\hat{\gamma}_i|a_i, b_i) \propto \frac{|\hat{\gamma}_i - a_i|}{b_i}. \tag{22}$$

3.4 Empirical Approximation

One issue of using the approximate point distribution in (22) is that the variance of points in low-density regions is close to 0, which results in an unstable loss. A common practice is to add a small value ξ to the variance for computational stability. However, there still exists other problems. First, the computation of variance and diversity in (16) and (22) is time-consuming. Second, the weight in the background region is still too large (even when conditioned with ξ), which hinders the learning of high density regions. To address these issues, we propose an empirical loss function to directly map the nearest neighbor distance to the normalized pixel-wise and point-wise weights. In particular, we define the pixel and point-wise loss functions as:

$$\mathcal{L}_{pixel} = \mathbf{w}_1^T (\mathbf{\Psi} - \boldsymbol{\mu})^2, \tag{23}$$

$$\mathcal{L}_{point} = \mathbf{w}_2^T |\hat{\boldsymbol{\gamma}} - \mathbf{a}|, \tag{24}$$

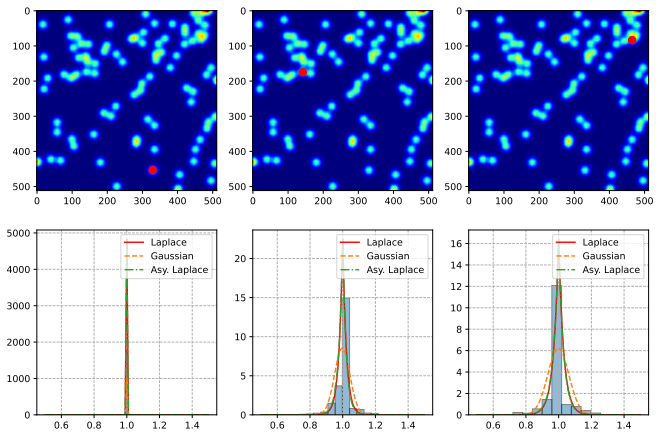

Fig. 6. Example of probability distribution of point-wise density with missing/duplicate annotations. (top) three evaluated points that are marked as red. (bottom) the distribution of point-wise density of the corresponding points. The histogram is obtained by sampling and the curves are the approximations. Here $\tilde{\pi} = 0.1$.

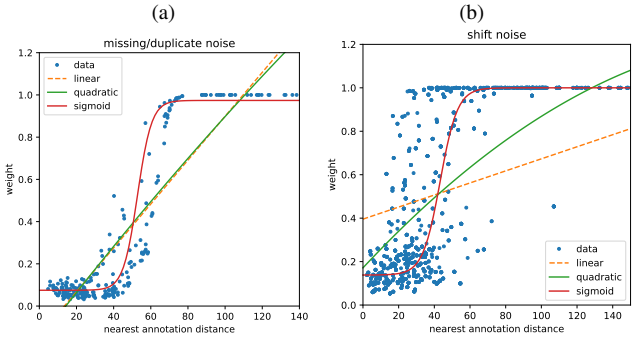

Fig. 7. The point and pixel weight v.s. nearest neighbor distance. The points are computed by sampling method.

where the element-wise square and absolute value are applied. The vectors $\mathbf{w}_1$ and $\mathbf{w}_2$ are the pixel-wise and point-wise weights, for approximating the inverse variance (precision) in (16) and inverse diversity in (22). Note that the covariance terms in (16) are not considered here.

For efficiency, we learn a function to directly map the nearest neighbor distance to the pixel-wise and point-wise weights $(\mathbf{w}_1, \mathbf{w}_2)$. In particular, we first compute the pixel's average nearest distance to the annotations and the corresponding weight (the inverse variance or inverse diversity) using training samples. Then, a function is used to approximate the relationship between the nearest neighbor distance and the weight. We use a sigmoid function since it fits the sampling result better as shown in Fig. 7. In Fig 8, we visualize the weight functions learned with different noise levels. As the noise level increases, the learned function becomes sharper – more annotations will be assigned low weights since the uncertainty of the dataset is increased.

3.4.1 Total count loss

We further include an epsilon-insensitve loss for total counting error, which takes the missing and duplicate annotations into consideration. In particular, the total loss is 0 if the error is within the assumed annotation noise,

$$\mathcal{L}_{count} = \max(0, |\text{sum}(\mathbf{\Psi}) - N| - N\tilde{\pi}), \tag{25}$$

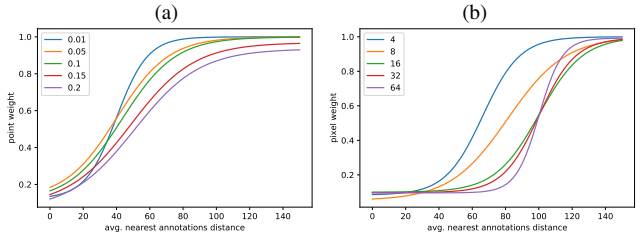

Fig. 8. The relationship between average distance to nearest annotations and the (a) point and (b) pixel weights for different noise levels.

where N is the ground-truth number of people.

The final loss function is the combination of pixel, point, and count loss,

$$\mathcal{L} = \lambda_1 \mathcal{L}_{pixel} + \lambda_2 \mathcal{L}_{point} + \mathcal{L}_{count}, \tag{26}$$

where λ_1, λ_2 are balancing hyperparameters.

4 Experiments

In this section, we evaluate the performance of the proposed method for different noise levels on three tasks: crowd counting, visual tracking, and human pose estimation. In crowd counting, shift noise, missing noise, and duplicate noise are considered. In visual tracking and human pose estimation, we only consider shift noise since missing and duplicate noise are rarely occur in their annotations.

4.1 Crowd Counting

We first consider the task of crowd counting, where the annotations may contain shift, missing, or duplicate noise.

4.1.1 Experiment Settings

Datasets: We use ShanghaiTech [5], UCF-QNRF [74], JHU-CROWD++ [75], and NWPU-Crowd [76] as the datasets for crowd counting. ShanghaiTech A contains 482 training and 300 testing images, and ShanghaiTech B has 716 and 400 training and testing images. UCF-QNRF comprises 1,535 high-resolution images (1201/334 for training and testing). JHU-CROWD++ is a large-scale dataset that contains 4,371 images (2,722, 500, and 1,600 images for training, validation, and testing). NWPU-Crowd is a large-scale benchmark that has 3,109 training images, 500 validation images and 1,500 testing images. Note that the annotations for the testing images for NWPU-Crowd are not released for fairer comparison.

Metric: The Mean Absolute Error (MAE) and rooted Mean Squared Error (MSE) are used as the metric:

$$MAE = \frac{1}{N}\sum_i |\hat{y}_i - y_i|, MSE = \sqrt{\frac{1}{N}\sum_i (\hat{y}_i - y_i)^2}, \tag{27}$$

where $\hat{y}_i$ and y_i are the predicted and ground-truth counts, and N is the number of images.

