Towards A Universal Model for Cross-Dataset Crowd Counting

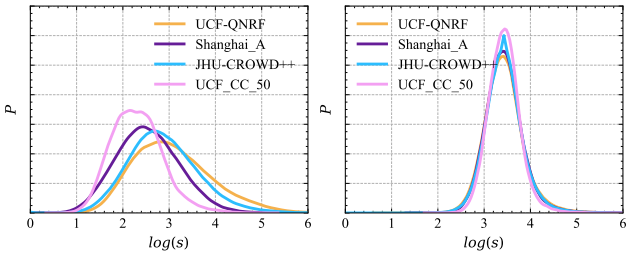
Zhiheng Ma[1], Xiaopeng Hong[3,4], Xing Wei[2]*, Yunfeng Qiu[2], Yihong Gong[2]
[1]College of Artificial Intelligence, Xi'an Jiaotong University
[2]School of Software Engineering, Xi'an Jiaotong University
[3]School of Cyber Science and Engineering, Xi'an Jiaotong University
[4]Research Center for Artificial Intelligence, Peng Cheng Laboratory
mazhiheng@stu.xjtu.edu.cn, {hongxiaopeng,weixing}@mail.xjtu.edu.cn,
yfqiu2015@stu.xjtu.edu.cn,ygong@mail.xjtu.edu.cn

Abstract

*This paper proposes to handle the practical problem of learning a **universal** model for crowd counting across scenes and datasets. We dissect that the crux of this problem is the catastrophic sensitivity of crowd counters to **scale shift**, which is very common in the real world and caused by factors such as different scene layouts and image resolutions. Therefore it is difficult to train a **universal** model that can be applied to various scenes. To address this problem, we propose **scale alignment** as a prime module for establishing a novel crowd counting framework. We derive a closed-form solution to get the optimal image rescaling factors for alignment by minimizing the distances between their scale distributions. A novel neural network together with a loss function based on an efficient sliced Wasserstein distance is also proposed for scale distribution estimation. Benefiting from the proposed method, we have learned a universal model that generally works well on several datasets where can even outperform state-of-the-art models that are particularly fine-tuned for each dataset significantly. Experiments also demonstrate the much better generalizability of our model to unseen scenes.*

1. Introduction

Recently, crowd counting has drawn great attention since it has a variety of applications in the real world. Counting people in the crowd is a challenging task due to severe occlusions and large scale variations of objects caused by factors such as different scene layouts, image resolutions, and viewpoint changes. State-of-the-art crowd counters [55, 44, 28] usually pre-train deep neural networks on large-scale classification datasets like ImageNet [9] and then particularly fine-tune for each crowd counting dataset. Despite the fact that significant progress has been made, existing methods usually follow the training-testing protocol within a single dataset and suffer from significant cross-dataset performance degeneration. On the one hand, the accuracy drops a lot when models are applied to unseen datasets (Table. 6). On the other hand, the model jointly trained on multiple datasets is often inferior to models specifically learned on each dataset (Table. 2), even though much more training images are used. Such poor generalizability of existing crowd counters has seriously restricted their applications in the real scenario.



(a) (b)

Figure 1. (a) Scale distributions (in log-domain) of crowds in four different datasets (before alignment). (b) Scale distributions aligned by our method. Note that there exist significant scale shifts within and across the datasets before alignment, while the distributions can be well aligned through our method.

This paper proposes the practical problem of *universal cross-dataset crowd counting* for real-world applications. The goal is to effectively absorb knowledge from more training data to improve the counting performance and re-

*Corresponding author

Acknowledgements. This work is funded by National Key Research and Development Project of China under Grant No. 2020AAA0105600 and 2019YFB1312000, National Natural Science Foundation of China under Grant No. 62006183 and 62076195, China Postdoctoral Science Foundation under Grant No. 2020M683489, and the Fundamental Research Funds for the Central Universities under Grant No. xzy012020013.

duce deployment cost by obtaining a universal model that can be applied to various scenes. Even though generalization poses a challenge for any machine learning problem, it is especially grievous for crowd counting in two critical respects.

First of all, human annotation is highly labor-intensive for crowd counting, where scenes could be over-crowded with severe occlusions. According to [14], the entire annotation process of UCF-QNRF involved 2,000 human-hours to its completion merely for 1,535 images. Datasets released before that are even much smaller [13, 60]. Due to such a small volume of some existing datasets, the learned crowd counters may easily suffer from overfitting to some extent [50]. Moreover, crowd density and scale distributions often vary substantially from scene to scene, and even for different subareas within the same image due to factors such as different scene layouts and perspective effects. For example, the image resolution in UCF-QNRF [14] ranges from 0.08 to 66.65 megapixels, where the number of persons contained ranges from 49 to 12,865. The scale variation becomes severe when we take into account multiple datasets together. As shown in Fig. 1(a), there exists a significant *scale shift* between different datasets, and the average resolution of UCF-QNRF [14] is three times about that of Shanghai_A [60] (Table. 1).

Because of such scale shift and data bias, it is difficult to directly train a universal model using images from multiple sources.

To further clarify this problem, we investigate how robust crowd counting is against scale shift quantitatively. We test the performance of a state-of-the-art crowd counter BL [28] concerning simple image rescaling. We rescale the testing images of UCF-QNRF from 0.75 to 1.5 and dissect how this can affect the models trained on images with their original resolutions, *i.e.*, scale is shifted from -25% to $+50\%$. The experimental results are catastrophic as can be seen from Fig. 2. For instance, the MAE of BL [28] increases more than 10 points from 88.7 to 100.3 and 103.5 when scales are shifted by -25% and $+50\%$, respectively. Moreover, there is still a loss of accuracy by 3 points even if we slightly enlarge ($+15\%$ shift) images, albeit such a slight amplification does not introduce significant distortion visually or any information loss.

Inspired by the fact that facial analysis tasks such as face recognition [43, 33] and expression estimation [3] often benefit from *face alignment* before analyzing the face in unconstrained environments, we indicate a prime module (termed *scale alignment*) before counting crowds in the scene. Such a step aims to align scale distributions to facilitates learning a single model in various scenes. We first calculate the scale distributions of all scenes and then normalize them to a "standard" one, which is represented by the Wasserstein barycenter [32] of all distributions. Scale

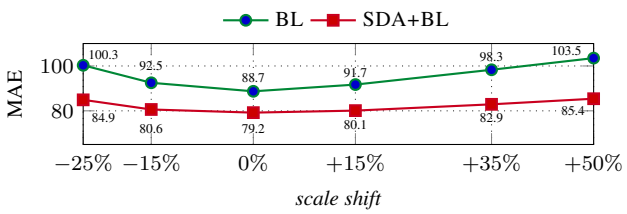


Figure 2. Robustness of crowd counting against scale shift. The curves show the results of BL [28] with respect to different scale shifts of testing images on UCF-QNRF while trained on original resolutions, *e.g.*, $+50\%$ denotes that testing images are rescaled by a factor of 1.5. Our proposed method SDA+BL is much more robust than BL.

shift can be quantified as the sum of Wasserstein distances from each scale distribution to the barycenter. Then, our target is to find the distribution transformation and the corresponding image transformation to minimize the scale shift. Particularly, the translation of distribution in the logarithmic domain corresponds to image rescaling. On this basis, we derive a closed-form solution to obtain the optimal translations and their corresponding rescaling factors. We can easily handle intra-image scale variations at a finer level by dividing the image into patches and seek the rescaling factor for each patch. As shown in Fig. 1 (b), the scale shift between different datasets can be greatly reduced after scale alignment. Moreover, we propose SDNet to predict scale distributions of scenes end-to-end without the need to detect each person. With noticing that scale distributions are highly correlated with spatial positions due to perspective effects, we propose a novel objective function based on a joint distribution representation of scale and position and sliced Wasserstein distance [17] to trained SDNet. To be summarized, we make the following contributions:

- We propose to address the practical problem of universal cross-dataset crowd counting. We establish a prime building block termed *scale alignment* for crowd counting and demonstrate its necessity to this problem.
- We present a scale alignment method by translating scale distributions to their Wasserstein barycenter and derive a closed-form solution to get the optimal translations and corresponding image rescaling factors.
- We propose a novel neural network (SDNet) to directly predict scale distributions for various scenes without detecting each person. A novel loss function based on a joint distribution representation of scale and position and efficient sliced Wasserstein distance is also proposed to optimize SDNet.

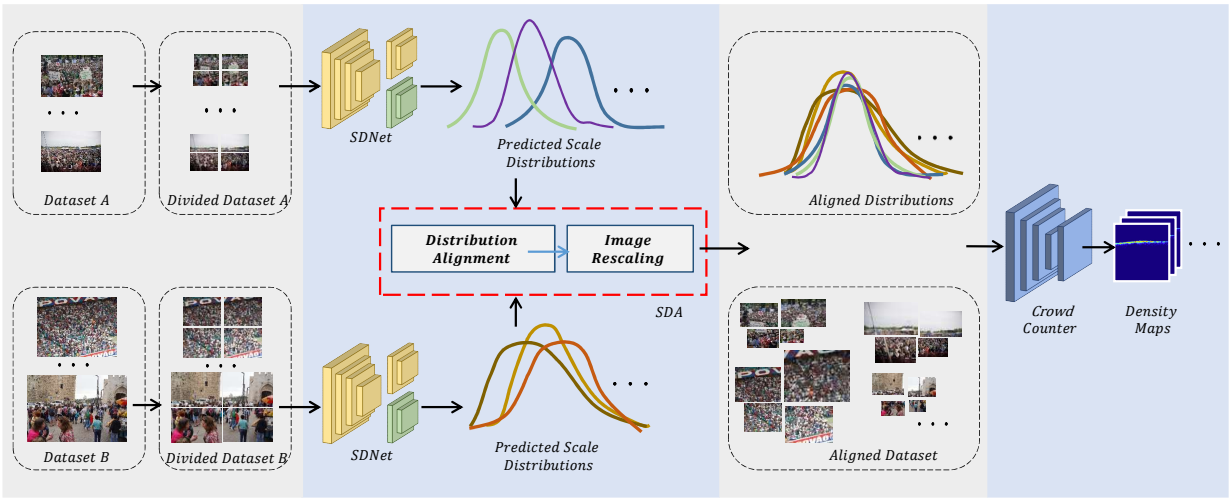


Figure 3. The overall framework of our method. First, we divide images into non-overlapping patches and feed them into SDNet to predict their scale distributions. Then, we perform scale distribution alignment and obtain the optimal rescaling factor for each patch. Finally, we rescale patches and use them to learn the counting model.

2. Related Work

Cross-Scene Crowd Counting. The earlier works of crowd counting are scene-specific so that the model learned for a particular scene can only work in the same scene. The major reason can be ascribed to that earlier benchmark datasets like UCSD [5] and Mall [7] only consist of video clips collected from one or two scenes. To meet the need for crowd counting in real applications, several datasets such as ShanghaiTech [60], UCF-QNRF [14] and JHU-Crowd++ [40] are proposed, where the images are collected from the Internet and consist of various scenes. The emergence of such datasets has led crowd counting to a much more challenging task and gained a lot of attention. With the development of deep learning [53, 54, 6] and such finely annotated datasets, there have made great progresses [46, 59, 60, 44, 36, 29, 47, 52] in this research area. However, current methods usually do not take a strategy to narrow the gaps between images and also do not generalize well to unseen scenes.

Domain Adaptation for Crowd Counting. Recently, domain adaptation has gained increasing attention in crowd counting for adapting the trained model to be used in another domain [11, 18, 50, 10]. CODA [18] adopts an adversarial training strategy to deal with density distribution variations of source and target domain. Wang *et al*. [50] build large-scale synthetic data and translate them to photo-realistic images for crowd counting in real scenes. Han *et al*. [10] introduce a semantic extractor to align features in a semantic space. Wang *et al*. [51] propose to learn the domain shift at the parameter-level and then transfer the source model to the target model. However, there remain significant limitations to our problem to deal with the scale shift for conventional domain adaptation methods. Domain adaptation usually does not leverage the target data to improve performance on the source data and further obtain a universal model.

Scale Handling for Crowd Counting. A variety of work deal with large scale variations by multi-scale feature fusion [25, 24, 29, 8, 39, 31, 24, 16, 2]. Some work proposes to reconstruct the perspective map of the scene. PGCNet [57] fuses multi-scale features and PACNN [37] fuses multi-scale densities according to the predicted perspective map. [59] uses perspective maps to generate ground-truth density maps. While many methods try to handle a wider range of scales, our philosophy differs in that we try to align the scales so that the problem can be easier to tackle. Some work also considers to rescale the image. [1, 35, 34] first classify image patches into different density levels, then resize them with the fixed pre-defined ratio or feed them into different CNN models according to the predicted density levels. L2SM [56] tries to predict rescaling factors for image patches according to their density levels, then these factors are used to resample feature maps to obtain the final prediction. However, density does not directly indicate scale, for example, density could be the same for scenes with only of few of small or large objects. RPNet [58] estimates a perspective map and then warps images to make people have similar scales in that image, but it does not handle scale shifts between images.