Training: Three counting networks are tested: VGG19 [73], CSRNet [77], and MCNN [5]. VGG19 and CSRNet are pre-trained on ImageNet, while MCNN is trained from scratch. The implementations follow their respective papers, and we replace the loss function for training. We train with two versions of our proposed loss: only with shift noise [1], which is denoted as "Ours (shift)"; with shift, missing & duplicate noise, denoted as "Ours

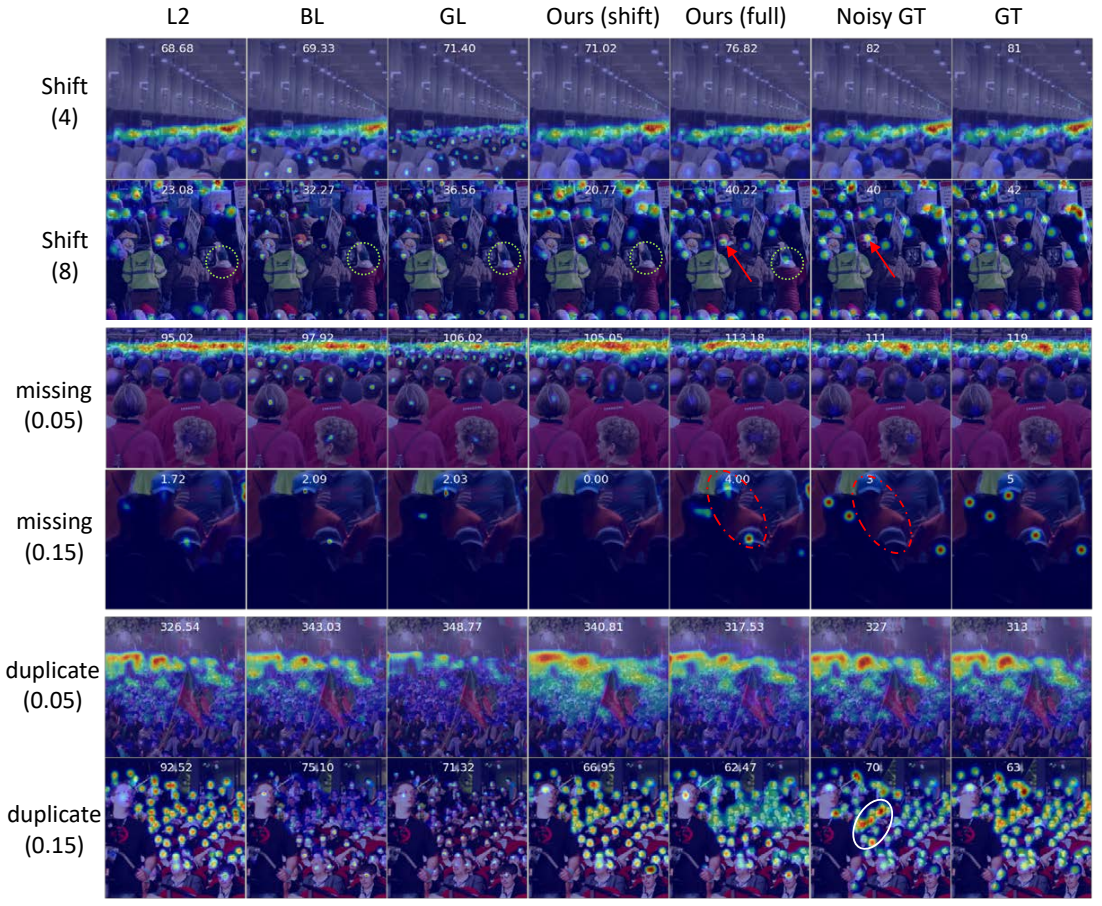

Fig. 9. Comparison of density maps learned with different loss functions under different noises. **(top)** The first two rows show different shift noises. First, the proposed loss function can correct the shift noise as shown in red arrows. Second, GL and BL totally miss the person in dotted circle since the prediction is over-confident. **(middle)** The middle two rows show missing annotation and the noisy GT count is less than GT count. First, the predicted counts (the white number at the top of density maps) of the proposed method are between noisy GT and the real GT while the others are inaccurate. Second, two persons are miss annotated while the proposed method can detect them as shown in red dotted ellipse. **(bottom)**The last two rows are duplicate annotations and the noisy GT count is greater than the real count. Typical duplicate annotations are shown in white ellipse. Similarly, predictions of the proposed method are close to the real GT while the other losses tend to over-predict the counts.

(full)". For comparison, we also train with standard L2 loss (i.e., MSE), Bayesian Loss [73] (BL), and the generalized loss [21] (GeneralizedLoss, GL).

The network is trained with an Adam optimizer [78] with 1e-5 learning rate. The weight decay is 1e-5 and all experiments trained for 500 epochs except for MCNN. Since MCNN is trained from scratch, we use larger learning rate (1e-4) and more training epochs (1000). The shift noise α and missing/annotation noise $\tilde{\pi}$ is set to 8 and 0.05 respectively. We set $\beta = 12$, $\lambda_1 = 1$, and $\lambda_2 = 1$ according to the ablation experiment shown in Fig. 11. A larger β works better because a sharp density map (small β) will enhance the effect of annotation noise. Small λ_1 and λ_2 do not work well since the pixel-wise and point-wise supervisions provide important information about the density arrangement. In addition, large λ_1 and λ_2 do not work well since the total count loss is needed to ensure the total count is accurate.

4.1.2 Robustness to shift noise

We first evaluate the robustness to shift noise for different loss functions on UCF-QNRF [74]. The noisy datasets are generated by randomly moving annotated locations by $\{4, 8, 16, 32, 64\}$ pixels. Note that the average head size is around 33 pixels, so the larger noises correspond to moving the annotation completely off the head. Next, we train the counting network with different loss functions on the noisy datasets. The performance is shown in Fig. 10 (a). First, the performance of all loss functions decreases dramatically with the increase of the noise level, which shows the impact of annotation shift noise. Second, the proposed loss function is more robust to annotation noise, especially for large noise levels. Finally, the density maps generated by networks trained with different loss functions are shown in Fig. 9 (top two rows). The proposed loss function can correct the shift noise. For comparison, GL and BL may totally miss a person since the prediction is over-confident, while our prediction is more accurate.

Finally, we note that GL has better performance than Ours (shift), the shift-only version of our loss, while worse performance than our full model. Note that GL can handle shift noise, duplicate annotations, and missing annotations, since its unbalanced optimal transport framework can ignore or hallucinate some annotations using its point-wise and pixel-wise loss. In contrast, Ours (shift) only handles the shift noise. Thus, GL might have advantage over Ours (shift) since there is some inherent such missing/duplicate noise in the dataset. However, once we additionally model the duplicate/missing noise, our method performs better.

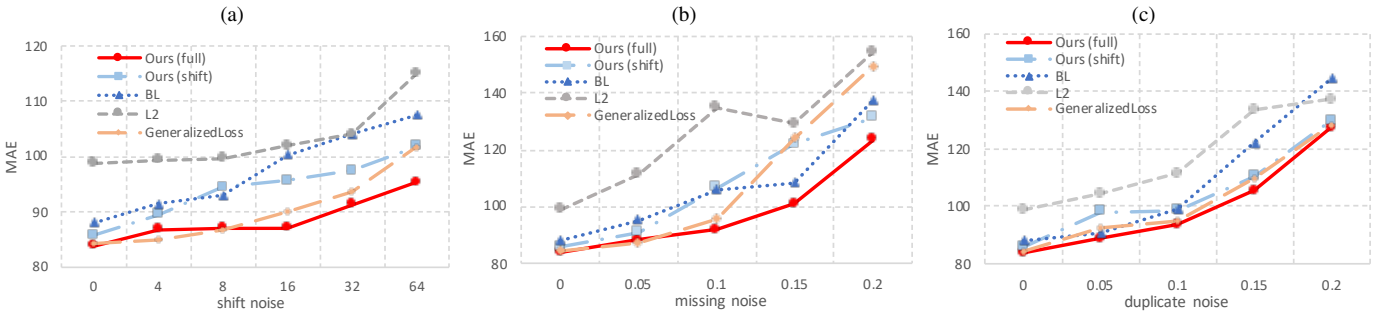

Fig. 10. Experiment results: robustness to (a) shift noise, (b) missing noise, and (c) duplicate noise.

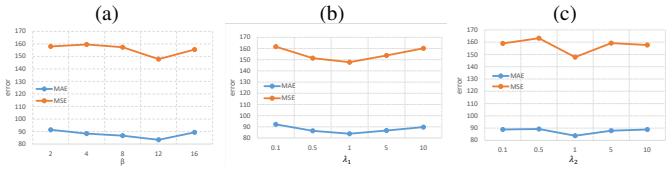

Fig. 11. Ablation study: effect of β, λ_1 and λ_2.