3. Method

In this section, we first give the closed-form solution to scale alignment and then describe a novel network structure

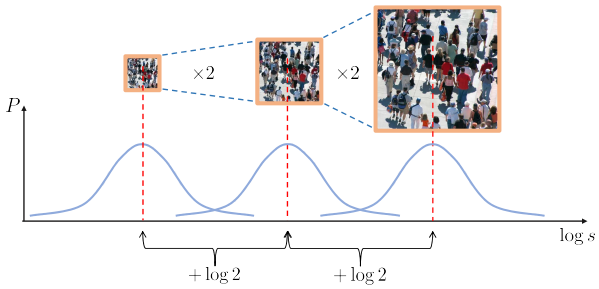


Figure 4. The relationship between image rescaling and scale distribution translation in log-domain.

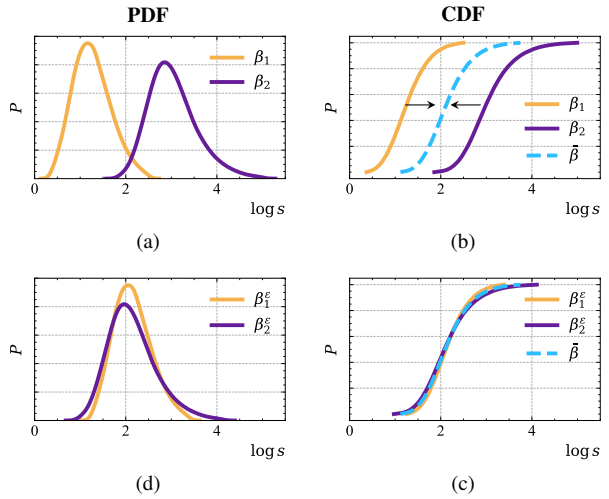


Figure 5. Scale distribution alignment. (a) Scale distributions before alignment. (b) Calculation of the Wasserstein barycenter. (c) Translating scale distributions to the barycenter. (d) Scale distributions after alignment.

(SDNet) used to predict the scale distribution along with its training loss. Fig. 3 visualizes the overall framework of our method.

3.1. Scale Alignment for Crowd Counting

In this section, we establish a dividing-and-rescaling strategy to adjust scales. Particularly, we first divide each image into $C \times C$ non-overlapping patches, then seek the optimal rescaling factor for each patch for alignment. In this way, we can also handle intra-image scale variations in addition to inter-image scale shift. The first task is to find out the suitable scale distribution transformation and its corresponding image transformation. In particular, we observe that once scale distribution is transformed into the logarithmic domain, image rescaling only induces a translation of the distribution in that domain, as illustrated in Fig. 4.

Based on this property, we introduce the alignment process, as illustrated in Fig. 5. First, we start by aligning two scale distributions through translation (image rescaling), then expand it to align multiple. The scale distribution in logarithmic domain is defined as follows:

$$\beta = \frac{1}{M}\sum_{m=1}^{M}\delta(z_m), \tag{1}$$

where M is the total number of people within the image, s_m is the scale of the m-th person, $z_m = \log(s_m)$ is the logarithmic scale, and $\delta(\cdot)$ represents one-dimentional Dirac delta function. Let's denote two different scale distributions as β and $\bar{\beta}$, respectively. For aligning two scale distributions, we can translate one distribution towards another to minimize the distance between them, the objective function is defined as follows:

$$\epsilon^* = \arg\min_{\epsilon\in\mathbb{R}} W_2^2(\beta^\epsilon, \bar{\beta}), \tag{2}$$

where $\beta^\epsilon(z) = \beta(z-\epsilon)$ is the translated scale distribution, ϵ^* is the optimal translation, $\exp(\epsilon^*)$ is the corresponding optimal rescaling factor, and $W_2^2(\cdot,\cdot)$ represents the 2-Wasserstein distance. We adopt Wasserstein distance instead of usual distance such as p-norm distance or KL divergence, because scale distributions have different support sets and may not overlap. Wasserstein distance, representing the least cost of pushing one distribution towards another, is just fit for this situation [32, 30, 21].

Calculating the exact 2-Wasserstein distance between multi-dimensional distributions is costly, which requires solving a linear programming problem [32]. However, 2-Wasserstein distance between one-dimensional distributions has a closed-form solution:

$$W_2^2(u,v) = \int_0^1 (F_u^{-1}(t) - F_v^{-1}(t))^2 \mathrm{d}t, \tag{3}$$

where F_u is the cumulative distribution function (CDF), *i.e.*, $F_u(t) = \int_{-\infty}^{t} I_u(x)\mathrm{d}x$, $I_u(x) = \mathrm{d}u(x)$ is the probability density function (PDF), and F_u^{-1} is the corresponding inverse function. Based on the one-dimensional Wasserstein distance, we can derive the optimal translation ϵ^* as follows:

$$\begin{aligned}
\epsilon^* &= \arg\min_{\epsilon\in\mathbb{R}} \int_0^1 (F_\beta^{-1}(t) + \epsilon - F_{\bar{\beta}}^{-1}(t))^2 \mathrm{d}t \\
\nabla_\epsilon &\int_0^1 (F_\beta^{-1}(t) + \epsilon - F_{\bar{\beta}}^{-1}(t))^2 \mathrm{d}t = 0 \\
\Longrightarrow\ \epsilon^* &= \int_0^1 (F_{\bar{\beta}}^{-1}(t) - F_\beta^{-1}(t))\mathrm{d}t
\end{aligned} \tag{4}$$

where $\beta^\epsilon(z) = \beta(z-\epsilon) \rightarrow F_{\beta^\epsilon}^{-1}(t) = F_\beta^{-1}(t) + \epsilon$. With the above formulation, we can align two arbitrary scale distributions with the optimal translation ϵ^*.

The strategy to align multiple scale distributions includes two steps. First, we calculate the "mean" of all scale distributions. Then, we translate each scale distribution to the

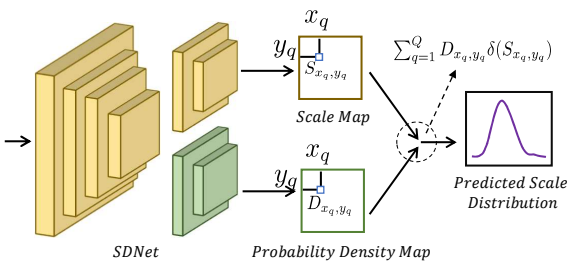


Figure 6. The overall architecture of SDNet.

"mean" scale distribution using Eq. (4). The "mean" scale distributions in the Wasserstein viewpoint actually is the Wasserstein barycenter, which is defined as follows:

$$\bar{\beta} = \arg\min_{\beta} \frac{1}{N} \sum_{n=1}^{N} W_2^2(\beta, \beta_n), \tag{5}$$

where N is the total number of scale distributions. The one-dimensional Wasserstein barycenter also has a closed-form solution. As proven in [4], it can be calculated as follows:

$$F_{\bar{\beta}}^{-1}(t) = \frac{1}{N} \sum_{n=1}^{N} F_{\beta_n}^{-1}(t). \tag{6}$$

Finally, the optimal translation of the n_{th} scale distribution can be calculated as:

$$\epsilon_n^* = \int_0^1 (\frac{1}{N} \sum_{n=1}^{N} F_{\beta_n}^{-1}(t) - F_{\beta_n}^{-1}(t))\mathrm{d}t, \tag{7}$$

and the corresponding optimal rescaling factor of the n_{th} image is equal to $\exp(\epsilon_n^*)$. Once we obtain the scale distribution of each patch, we can align all patches of multiple datasets by the optimal rescaling factors. In the following section, we propose a novel network structure to predict scale distributions without detecting each person.

3.2. Scale Distribution Predictor (SDNet)

This section introduces how to predict the scale distribution of a scene through a CNN model (SDNet). Since the scale of a person is highly correlated with its position in the image, we predict the spatial distribution and the scale distribution at the same time, and use their joint distribution as the supervision. The ground-truth joint distribution is defined as follows:

$$\alpha = \frac{1}{M} \sum_{m=1}^{M} \delta(x_m, y_m, z_m), \tag{8}$$

where x is the abscissa, y is the ordinate, z is the logarithmic scale, $\delta(\cdot,\cdot,\cdot)$ is the three-dimensional Dirac delta function, and M is the total number of points (people) in the image.

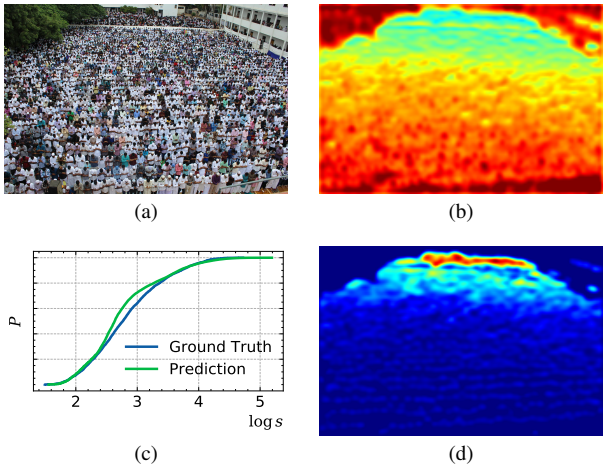


Figure 7. Visualization of the outputs of SDNet. (a) Input image. (b) Predicted scale map. (c) Predicted and the ground-truth scale distribution's CDF. (d) Predicted spatial map.

SDNet consists of a fully-convolutional-network (FCN) with two independent output headers, as shown in Fig. 6. These two headers share the same network structure but have different weights, which are used to predicts the spatial map and the scale map, respectively. We denote the output spatial map as D (D has been normalized by its summation) and the output scale map as S. $S, D \in \mathbb{R}^{W \times H}$ have the same spatial resolution, where W and H are the height and width of each output. Then the predicted joint distribution is defined as follows:

$$\alpha^{pre} = \sum_{q=1}^{Q} D_{x_q,y_q} \delta(d \cdot x_q, d \cdot y_q, S_{x_q,y_q}), \tag{9}$$

where $Q = W \times H$ is the total number of output pixels, and x_q, y_q are the shared abscissa and the ordinate of the two output (the scale map or the spatial map) respectively. D_{x_q,y_q} is the value of the q_{th} pixel in D, and S_{x_q,y_q} is the value of q_{th} pixel in S. Since FCN is adopted as the basic framework, the spatial correspondence between the input image and the outputs is retained. Thus, x_q and y_q can be mapped back to the input spatial coordinate by multiplying the downsample ratio d of SDNet.

As can be seen, the predicted joint distribution has fixed spatial coordinates (determined by the shape of the output), but has a learnable scale coordinate which is predicted by SDNet. We can easily obtain the predicted spatial distribution and the scale distribution by marginalizing the predicted joint distribution. The spatial distribution can be calculated as $\sum_{q=1}^{Q} D_{x_q,y_q} \delta(d \cdot x_q, d \cdot y_q)$, while the predicted scale distribution is derived as follows:

$$\beta^{pre} = \sum_{q=1}^{Q} D_{x_q,y_q} \delta(S_{x_q,y_q}). \tag{10}$$

We visualize the outputs of SDNet in Fig. 7. In the following section, we introduce the objective function used to train SDNet, which is based on sliced Wasserstein distance.

3.3. The Training Objective

The ground-truth joint distribution α and the predicted joint distribution α^{pre} have different support sets. Therefore, Wasserstein distance is a preferable solution to measure their divergence. We can design a objective function to minimize 2-Wasserstein distance between them, *i.e.*, $\mathcal{L} = W_2^2(\alpha, \alpha^{pre})$. However, as mentioned in Sec. 3.1, calculating the exact 2-Wasserstein distance between multi-dimensional distributions is costly. Therefore we adopt sliced 2-Wasserstein distance [17] as an approximation. The loss function is defined as follows:

$$\mathcal{L} = SW_2^2(\alpha, \alpha^{pre}), \tag{11}$$

where $SW_2^2(\cdot,\cdot)$ indicates sliced 2-Wasserstein distance. Sliced Wasserstein distance is proposed to efficiently approximate Wasserstein distance between multi-dimensional distributions, which is built upon Wasserstein distance's one-dimensional closed-form solution (Eq. (3)). Specifically, It first obtains a family of one-dimensional marginal distributions of multi-dimensional distributions through random transform, then calculates the integration of one-dimensional Wasserstein distances:

$$SW_2^2(u, v) = \int_{\mathbb{S}^{d-1}} W_2^2(\mathcal{R}I_u(\cdot,\theta), \mathcal{R}I_v(\cdot,\theta))\mathrm{d}\theta, \tag{12}$$