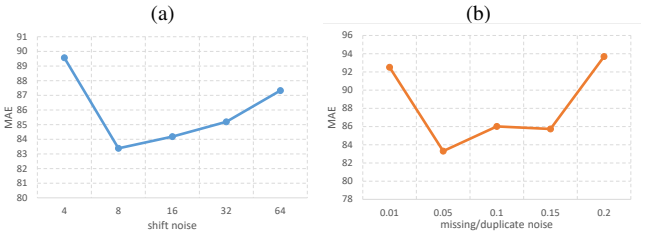

Fig. 12. Ablation study: Effect of shift noise and missing/duplicate noise parameters, $\sqrt{\alpha}$ and $\tilde{\pi}$, on the original UCF-QNRF.

TABLE 1
Effect of terms in the loss function on UCF-QNRF.

Component	Combinations			
pixel loss		✓	✓	✓
point loss	✓		✓	✓
count loss	✓	✓		✓
MAE	87.6	89.0	89.4	**83.8**

TABLE 2
Ablation study on the modeling of different noise types on UCF-QNRF. Std is the standard deviation over 5 trials.

	baseline	baseline + shift	baseline + miss/duplicate	Ours
MAE	91.8	90.9	91.4	**87.8**
std	2.3	1.1	3.4	2.6

4.1.3 Robustness to missing/duplicate noise

We next evaluate the robustness of different loss functions to missing/duplicate noise. To generate missing and duplicate noise, we randomly remove the annotation point or add additional points close to the current annotation with a probability $\tilde{\pi} \in \{0.01, 0.05, 0.1, 0.15, 0.2\}$. The experimental results are shown in Figs. 10 (b) and (c). First, the performance of the proposed method is limited if we only model the shift noise (see "Ours (shift)" in Fig. 10 (b)). Second, our full model with missing noise considered is more robust to missing noise compared to other loss functions. Third, BL handles missing noise better than GL, whereas GL handles duplicate noise better than BL, and thus they are only good at one type of missing/duplicate noise. In contrast our full model is consistently good on both types of noise. Finally, the improvement of modeling the duplicate noise is limited compared to modeling the shift noise only as shown in Fig. 10 (c). We believe the reason is that the duplicate annotation is more likely to appear in high-density regions and the pixel weight at those regions is low according to the modeling of shift noise. Therefore, the modeling of shift noise is also useful to address duplicate noise. As visualized in Fig. 9, the comparison methods tend to under-estimate or over-estimate the counts because of the noisy GT, while the proposed loss is more robust to the missing and duplicate noises.

4.1.4 Ablation studies

We next conduct a series of ablation studies to study the effect of various components.

Effect of loss components. We first investigate the effectiveness of different components in the proposed loss function on UCF-QNRF [74]. As shown in Tab. 1, the count loss is the most important as the performance drops significantly if we remove this component. Since the uncertain regions are trained with low weights, the count loss is required to ensure the total count prediction is accurate. In addition, the point-wise loss is also useful for counting since it ensures the prediction around the head region sums to 1. The pixel loss is less important since the spatial arrangement of density affect less the total count.

Effect of shift noise and missing/duplicate noise parameters. Since the noise level is unknown, we conduct an experiment on the original UCF-QNRF dataset with different assumed noise level parameters. In particular, we first compute the weight mapping function based on different shift noise and missing/duplicate noise parameters as shown in Fig. 8. Then, different functions are used to compute the pixel and point weights for comparison, and the experiment result is shown in Fig. 12. We find that the performance is limited if the assumed noise level is too small, which confirms the original dataset is noisy. Specifically, from the MAE results, we may infer that the shift noise in the dataset is about 8 pixels, and the probability of duplicate/missing annotations is around 0.05.

Effect of different noise types. We conduct an experiment to verify the effectiveness of modeling shift noise and missing/duplicate noise on UCF-QNRF. Note that the missing noise and duplicate noise are modeled together, and all experiments are repeated for 5 times. The results are presented in Table 2. First, the performance is improved with either shift or missing/duplicate noise modeled, which demonstrate the effectiveness of noise mod-

TABLE 3
Experiment results for mixed shift noise levels on UCF-QNRF.

	L2	BL	GL	Ours (shift)	Ours (miss/duplicate)	Ours (full)
MAE	177.0	110.5	91.8	90.9	92.5	**88.5**
MSE	251.5	191.9	162.8	159.1	160.9	**157.0**

TABLE 4
Experiment results using different counting models on UCF-QNRF.

	VGG19		CSRNet		MCNN	
	MAE	MSE	MAE	MSE	MAE	MSE
L2	98.7	176.1	110.6	190.1	186.4	283.6
BL	88.8	154.8	96.5	163.3	177.4	259.0
GL	84.3	**147.5**	92.0	165.7	142.8	227.9
Ours	**83.8**	147.8	**87.8**	**156.8**	**134.3**	**223.2**

TABLE 5
Comparison of our method with other loss functions on UCF-QNRF over 5 trials.

	L2	BL	GL	Ours
MAE	191.4	92.9	90.1	**87.8**
std	21.5	5.0	4.3	**2.6**

eling. Second, the improvement of shift noise modeling is more significant compared to the missing/duplicate noise modeling, which shows that the shift noise has more impact compared to missing/duplicate noise. Finally, the largest improvement comes from modeling all three types of noise together (ours).

Mixed levels of shift noise To better simulate a real situation, we conduct an experiment based on mixed levels of shift noise by randomly selecting a noise level from 0-64 pixels during generation. The results are shown in Tab. 3. First, the performance of Ours(shift) is better than comparison methods, which confirms the effectiveness of shift noise modeling. Second, our full model is even better than Ours(shift) since the missing/duplicate noise still exists in the dataset.

Comparison of counting models To evaluate the effectiveness of the proposed loss function, we compare it with L2, BL, and GL for training different counting models. The results are shown in Tab. 4. The model trained with the proposed loss is generally better than other loss functions, for a variety of models. This demonstrates that the modeling of noise is an important factor.

We further compare the proposed loss with other loss functions over 5 repeated trials on UCF-QNRF, and the experiment results are shown in Tab. 5. The proposed loss achieves the best MAE and the standard deviation is smaller, which demonstrates the effectiveness of our loss.

To further improve the performance, we apply the proposed loss function to the recently proposed Transformer-based model MAN [26], and the results are shown in Table 8 as "Ours (full) + MAN". Our loss function improves the performance of MAN, outperforming the traditional VGG19-based model, on the three large-scale datasets. Thus, our loss is applicable to state-of-the-art transformer-based models. Note that, the patch size is set to 2048 in our reproduction of MAN.

Combining with GL We next consider an experiment combining our proposed method with generalized loss. In particular, we directly change the point- and pixel-wise losses into weighted pixel- and point-wise losses defined in (23) and (24). As shown in Tab. 6, the performance can be further improved by incorporating our weights with the generalized loss.

TABLE 6
Incorporating noise modeling with Generalized Loss (GL) on UCF-QNRF.

	GL	Ours	Ours + GL
MAE	84.3	83.8	**81.4**
MSE	147.5	147.8	**143.7**

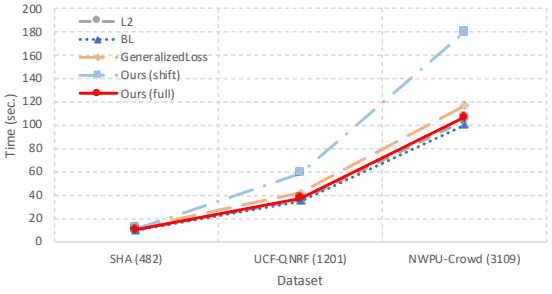

Fig. 13. Comparison of training time per epoch for different losses and datasets. The number in parentheses is the number of training images.