where $\mathbb{S}^{d-1} \in \mathbb{R}^d$ represents the d-dimentional unit sphere, $W_2^2(\mathcal{R}I_u(\cdot,\theta), \mathcal{R}I_v(\cdot,\theta))$ can be solved by Eq. (3), and $\mathcal{R}$ represents the d-dimentional random transform, which maps a function I to the set of its integrals over the hydroplanes of $\mathbb{R}^d$ as follows:

$$\mathcal{R}I(h,\theta) = \int_{\mathbb{R}^d} I(x)\delta(h - \langle x, \theta\rangle)\mathrm{d}x, \tag{13}$$

where $\delta(\cdot)$ represent the one-dimentional Dirac delta function, $\langle\cdot,\cdot\rangle$ represents the Euclidean inner-product, and $\theta \in \mathbb{S}^{d-1}$. In practice, the integration over the unit sphere $\mathbb{S}^{d-1}$ in Eq. (12) can be approximated by Monte Carlo sampling, which draws samples $\{\theta_l\}_{l=1}^L$ from the uniform distribution on $\mathbb{S}^{d-1}$, where L is the total sample number. Finally, the integration is replaced by a finite-sample average:

$$SW_2^2(u, v) \approx \frac{1}{L}\sum_{l=1}^{L} W_2^2(\mathcal{R}I_u(\cdot,\theta_l).\mathcal{R}I_v(\cdot,\theta_l)). \tag{14}$$

Specifically, if $\theta = (0, 0, 1)^t$, the sliced loss is equal to $W_2^2(\beta, \beta^{pre})$, which is the 2-Wasserstein distance between the ground-truth and predicted scale distributions.

Table 1. Statistics of training datasets. Note that there exist significant dataset biases.

Dataset	Images	Avg. Resolution	Total Count	Avg. Count
UCF-QNRF	1201	2897x2006	1,006,800	838
Shanghai_A	300	872x598	162,350	541
JHU-Crowd++	2272	1450x919	844,387	372
UCF_CC_50	50	902X654	63,969	1279

4. Experiments

In this section, we first introduce the public crowd counting benchmarks used in our experiments. Second, the evaluation metrics and the implementation details of our method are described. Third, we compare our methods with baseline and state-of-the-art methods. Finally, we conduct extensive experiments to study the effect of each component.

Datasets. Our experiments are conducted on four widely-used counting benchmark datasets, *i.e.*, UCF-QNRF [14], Shanghai_A [60], JHU-Crowd++ [40], and UCF_CC_50 [13]. We summarize basic information of these datasets (training data)in Table. 1. Note that these datasets consist of various free-view images in all kinds of environments on which our proposed method is especially focused.

Implement Details. For the crowd counting problem, the scale of a person can be represented by the size of its head, and some datasets (*e.g.*, JHU-Crowd++[40]) provide bounding-box annotations to extract such information. Manually annotating bounding-box is costly, and in most cases, the dataset only provides point annotations. Nevertheless, we can estimate scales from the geometrical distribution of labeled points [60] roughly. We could also leverage object detectors to obtain more accurate scales in scenes without severe occlusions [22]. To keep it simple, we only leverage the point annotations to estimate scales in this work.

Scale Distribution Alignment (SDA) we proposed is a pre-processing technology that can be plugin in front of any crowd counting models. In experiments, We evaluate SDA with four state-of-art crowd counting models, which are CSRNET [20], BL [28], DM [49], and M-SFANET [42]. We use the same hyper-parameters given in the original papers and implement them with their official open-source code [27, 19, 41, 48].

The network structure of SDNet is shown in Fig. 6. As can be seen, SDNet is a fully convolutional network that consists of a single backbone and two regression headers. Specifically, we adopt VGG19 [38] truncated before the last pooling layer as the backbone, and three-layer convolutional regressors as the headers, *i.e.*, $\mathrm{Conv}_{512\times256\times3\times3} + \mathrm{Conv}_{256\times128\times3\times3} + \mathrm{Conv}_{128\times1\times3\times3}$ (Input Channels $\times$ Output Channels $\times$ Kernel Height $\times$ Kernel Width). SDNet's downsample ratio d is 16. We optimize SDNet by

Table 2. Counting performance comparisons with baseline methods. (S) indicates that the model is trained on each dataset separately. (M) indicates that the model is jointly trained on multiple datasets.

Method	UCF-QNRF		Shanghai_A		JHU-Crowd++		UCF_CC_50	
	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE
CSRNET(S) [20]	110.6	190.1	68.2	115.0	85.9	309.2	266.1	397.5
CSRNET(M)	158.0	163.4	76.6	119.5	91.1	276.3	323.7	401.9
SDA+CSRNET(M)	96.3	155.7	58.4	97.9	65.1	269.3	183.4	272.1
BL(S) [28]	88.7	154.8	62.8	101.8	67.1	268.9	229.3	308.2
BL(M)	97.3	168.5	66.1	108.7	66.7	270.4	231.1	313.2
SDA+BL(M)	79.2	134.8	53.6	84.4	58.3	254.5	169.4	243.6
M-SFANET(S) [42]	85.6	151.2	59.7	95.7	65.5	257.4	162.3	276.8
M-SFANET(M)	111.8	186.6	65.1	119.7	61.4	256.9	233.6	385.1
SDA+M-SFANET(M)	79.5	140.7	52.9	87.3	57.4	251.6	159.1	239.4
DM(S) [49]	85.6	148.3	59.7	95.7	66.0	261.4	211.0	291.5
DM(M)	102.6	171.4	63.3	113.5	64.4	229.5	263.9	417.5
SDA+DM(M)	80.7	146.3	55.0	92.7	59.3	248.9	197.5	264.1

Table 3. Counting performance comparisons with state-of-the-art methods. RED indicates the best performance and BLUE indicates the second-best.