Training speed To demonstrate the efficiency of the empirical approximation, we compare the training time of different loss functions. In particular, VGG19 is used as the backbone and the crop size is 512×512. The result is shown in Fig. 13. First, the parametric modeling ("Ours (shift)") is the most time-consuming because of the computation of the covariance matrix. Second, the computation of transport matrix in GL is also time-consuming, but can be sped up by Sinkhorn iterations. Therefore, the speed of GL is still faster than "Ours (shift)". Finally, our proposed method using the empirical approximation is as fast as L2 and BL which confirms its efficiency.

4.1.5 *Comparison with state-of-the-art methods*

Finally, we compare the VGG19 backbone trained with our loss function with state-of-the-art models. and the results are shown in Tab 7. First, by modeling the annotation noise, our methods achieves better performance than BL, which uses the same backbone. It confirms that the modeling of annotation noise is effective. Second, the proposed method achieves the best MAE for most of the datasets including three largest datasets: NWPU-Crowd, JHU-CROWD++, and UCF-QNRF. We also compare with an uncertainly method [79] on UCF-QNRF and our method achieves better performance. Finally, the performance is improved when modeling the missing and duplicate noise, compared to modeling shift noise only (Ours full vs. Ours shift). DSSINet [80] is better than our method on the smaller ShanghaiTech A dataset since it uses multi-scale images to extract features. Similarly, MBTTBF [81] achieves better performance on ShanghaiTech B by fusing multi-level features together. However, those methods do not generalize to large-scale datasets. Note that we use VGG19 as the backbone, which does not do any special operation for multi-scale feature extraction.

Finally, we note that our MSE is worse than other methods because our model makes more mistakes on images with more than 5,000 people, as shown in Tab. 8. Since the proposed method assumes more noise in high-density regions and decreases the weight of those regions, the fitting of normal images is better than images with extreme counts (outliers). Furthermore, since we modeled the missing annotation noise, some challenging hard-negative images with fake crowd scenes will be over-counted.

TABLE 7
Comparison with state-of-the-art crowd counting methods.

	Venue	NWPU-Crowd		JHU-CROWD++		UCF-QNRF		ShanghaiTech A		ShanghaiTech B	
		MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE
Uncertainty [79]	NeurIPS'17	-	-	-	-	103.2	103.2	-	-	-	-
CP-CNN [13]	CVPR'17	-	-	-	-	-	-	73.6	106.4	20.1	30.1
ASACP [82]	CVPR'18	-	-	-	-	-	-	75.7	102.7	17.2	27.4
Switch-CNN [11]	CVPR'17	-	-	-	-	228.0	445.0	90.4	135.0	21.6	33.4
CMTL [83]	AVSBS'17	-	-	157.8	490.4	252.0	514.0	101.3	152.4	20.0	31.1
CL [74]	CVPR'18	-	-	-	-	132.0	191.0	-	-	-	-
LSCCNN [84]	TPAMI'20	-	-	112.7	454.4	120.5	218.2	66.5	101.8	7.7	12.7
MCNN [5]	CVPR'16	232.5	714.6	188.9	483.4	277.0	426.0	110.2	173.2	26.4	41.3
CSRNet [77]	CVPR'18	121.3	387.8	85.9	309.2	110.6	190.1	68.2	115.0	10.6	16.0
SANet [85]	ECCV'18	190.6	491.4	91.1	320.4	-	-	67.0	104.5	8.4	13.6
DSSINet [80]	ICCV'19	-	-	133.5	416.5	99.1	159.2	60.6	96.0	6.8	10.3
MBTTBF [81]	ICCV'19	-	-	81.8	299.1	97.5	165.2	60.2	94.1	8.0	15.5
BL [73]	ICCV'19	105.4	454.2	75.0	299.9	88.7	154.8	62.8	101.8	7.7	12.7
P2PNet [22]	ICCV'21	77.4	362.0	-	-	85.3	154.3	**52.7**	**85.1**	**6.3**	**9.9**
MAN [26]	CVPR'22	76.5	323.0	53.4	209.9	77.3	131.5	56.8	90.3	-	-
Ours (shift)		96.9	534.2	67.7	258.5	85.8	150.6	61.9	99.6	7.4	11.3
Ours (full)		87.9	444.5	59.1	259.6	83.8	147.8	61.8	104.3	7.1	12.4
Ours (full) + MAN		**75.3**	**313.1**	**53.0**	**208.6**	**75.3**	**128.3**	56.4	89.4	6.5	10.3

TABLE 8
The performance of different density levels on NWPU-Crowd test set. Note that the number of images is for the whole dataset for reference, since the distribution of the test set is unknown.

Crowd level	0		$(0, 100]$		$(100, 500]$		$(500, 5000]$		> 5000	
No. of images	351		1500		2371		889		52	
Metric	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE
MCNN	356.0	1232.5	72.1	151.0	103.5	154.7	509.5	768.3	4818.2	5179.5
SANet	432.0	974.4	65.0	169.8	104.2	180.8	385.1	580.6	2595.4	2974.2
CSRNet	176.0	572.3	35.8	69.1	59.7	106.3	285.8	448.3	**2055.8**	**2846.5**
BL	**66.5**	**258.4**	8.7	13.9	41.2	87.1	249.9	437.3	3386.4	3932.2
Ours	179.0	1056.1	**7.3**	**13.7**	**33.4**	**75.1**	**186.0**	**318.1**	2435.0	3075.1

For other scene types, our proposed method achieves the best performance for both MAE and MSE.

4.2 Visual Tracking

In this section, we apply our proposed loss to visual object tracking, where MSE loss is typically used between the predicted target response map and the ground-truth heat map.

4.2.1 Settings

Dataset: For the visual tracking task. we use the widely used OTB [37] tracking dataset to evaluate the performance of the proposed method. The OTB dataset contains 100 challenging video sequences with various attributes (e.g., occlusion, rotation, illumination variation, background cluster and fast motion), which can effectively demonstrate the effectiveness of our method. We evaluate the robustness of the proposed method by adding shift noise to the tracking dataset. Specifically, noisy training datasets are generated by randomly moving the center of the annotated bounding boxes by $\{20, 30, 40\}$ pixels in a random direction.

Metrics: Following OTB [37], we use the precision and success metrics for evaluation. The precision is the percentage of frames whose center errors with respect to the ground-truth are smaller than a predefined distance threshold. The success is defined as the percentage of frames whose overlap ratios with the ground truth bounding box are larger than an overlap threshold. The distance precision at a threshold (DPR) of 20 pixels and the area under curve (AUC) of the success plot are reported for comparison.

Training and Inference: We use DiMP18 [55] as our baseline tracker since DiMP18 uses a standard MSE loss as the target classification loss, which can be directly replaced with our proposed loss in (23) without further modifications. We use the GOT-10K [86] dataset to train both the DiMP18 baseline and our variant. For the scale estimation in the DiMP18 baseline and our variant, we use the pre-trained IOUNet [87] for scale estimation and do not update it during training in order to better study the effect of noisy annotation on target localization ability. We use the same training settings described in [55] for fair comparison.

During the online tracking stage, we use the same online updating strategies used in DiMP18 [55] for fair comparison. We also use the ground-truth bounding box in the first video frame to initialize our tracker, which is the same as DiMP18. In this way, the experiments will better demonstrate our method can learn a more robust offline tracking model even using noisy annotated tracking data.

4.2.2 Robustness to noise

Fig. 14 shows the tracker performance vs. the spatial noise level. Our method achieves better results than the baseline for different noise levels. When the noise level becomes larger, the improvements of our method is more significant, which indicates that our proposed loss can more effectively handle the noisy annotations, whereas standard MSE loss cannot. Another interesting phenomenon is that even when there is no added noise (i.e., noise=0), the performance of our method is still better than the baseline in terms of both the DPR and AUC metrics. This is mainly because the training dataset GOT-10K naturally contains

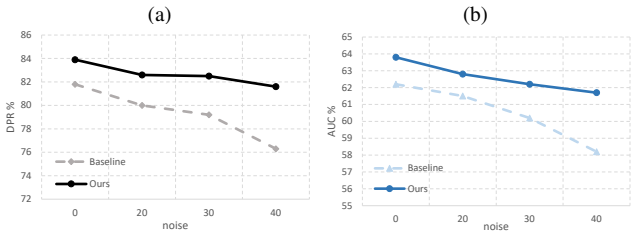

Fig. 14. Robustness to shift noise for visual tracking.

noisy annotations, and our method leads to more effective training on the original GOT-10K dataset.

We further visualize the classification maps and predicted bounding boxes (denoted as red boxes) with different noise levels in Fig. 15. The classification score of the target center location (denoted as red cross) is shown in the top-right of the video frame. The DiMP tracker offline-trained with our loss is more robust to handle online distractors and avoids some tracking failures, even with large added spatial noise levels. In addition, when the spatial annotation noise increases, our method can still accurately predict the target center location with high classification score (CS), e.g., when noise=40, cs=0.42.

4.3 Human Pose Estimation

In this section, we apply our proposed loss to human pose estimation (HPE), where L2 loss is typically used to supervise the model to learn from pose joint heat maps.

4.3.1 Settings

Dataset: MPII [88] and CrowdPose [89] are used to evaluate the performance of the proposed method on human pose estimation. MPII contains around 25k images with 40k people. To demonstrate the effectiveness of the proposed method on dense crowds, we further apply the method on CrowdPose, which is divided into 3 crowd levels based on the ratio of overlapped joints in an image [89]: easy (0-0.1), medium (0.1-0.8), and hard (0.8-1).

Metrics: Followed by previous works [88, 89], we evaluate using Percentage of Correct Keypoints (PCKh) based on head size for MPII, and Average Precision (AP) for CrowdPose.

Training: Our baseline networks are HRNet-W32 [3] for MPII dataset and HigherHRNet-W48^{+} [90] for CrowdPose. During training, we directly replace the MSE loss with our proposed loss in (23), and the remaining training details follow the baselines [3, 90]. Adam optimizer [78] is used for optimization and the base learning is set to 1e-3. The model is trained for 210 epochs and the learning rate is decreased by 10 at 170 and 200 epochs.

4.3.2 Robustness to noise

We now evaluate the robustness of the proposed loss to shift noise on human pose estimation. Similar to crowd counting, the noisy datasets are generated by randomly moving annotation points by $\{2, 4, 8, 16\}$ pixels. The experimental results are shown in Fig. 16. First, the performance of our loss function is almost the same as the traditional method. We believe the reason is that the annotations in HPE are less noisy compared to crowd counting, since the number of annotations is less and the occluded joints are usually ignored. Second, the improvement of the proposed method becomes larger as the increase of the noise level, which confirms the robustness of the method. Finally, the improvement becomes more significant when a more accurate localization criteria is used (10% of head size is used for calculating PCKh). This suggests that the localization of our method is more accurate than traditional L2 loss. As shown in Fig. 17, heat maps generated by our loss function are sharper than L2 loss, which further confirm that the proposed loss is more robust to shift noise.

4.3.3 Comparison with state-of-the-art methods

We evaluate the proposed method on crowded images in CrowdPose dataset, and compare it with state-of-the-art methods. As shown in Tab. 9, the proposed method achieves the best performance compared to other methods. In particular, the proposed method uses the same backbone network but is superior to the baseline method HigherHRNet [90]. Finally, the improvement on medium and crowded images is greater than on the easy images with less occlusion. This shows the potential of the proposed method to handle challenging noisy scenarios.

5 Conclusion

In this paper, we investigate three different types of noise in point-wise annotations: shift noise, missing-point noise, and duplicate-point noise. To model the more prevalent shift noise, we propose to model real locations as random variables and derive the distribution of the ground-truth map. To model the missing and duplicate noise in dense annotations, we further derive the distribution of the point-wise densities. Then, the negative log-likelihood is used as the loss function which is equivalent to a weighted L2/L1 loss. Finally, to accelerate the training process, we propose an empirical approximation to the weights in the loss function. We apply the proposed loss function to crowd counting, tracking and human pose estimation. Experimental results show that the proposed method is more robust to different types and levels of noises. The further work will focus on applying the noise-modeling principles to derive other robust loss functions for structured ground-truth annotations. For example, our robust loss framework could be applied to regressing bounding box annotations under noisy annotation conditions, e.g., in semi-supervised or self-supervised learning where pseudo-annotations are noisy (similar to [56]).

Acknowledgements. This work is supported by grants from the Research Grants Council of the Hong Kong Special Administrative Region, China (CityU 11212518 and CityU 11215820), and a Strategic Research Grant from City University of Hong Kong (Project No. 7005665).

References

[1] J. Wan and A. Chan, "Modeling noisy annotations for crowd counting," *Advances in Neural Information Processing Systems*, vol. 33, pp. 3386–3396, 2020.

[2] B. Wang, H. Liu, D. Samaras, and M. H. Nguyen, "Distribution matching for crowd counting," *Advances in Neural Information Processing Systems*, vol. 33, pp. 1595–1607, 2020.

[3] K. Sun, B. Xiao, D. Liu, and J. Wang, "Deep high-resolution representation learning for human pose estimation," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2019, pp. 5693–5703.

[4] B. Xiao, H. Wu, and Y. Wei, "Simple baselines for human pose estimation and tracking," in *European Conference on Computer Vision*, 2018, pp. 466–481.

[5] Y. Zhang, D. Zhou, S. Chen, S. Gao, and Y. Ma, "Single-image crowd counting via multi-column convolutional neural network," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2016, pp. 589–597.

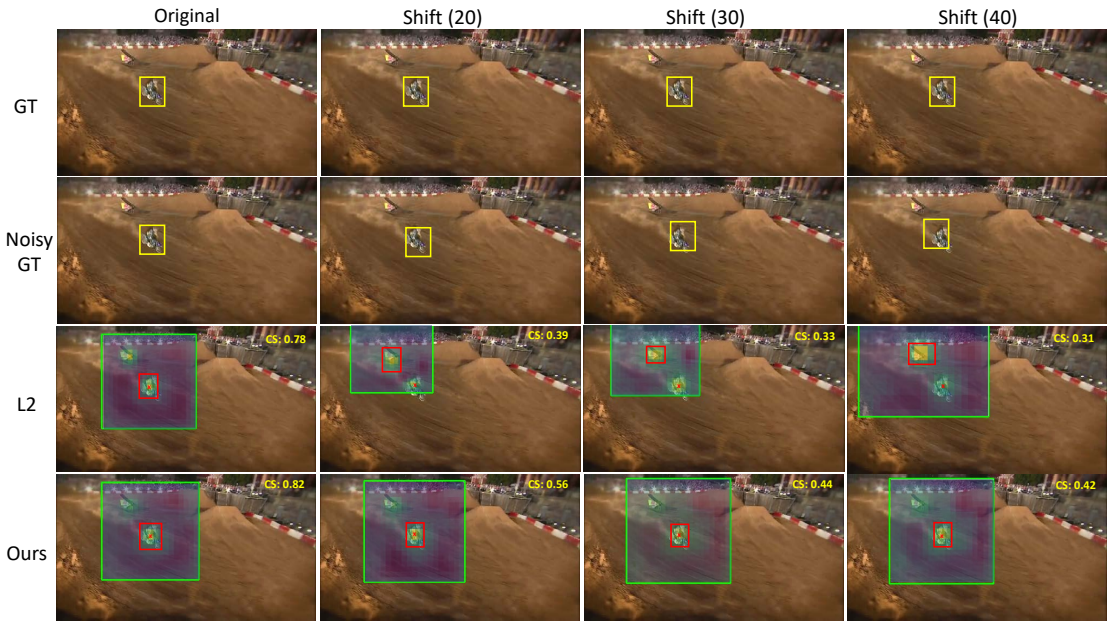

Fig. 15. Visualization of visual tracking results with different loss functions and noise levels. The tracker offline-trained with our loss is more robust to handle online distractors and avoid some tracking failures, even with large added spatial noises levels.