Method	UCF-QNRF		Shanghai_A		JHU-Crowd++		UCF_CC_50	
	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE
L2SM [56]	104.7	173.6	64.2	98.4	-	-	188.4	315.3
S-DCNET [55]	104.4	176.1	58.3	95.0	277	426	204.2	301.3
AMSNET [12]	101.8	163.2	56.7	93.4	-	-	208.4	297.3
AMRNET [26]	86.6	152.2	61.59	98.36	-	-	184.0	265.8
LIBRANET [23]	88.1	143.7	55.9	97.1	-	-	181.2	262.2
ASNET [15]	91.6	159.7	57.8	90.1	-	-	174.8	251.6
RPNET [58]	-	-	61.2	96.9	-	-	-	-
MNA [45]	85.8	150.6	61.9	99.6	67.7	258.5	-	-
ADSCNET [2]	71.3	132.5	55.4	97.7	-	-	198.4	267.3
CSRNET [20]	110.6	190.1	68.2	115.0	85.9	309.2	266.1	397.5
BL [28]	88.7	154.8	62.8	101.8	67.1	268.9	229.3	308.2
M-SFANET [42]	85.6	151.2	59.7	95.7	65.5	257.4	162.3	276.8
DM [49]	85.6	148.3	59.7	95.7	66.0	261.4	211.0	291.5
SDA+CSRNET	96.3	155.7	58.4	97.9	65.1	269.3	183.4	272.1
SDA+BL	79.2	134.8	53.6	84.4	58.3	254.5	169.4	243.6
SDA+M-SFANET	79.5	140.7	52.9	87.3	57.4	251.6	159.1	239.4
SDA+DM	80.7	146.3	55.0	92.7	59.3	248.9	197.5	264.1

Adam with the initial learning rate 10^{-5}. Random horizontal flip and random resizing are used to augment the training data. We set $C = 2$ and $L = 5$ in our experiments. SDNet is trained on the original images, and the general crowd counting model is trained on the aligned images. We manually check the training data and test data during multi-dataset training to ensure that there is no data leakage. The training data is aligned according to the ground-truth scale distributions and the Wasserstein barycenter calculated from them, while the testing data is aligned according to the scale distributions predicted by SDNet and the same Wasserstein barycenter used in the training phase.

Comparison with Baselines. We conduct experiments with four state-of-art methods to illustrate the necessity of scale alignment before training on multiple datasets. As shown in Table. 2, the performance of the model trained on multiple datasets before scale alignment is worse than that of the same model trained on each dataset separately, even if more data is used for training. It is because that a single counting model cannot handle such significant scale variations of multiple datasets (as shown in Fig. 1 (a)), which makes the model under-fitting. In stark contrast, models trained on multiple aligned datasets is not only better than the model trained on multiple datasets before scale alignment, but also better than the model trained on each dataset specifically.

Table 4. Upper-bound counting performance of scale alignment. ALIGNED indicates the model is trained on images aligned according to the ground-truth scale distribution.

Method	UCF-QNRF		Shanghai_A		JHU-Crowd++		UCF_CC_50	
	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE
BL [28]	88.7	154.8	62.8	101.8	75.0	299.9	229.3	308.2
BL+ALIGNED	66.2	112.3	42.3	72.9	48.5	250.6	114.7	153.4
Improvement	**22.5**	**42.5**	**20.5**	**28.9**	**26.5**	**49.3**	**114.6**	**154.8**

Table 5. Ablation study. (S) indicates that the model is trained on each dataset separately. (M) indicates that the model is jointly trained on multiple dataset.

Method	UCF-QNRF		Shanghai_A		JHU-Crowd++		UCF_CC_50	
	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE
BL(S) [28]	88.7	154.8	62.8	101.8	75.0	299.9	229.3	308.2
SDA+BL(S)	83.3	143.1	58.4	95.7	62.6	264.1	186.3	261.5
SDA+BL(M)	79.2	134.8	53.6	84.4	58.3	254.5	169.4	243.6

Comparison with State of The Arts. We extensively compare our method with other state-of-the-art methods on four benchmark datasets. Quantitative results are illustrated in Table. 3, and the highlights can be summarized as follows: 1) Our method achieves the best counting performance on JHU-CROWD++, Shanghai_A and UCF_CC_50. Moreover, all predictions are given by a universal model rather than different models specifically trained on each dataset. 2) Our method consistently improves all baseline methods. It is proved that our method can be an effective plug-in to the existing methods.

Ablation Studies. The experiment shown in Table. 4 explores the upper-bound performance of scale alignment. Instead of aligning images according to the scale distribution β^{pre} predicted by SDNet, we directly align images according to the ground-truth scale distribution β, which avoids the error introduced in the prediction of distribution. As can be seen, BL+ALIGNED has made incredible improvements over the baseline. MAEs are reduced by around 20-100 points on the four datasets.

The relative contributions of SDA and multi-dataset training are shown in Table. 5. From the second row of the table we can see that SDA can improve the baseline method trained each dataset separately. It is because that our method can greatly decrease the scale variations. The third row shows that the performance can be further improved if

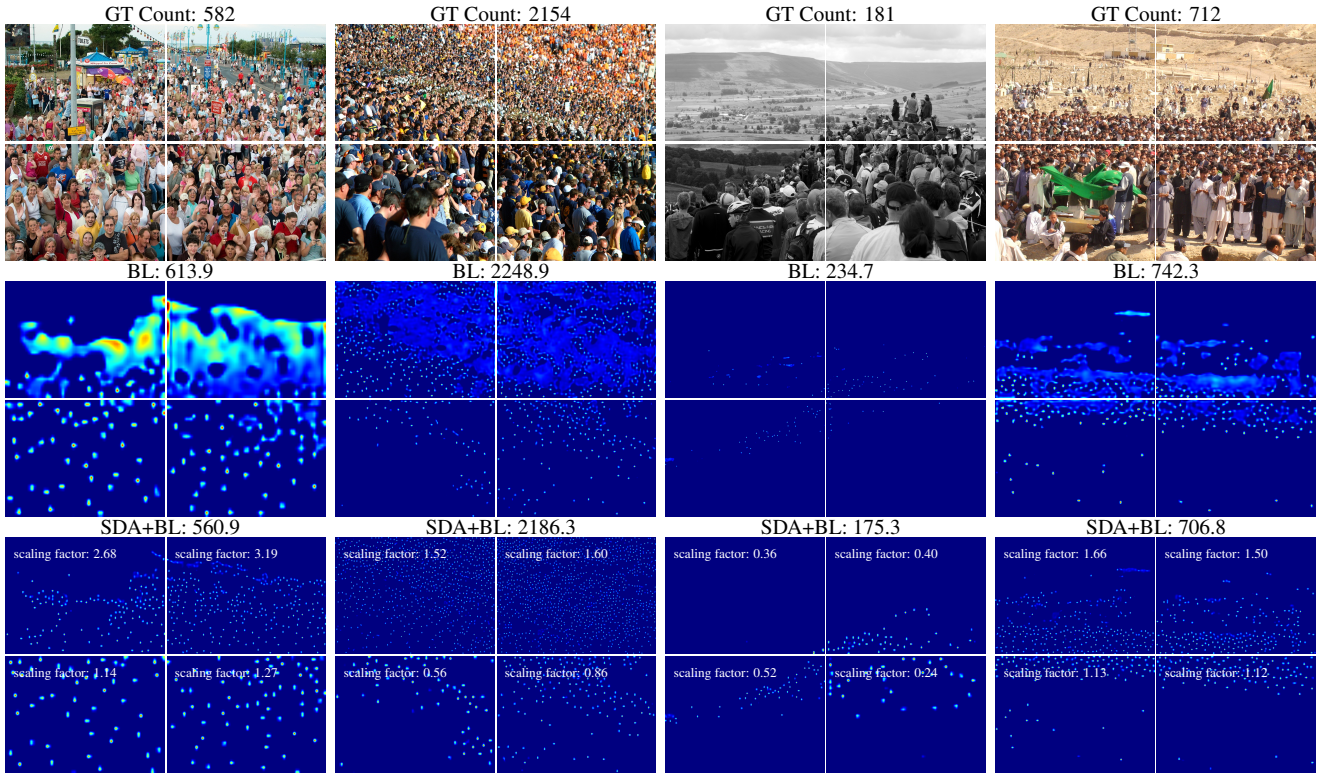


Figure 8. Density maps estimated by BL (the second row) and our SDA+BL (the third row). We also present the scaling factors predicted by our method for comparison across scenes. Please note how scale alignment affects on the density maps.

Table 6. Generalization to unseen datasets. The model is trained on UCF-QNRF while tested on the other datasets.

UCF-QNRF →	Shanghai_A		JHU-Crowd++		UCF_CC_50	
Method	MAE	MSE	MAE	MSE	MAE	MSE
L2SM [56]	73.4	119.4	-	-	-	-
S-DCNET [55]	61.8	102.8	-	-	-	-
CSRNET [20]	75.3	138.7	91.4	317.0	389.8	659.6
SDA+CSRNET	67.3	107.4	80.8	290.2	296.5	426.1
Improvement	**8.0**	**31.3**	**10.6**	**26.8**	**93.3**	**233.5**
BL [28]	69.8	123.8	81.2	303.8	309.6	537.1
SDA+BL	60.5	98.3	76.9	287.4	244.7	354.2
Improvement	**9.3**	**25.5**	**4.3**	**16.4**	**64.9**	**182.9**
M-SFANET [42]	70.1	128.1	84.7	298.2	397.5	666.6
SDA+M-SFANET	62.5	103.4	79.1	283.0	314.9	456.9
Improvement	**7.6**	**24.7**	**5.6**	**15.2**	**82.6**	**209.7**
DM [49]	69.3	120.6	85.2	303.4	317.8	550.2
SDA+DM	59.2	97.4	79.8	289.7	261.6	384.3
Improvement	**10.1**	**23.2**	**5.4**	**13.7**	**56.2**	**165.9**

the model is trained on multiple aligned datasets. It proves that CNN based counting methods can benefit from the increase of data if they are properly aligned.

Generalization to Unseen Datasets. To further illustrate that SDA can help the counting models better generalize to unseen scenes, we perform the cross-dataset evaluation. In this experiment, both SDNet and counting models are trained on one dataset (UCF-QNRF) while evaluated on others. The experimental result is shown in Table. 6. As can be seen, Our method consistently improves baselines as well as achieves the best cross-dataset evaluation performance.

Visualizations. We visualize the estimated density maps of the models trained with BL on the original images and SDA+BL trained on the aligned images respectively in Fig. 8. It can be seen that our SDA+BL is much more robust to the background like trees and gives more accurate and sharper estimates in congested areas. Moreover, our method successfully predicts large-scale people, which the baseline model cannot predict.

5. Conclusion

In this work, we propose and address a practical problem of learning a universal model for counting crowd across scenes and datasets. We dissect that the crux of this problem is the sensitivity for crowd counting to scale shift. Then we propose a simple yet effective scale alignment method in which a closed-form solution is derived to obtain the optimal image rescaling factor. SDNet is further proposed to predict scale distributions. We hope that the proposed method can enlighten the study on adaptability and generalizability of crowd counting and look forward to more research efforts in this direction.

References

[1] Deepak Babu Sam, Shiv Surya, and R Venkatesh Babu. Switching convolutional neural network for crowd counting. In *CVPR*, pages 5744–5752, 2017. 3

[2] Shuai Bai, Zhiqun He, Yu Qiao, Hanzhe Hu, Wei Wu, and Junjie Yan. Adaptive dilated network with self-correction supervision for counting. In *CVPR*, pages 4594–4603, 2020. 3, 7

[3] Vinay Bettadapura. Face expression recognition and analysis: the state of the art. *arXiv preprint arXiv:1203.6722*, 2012. 2

[4] Nicolas Bonneel and Hanspeter Pfister. Sliced wasserstein barycenter of multiple densities. 2013. 5

[5] Antoni B Chan, Zhang-Sheng John Liang, and Nuno Vasconcelos. Privacy preserving crowd monitoring: Counting people without people models or tracking. In *CVPR*, 2008. 3

[6] Dongliang Chang, Yifeng Ding, Jiyang Xie, Ayan Kumar Bhunia, Xiaoxu Li, Zhanyu Ma, Ming Wu, Jun Guo, and Yi-Zhe Song. The devil is in the channels: Mutual-channel loss for fine-grained image classification. *IEEE Transactions on Image Processing*, 29:4683–4695, 2020. 3

[7] Ke Chen, Chen Change Loy, Shaogang Gong, and Tony Xiang. Feature mining for localised crowd counting. In *BMVC*, volume 1, page 3, 2012. 3

[8] Zhi-Qi Cheng, Jun-Xiu Li, Qi Dai, Xiao Wu, Jun-Yan He, and Alexander G Hauptmann. Improving the learning of multi-column convolutional neural network for crowd counting. In *Proceedings of the 27th ACM International Conference on Multimedia*, pages 1897–1906, 2019. 3

[9] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *2009 IEEE conference on computer vision and pattern recognition*, pages 248–255. Ieee, 2009. 1

[10] Tao Han, Junyu Gao, Yuan Yuan, and Qi Wang. Focus on semantic consistency for cross-domain crowd understanding. In *ICASSP 2020-2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 1848–1852. IEEE, 2020. 3

[11] Yuhang He, Zhiheng Ma, Xing Wei, Xiaopeng Hong, Wei Ke, and Yihong Gong. Error-aware density isomorphism reconstruction for unsupervised cross-domain crowd counting. *Proceedings of the AAAI Conference on Artificial Intelligence*, 35(2):1540–1548, May 2021. 3

[12] Yutao Hu, Xiaolong Jiang, Xuhui Liu, Baochang Zhang, Jungong Han, Xianbin Cao, and David Doermann. Nas-count: Counting-by-density with neural architecture search. *arXiv preprint arXiv:2003.00217*, 2020. 7