TABLE 9
Comparison with state-of-the-art human pose estimation methods on CrowdPose dataset. $^+$ means using multi-scale test.

	mAP@0.5:0.95	mAP@0.5	mAP@0.75	AP_{easy}	AP_{medium}	AP_{hard}
Mask R-CNN [91]	57.2	83.5	60.3	69.4	57.9	45.8
AlphaPose [33]	61.0	81.3	66.0	71.2	61.4	51.1
Xiao *et al.* [4]	60.8	81.4	65.7	71.4	61.2	51.2
CrowdPose [89]	66.0	84.2	71.5	75.5	66.3	57.4
HigherHRNet-W48$^+$ [90] (baseline)	67.6	87.4	72.6	75.8	68.1	58.9
Ours	**68.2**	**87.6**	**73.6**	**76.3** (↑0.5)	**68.8** (↑0.7)	**59.7** (↑0.8)

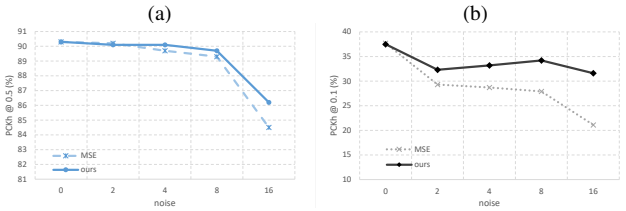

Fig. 16. Robustness to shift noise for human pose estimation on MPII dataset using (a) 50% and (b) 10% of the head segment length for matching.

[6] W. Ge and R. T. Collins, "Marked point processes for crowd counting," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2009, pp. 2913–2920.

[7] M. Li, Z. Zhang, K. Huang, and T. Tan, "Estimating the number of people in crowded scenes by mid based foreground segmentation and head-shoulder detection," in *International Conference on Pattern Recognition*, 2008, pp. 1–4.

[8] A. B. Chan, Z.-S. J. Liang, and N. Vasconcelos, "Privacy preserving crowd monitoring: Counting people without people models or tracking," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2008, pp. 1–7.

[9] A. B. Chan and N. Vasconcelos, "Bayesian poisson regression for crowd counting," in *International Conference on Computer Vision*, 2009, pp. 545–551.

[10] H. Idrees, I. Saleemi, C. Seibert, and M. Shah, "Multi-source multi-scale counting in extremely dense crowd images," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2013, pp. 2547–2554.

[11] D. B. Sam, S. Surya, and R. V. Babu, "Switching convolutional neural network for crowd counting," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2017, pp. 4031–4039.

[12] D. Kang and A. B. Chan, "Crowd counting by adaptively fusing predictions from an image pyramid," in *British Machine Vision Conference*, 2018, p. 89.

[13] V. A. Sindagi and V. M. Patel, "Generating high-quality crowd density maps using contextual pyramid cnns," in *International Conference on Computer Vision*, 2017, pp. 1879–1888.

[14] F. Xiong, X. Shi, and D. Y. Yeung, "Spatiotemporal modeling for crowd counting in videos," in *International Conference on Computer Vision*, 2017, pp. 5161–5169.

[15] C. Zhang, H. Li, X. Wang, and X. Yang, "Cross-scene crowd counting via deep convolutional neural networks," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2015, pp. 833–841.

[16] Q. Wang, J. Gao, W. Lin, and Y. Yuan, "Learning from synthetic data for crowd counting in the wild," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2019, pp. 8198–8207.

[17] J. Wan, W. Luo, B. Wu, A. B. Chan, and W. Liu, "Residual regression and semantic prior for crowd counting," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2019, pp. 4036–4045.

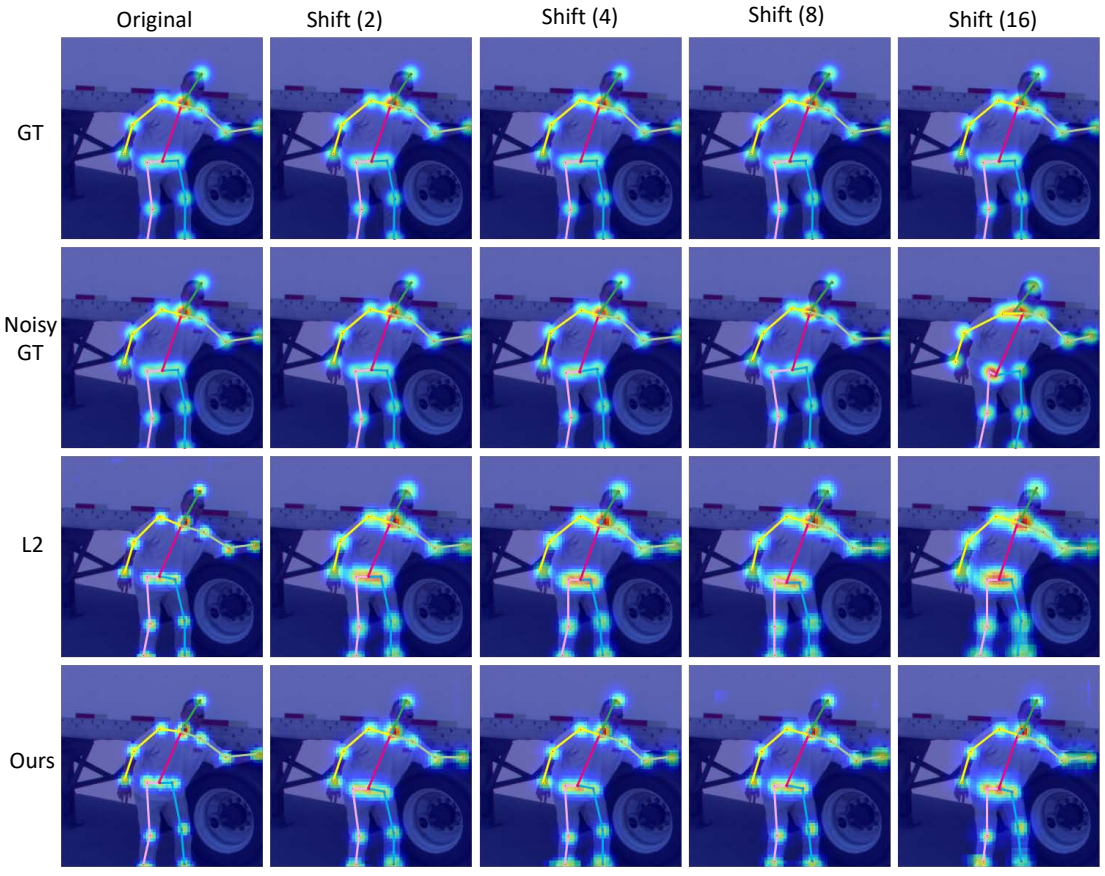

Fig. 17. Visualization of human pose estimation results with different loss functions and noise levels. In human pose estimation, we only assume shift noise since missing and duplicate noises are not likely to appear. As the noise level increases, density maps generated by traditional L2 loss become more blurry while our density maps are sharper.

[18] Q. Wu, J. Wan, and A. B. Chan, "Dynamic momentum adaptation for zero-shot cross-domain crowd counting," in *Proceedings of the ACM International Conference on Multimedia*, 2021, pp. 658–666.

[19] J. Wan and A. Chan, "Adaptive density map generation for crowd counting," in *International Conference on Computer Vision*, 2019, pp. 1130–1139.

[20] J. Wan, Q. Wang, and A. B. Chan, "Kernel-based density map generation for dense object counting," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 44, no. 3, pp. 1357–1370, 2020.

[21] J. Wan, Z. Liu, and A. B. Chan, "A generalized loss function for crowd counting and localization," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2021, pp. 1974–1983.

[22] Q. Song, C. Wang, Z. Jiang, Y. Wang, Y. Tai, C. Wang, J. Li, F. Huang, and Y. Wu, "Rethinking counting and localization in crowds: A purely point-based framework," in *International Conference on Computer Vision*, 2021, pp. 3365–3374.

[23] C. C. Loy, S. Gong, and T. Xiang, "From semi-supervised to transfer counting of crowds," in *International Conference on Computer Vision*, 2013, pp. 2256–2263.

[24] Y. Meng, H. Zhang, Y. Zhao, X. Yang, X. Qian, X. Huang, and Y. Zheng, "Spatial uncertainty-aware semi-supervised crowd counting," in *International Conference on Computer Vision*, 2021, pp. 15 549–15 559.

[25] Y. Xu, Z. Zhong, D. Lian, J. Li, Z. Li, X. Xu, and S. Gao, "Crowd counting with partial annotations in an image," in *International Conference on Computer Vision*, 2021, pp. 15 570–15 579.

[26] H. Lin, Z. Ma, R. Ji, Y. Wang, and X. Hong, "Boosting crowd counting via multifaceted attention," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022, pp. 19 628–19 637.

[27] D. Liang, W. Xu, and X. Bai, "An end-to-end transformer model for crowd localization," *European Conference on Computer Vision*, pp. 38–54, 2022.

[28] S. Yang, W. Guo, and Y. Ren, "Crowdformer: An overlap patching vision transformer for top-down crowd counting," in *Proceedings of the International Joint Conference on Artificial Intelligence*. International Joint Conferences on Artificial Intelligence Organization, 2022, pp. 1545–1551.

[29] W. Shu, J. Wan, K. C. Tan, S. Kwong, and A. B. Chan, "Crowd counting in the frequency domain," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022, pp. 19 618–19 627.

[30] W. Liu, N. Durasov, and P. Fua, "Leveraging self-supervision for cross-domain crowd counting," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022, pp. 5341–5352.

[31] Q. Zhang and A. B. Chan, "Calibration-free multi-view crowd counting," in *European Conference on Computer Vision*, 2022, pp. 227–244.

[32] A. Newell, K. Yang, and J. Deng, "Stacked hourglass networks for human pose estimation," in *European Conference on Computer Vision*, 2016, pp. 483–499.

[33] H. S. Fang, S. Xie, Y. W. Tai, and C. Lu, "Rmpe: Regional multi-person pose estimation," in *International Conference on Computer Vision*, 2017, pp. 2334–2343.

[34] Z. Cao, G. Hidalgo, T. Simon, S. E. Wei, and Y. Sheikh, "Openpose: realtime multi-person 2D pose estimation using part affinity fields," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 43, no. 1, pp. 172–186, 2019.

[35] L. Pishchulin, E. Insafutdinov, S. Tang, B. Andres, M. Andriluka, P. V. Gehler, and B. Schiele, "Deepcut: Joint subset partition and labeling for multi person pose estimation," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2016, pp. 4929–4937.

[36] Z. Geng, K. Sun, B. Xiao, Z. Zhang, and J. Wang, "Bottom-up human pose estimation via disentangled keypoint regression," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2021, pp. 14 676–14 686.

[37] Y. Wu, J. Lim, and M. H. Yang, "Object tracking benchmark," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 37, no. 9, pp. 1834–1848, 2015.

[38] H. Fan, L. Lin, and F. Yang, "Lasot: A high-quality benchmark for large-scale single object tracking," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2019, pp. 5374–5383.

[39] M. Muller, A. Bibi, and G. S, "Trackingnet: A large-scale dataset and benchmark for object tracking in the wild," in *European Conference on Computer Vision*, 2018, pp. 300–317.

[40] M. Danelljan, G. Hager, F. S. Khan, and M. Felsberg, "Convolutional features for correlation filter based visual tracking," in *International Conference on Computer Vision Workshops*, 2015, pp. 58–66.

[41] Y. Liang, Q. Wu, and Y. Liu, "Robust correlation filter tracking with shepherded instance-aware proposals," in *Proceedings of the ACM International Conference on Multimedia*, 2018, pp. 420–428.

[42] M. Danelljan, A. Robinson, F. S. Khan, and M. Felsberg, "Beyond correlation filters: learning continuous convolution operators for visual tracking," in *European Conference on Computer Vision*, 2016, pp. 472–488.

[43] M. Danelljan, G. Bhat, F. S. Khan, and M. Felsberg, "ECO: Efficient convolution operators for tracking," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2017, pp. 21–26.

[44] Q. Wu, Y. Yan, and Y. Liang, "DSNet: deep and shallow feature learning for efficient visual tracking," in *Asian Conference on Computer Vision*, 2018, pp. 119–134.

[45] J. Valmadre, L. Bertinetto, J. Henriques, A. Vedaldi, and P. H. S. Torr, "End-to-end representation learning for correlation filter based tracking," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2017, pp. 5000–5008.

[46] L. Bertinetto, J. Valmadre, J. Henriques, A. Vedaldi, and P. Vedaldi, "Fully-convolutional siamese networks for object tracking," in *European Conference on Computer Vision Workshops*, 2016, pp. 850–865.

[47] R. Tao, E. Gavves, and A. W. Smeulders, "Siamese instance search for tracking," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2016, pp. 1420–1429.

[48] Y. Xu and Z. W. adn Z. Li, "SiamFC++: Towards robust and accurate visual tracking with target estimation guidelines," in *Proceedings of the AAAI Conference on Artificial Intelligence*, 2020, pp. 12 549–12 556.

[49] B. Li, W. Wu, Z. Zhu, and J. Yan, "High performance visual tracking with siamese region proposal network," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2018, pp. 8971–8980.

[50] Z. Zhang and H. Peng, "Deeper and wider siamese networks for real-time visual tracking," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2017.

[51] Q. Wang, L. Zhang, and L. Bertinetto, "Fast online object tracking and segmentation: A unifying approach," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2019.

[52] Q. Guo, W. Feng, C. Zhou, R. Huang, L. Wan, and S. Wang, "Learning dynamic siamese network for visual object tracking," in *International Conference on Computer Vision*, 2017, pp. 1763–1771.

[53] T. Yang and A. B. Chan, "Learning dynamic memory networks for object tracking," in *European Conference on Computer Vision*, 2018, pp. 152–167.

[54] L. Zhang, A. G. Garcia, J. V. Weijer, M. Danelljan, F. Shahbaz, and F. S. Khan, "Learning the model update for siamese trackers," in *arXiv:1908.00855*, 2019.

[55] G. Bhat, M. Danelljan, L. V. Gool, and R. Timofte, "Leanring discriminative model prediction for tracking," in *International Conference on Computer Vision*, 2019, pp. 6182–6191.

[56] Q. Wu, W. Jia, and A. Chan, "Progressive unsupervised learning for visual object tracking," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2021, pp. 2993–3002.

[57] M. Danelljan, L. Gool, and R. Timofte, "Probabilistic regression for visual tracking," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2020, pp. 7183–7192.

[58] Y. Wang, X. Ma, Z. Chen, Y. Luo, J. Yi, and J. Bailey, "Symmetric cross entropy for robust learning with noisy labels," in *International Conference on Computer Vision*, 2019, pp. 322–330.

[59] T. Xiao, T. Xia, Y. Yang, C. Huang, and X. Wang, "Learning from massive noisy labeled data for image classification," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2015, pp. 2691–2699.

[60] D. T. Nguyen, T. P. Ngo, Z. Lou, M. Klar, L. Beggel, and T. Brox, "Robust learning under label noise with iterative noise-filtering," *arXiv preprint arXiv:1906.00216*, 2019.

[61] L. Yang, F. Meng, H. Li, Q. Wu, and Q. Cheng, "Learning with noisy class labels for instance segmentation," in *European Conference on Computer Vision*, 2020, pp. 38–53.

[62] N. Natarajan, I. S. Dhillon, P. K. Ravikumar, and A. Tewari, "Learning with noisy labels," *Advances in Neural Information Processing Systems*, vol. 26, 2013.

[63] Y. Wang, X. Sun, and Y. Fu, "Scalable penalized regression for noise detection in learning with noisy labels," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022, pp. 346–355.

[64] T. Kim, J. Ko, J. Choi, S. Yun *et al.*, "Fine samples for learning with noisy labels," *Advances in Neural Information Processing Systems*, vol. 34, pp. 24 137–24 149, 2021.

[65] Y. Bai, E. Yang, B. Han, Y. Yang, J. Li, Y. Mao, G. Niu, and T. Liu, "Understanding and improving early stopping for learning with noisy labels," *Advances in Neural Information Processing Systems*, vol. 34, pp. 24 392–24 403, 2021.

[66] K. J. Liang, S. B. Rangrej, V. Petrovic, and T. Hassner, "Few-shot learning with noisy labels," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022, pp. 9089–9098.

[67] D. Fu, D. Chen, H. Yang, J. Bao, L. Yuan, L. Zhang, H. Li, F. Wen, and D. Chen, "Large-scale pre-training for person re-identification with noisy labels," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022, pp. 2476–2486.

[68] S. Liu, K. Liu, W. Zhu, Y. Shen, and C. Fernandez Granda, "Adaptive early-learning correction for segmentation from noisy annotations," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022, pp. 2606–2616.

[69] S. Bai, Z. He, Y. Qiao, H. Hu, W. Wu, and J. Yan, "Adaptive dilated network with self-correction supervision for counting," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2020, pp. 4594–4603.

[70] N. Kato, T. Li, K. Nishino, and Y. Uchida, "Improving multi-person pose estimation using label correction," *arXiv preprint arXiv:1811.03331*, 2018.

[71] C. M. Bishop, *Pattern recognition and machine learning*. springer, 2006.

[72] R. D'AGOSTINO and E. S. Pearson, "Tests for departure from normality," *Biometrika*, vol. 60, no. 3, pp. 613–622, 1973.

[73] Z. Ma, X. Wei, X. Hong, and Y. Gong, "Bayesian loss for crowd count estimation with point supervision," in *International*

Conference on Computer Vision, 2019, pp. 6142–6151.
[74] H. Idrees, M. Tayyab, K. Athrey, D. Zhang, S. Al-Maadeed, N. Rajpoot, and M. Shah, "Composition loss for counting, density map estimation and localization in dense crowds," in *European Conference on Computer Vision*, 2018, pp. 532–546.
[75] V. A. Sindagi, R. Yasarla, and V. M. Patel, "JHU-CROWD++: Large-scale crowd counting dataset and a benchmark method," *Technical Report*, 2020.
[76] Q. Wang, J. Gao, W. Lin, and X. Li, "NWPU-crowd: A large-scale benchmark for crowd counting and localization," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 43, no. 6, pp. 2141–2149, 2020.
[77] Y. Li, X. Zhang, and D. Chen, "CSRNet: Dilated convolutional neural networks for understanding the highly congested scenes," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2018, pp. 1091–1100.
[78] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," *arXiv preprint arXiv:1412.6980*, 2014.
[79] A. Kendall and Y. Gal, "What uncertainties do we need in bayesian deep learning for computer vision?" in *Advances in Neural Information Processing Systems*, 2017, pp. 5574–5584.
[80] L. Liu, Z. Qiu, G. Li, S. Liu, W. Ouyang, and L. Lin, "Crowd counting with deep structured scale integration network," in *International Conference on Computer Vision*, 2019, pp. 1774–1783.
[81] V. A. Sindagi and V. M. Patel, "Multi-level bottom-top and top-bottom feature fusion for crowd counting," in *International Conference on Computer Vision*, 2019, pp. 1002–1012.
[82] Z. Shen, Y. Xu, B. Ni, M. Wang, J. Hu, and X. Yang, "Crowd counting via adversarial cross-scale consistency pursuit," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2018, pp. 5245–5254.
[83] V. A. Sindagi and V. M. Patel, "Cnn-based cascaded multi-task learning of high-level prior and density estimation for crowd counting," in *IEEE International Conference on Advanced Video and Signal Based Surveillance*, 2017, pp. 1–6.
[84] D. B. Sam, S. V. Peri, M. N. Sundararaman, A. Kamath, and V. B. Radhakrishnan, "Locate, size and count: Accurately resolving people in dense crowds via detection," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 43, no. 8, pp. 2739–2751, 2020.
[85] X. Cao, Z. Wang, Y. Zhao, and F. Su, "Scale aggregation network for accurate and efficient crowd counting," in *European Conference on Computer Vision*, 2018, pp. 734–750.
[86] L. Huang, X. Zhao, and K. Huang, "Got-10k: A large high-diversity benchmark for generic object tracking in the wild," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2019.
[87] M. Danelljan, G. Bhat, F. S. Khan, and M. Felsberg, "Atom: Accurate tracking by overlap maximization," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2019, pp. 4660–4669.
[88] M. Andriluka, L. Pishchulin, P. Gehler, and B. Schiele, "2D human pose estimation: New benchmark and state of the art analysis," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2014, pp. 3686–3693.
[89] J. Li, C. Wang, H. Zhu, Y. Mao, H. S. Fang, and C. Lu, "Crowdpose: Efficient crowded scenes pose estimation and a new benchmark," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2019, pp. 10 863–10 872.
[90] B. Cheng, B. Xiao, J. Wang, H. Shi, T. S. Huang, and L. Zhang, "Higherhrnet: Scale-aware representation learning for bottom-up human pose estimation," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2020, pp. 5386–5395.
[91] K. He, G. Gkioxari, P. Dollár, and R. Girshick, "Mask R-CNN," in *International Conference on Computer Vision*, 2017, pp. 2961–2969.

Jia Wan received the B.Eng. degree in software engineering from Northwestern Polytechnical University, Xi'an, China, and M.Phil. degree from School of Computer Science and the Center for OPTical IMagery Analysis and Learning (OPTIMAL), Northwestern Polytechnical University, Xi'an, China, in 2015 and 2018, and the Ph.D. degree in Computer Science from City University of Hong Kong, in 2021. He is currently a postdoctoral scholar in the Department of Electrical & Computer Engineering , University of California, San Diego (UCSD). His research interests include congestion analysis and crowd counting.

Qiangqiang Wu received the BS degree from the School of Information and Electronic Engineering, Zhejiang Gongshang University in 2016, and the MS degree in Computer Science from Xiamen University in 2019. He is working toward the Ph.D. degree with the Department of Computer Science in City University of Hong Kong, Hong Kong, China. His research interests include computer vision and machine learning.

Antoni B. Chan received the B.S. and M.Eng. degrees in electrical engineering from Cornell University, Ithaca, NY, in 2000 and 2001, and the Ph.D. degree in electrical and computer engineering from the University of California, San Diego (UCSD), San Diego, in 2008. He is currently a Full Professor in the Department of Computer Science, City University of Hong Kong. His research interests include computer vision, machine learning, pattern recognition, and music analysis.