[13] Haroon Idrees, Imran Saleemi, Cody Seibert, and Mubarak Shah. Multi-source multi-scale counting in extremely dense crowd images. In *CVPR*, 2013. 2, 6

[14] Haroon Idrees, Muhmmad Tayyab, Kishan Athrey, Dong Zhang, Somaya Al-Maadeed, Nasir Rajpoot, and Mubarak Shah. Composition loss for counting, density map estimation and localization in dense crowds. In *ECCV*, pages 532–546, 2018. 2, 3, 6

[15] Xiaoheng Jiang, Li Zhang, Mingliang Xu, Tianzhu Zhang, Pei Lv, Bing Zhou, Xin Yang, and Yanwei Pang. Attention scaling for crowd counting. In *CVPR*, pages 4706–4715, 2020. 7

[16] Di Kang and Antoni Chan. Crowd counting by adaptively fusing predictions from an image pyramid. *BMVC*, 2018. 3

[17] Soheil Kolouri, Kimia Nadjahi, Umut Simsekli, Roland Badeau, and Gustavo Rohde. Generalized sliced wasserstein distances. In *Advances in Neural Information Processing Systems*, pages 261–272, 2019. 2, 6

[18] Wang Li, Li Yongbo, and Xue Xiangyang. Coda: Counting objects via scale-aware adversarial density adaption. In *2019 IEEE International Conference on Multimedia and Expo (ICME)*, pages 193–198. IEEE, 2019. 3

[19] Yuhong Li. https://github.com/leeyeehoo/CSRNet-pytorch, 2018. 6

[20] Yuhong Li, Xiaofan Zhang, and Deming Chen. Csrnet: Dilated convolutional neural networks for understanding the highly congested scenes. In *CVPR*, 2018. 6, 7, 8

[21] Hui Lin, Xiaopeng Hong, Zhiheng Ma, Xing Wei, Yunfeng Qiu, Yaowei Wang, and Yihong Gong. Direct measure matching for crowd counting. In *IJCAI*, 2021. 4

[22] Jiang Liu, Chenqiang Gao, Deyu Meng, and Alexander G. Hauptmann. Decidenet: Counting varying density crowds through attention guided detection and density estimation. In *CVPR*, 2018. 6

[23] Liang Liu, Hao Lu, Hongwei Zou, Haipeng Xiong, Zhiguo Cao, and Chunhua Shen. Weighing counts: Sequential crowd counting by reinforcement learning. *arXiv preprint arXiv:2007.08260*, 2020. 7

[24] Lingbo Liu, Zhilin Qiu, Guanbin Li, Shufan Liu, Wanli Ouyang, and Liang Lin. Crowd counting with deep structured scale integration network. In *ICCV*, pages 1774–1783, 2019. 3

[25] Lingbo Liu, Hongjun Wang, Guanbin Li, Wanli Ouyang, and Liang Lin. Crowd counting using deep recurrent spatial-aware network. In *IJCAI*, pages 849–855, 2018. 3

[26] Xiyang Liu, Jie Yang, and Wenrui Ding. Adaptive mixture regression network with local counting map for crowd counting. *arXiv preprint arXiv:2005.05776*, 2020. 7

[27] Zhiheng Ma. https://github.com/ZhihengCV/Bayesian-Crowd-Counting, 2019. 6

[28] Zhiheng Ma, Xing Wei, Xiaopeng Hong, and Yihong Gong. Bayesian loss for crowd count estimation with point supervision. In *ICCV*, 2019. 1, 2, 6, 7, 8

[29] Zhiheng Ma, Xing Wei, Xiaopeng Hong, and Yihong Gong. Learning scales from points: A scale-aware probabilistic model for crowd counting. In *Proceedings of the 28th ACM International Conference on Multimedia*, MM '20, page 220–228, New York, NY, USA, 2020. Association for Computing Machinery. 3

[30] Zhiheng Ma, Xing Wei, Xiaopeng Hong, Hui Lin, Yunfeng Qiu, and Yihong Gong. Learning to count via unbalanced optimal transport. *Proceedings of the AAAI Conference on Artificial Intelligence*, 35(3):2319–2327, May 2021. 4

[31] Daniel Onoro Rubio and Roberto J López-Sastre. Towards perspective-free object counting with deep learning. In *ECCV*, 2016. 3

[32] Gabriel Peyré, Marco Cuturi, et al. Computational optimal transport: With applications to data science. *Foundations and Trends® in Machine Learning*, 11(5-6):355–607, 2019. 2, 4

[33] Trina Russ, Chris Boehnen, and Tanya Peters. 3d face recognition using 3d alignment for pca. In *CVPR*, pages 1391–1398, 2006. 2

[34] Usman Sajid, Hasan Sajid, Hongcheng Wang, and Guanghui Wang. Zoomcount: A zooming mechanism for crowd counting in static images. *IEEE Transactions on Circuits and Systems for Video Technology*, 2020. 3

[35] Usman Sajid and Guanghui Wang. Plug-and-play rescaling based crowd counting in static images. In *The IEEE Winter Conference on Applications of Computer Vision*, pages 2287–2296, 2020. 3

[36] Deepak Babu Sam, Skand Vishwanath Peri, Amogh Kamath, R Venkatesh Babu, et al. Locate, size and count: Accurately resolving people in dense crowds via detection. *arXiv preprint arXiv:1906.07538*, 2019. 3

[37] Miaojing Shi, Zhaohui Yang, Chao Xu, and Qijun Chen. Revisiting perspective information for efficient crowd counting. In *CVPR*, 2019. 3

[38] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. *arXiv*, 2014. 6

[39] Vishwanath A Sindagi and Vishal M Patel. Multi-level bottom-top and top-bottom feature fusion for crowd counting. In *ICCV*, pages 1002–1012, 2019. 3

[40] Vishwanath A Sindagi, Rajeev Yasarla, and Vishal M Patel. Jhu-crowd++: Large-scale crowd counting dataset and a benchmark method. *arXiv preprint arXiv:2004.03597*, 2020. 3, 6

[41] Pongpisit Thanasutives. https://github.com/Pongpisit-Thanasutives/Variations-of-SFANet-for-Crowd-Counting, 2020. 6

[42] Pongpisit Thanasutives, Ken-ichi Fukui, Masayuki Numao, and Boonserm Kijsirikul. Encoder-decoder based convolutional neural networks with multi-scale-aware modules for crowd counting. *2020 26th International Conference on Pattern Recognition (ICPR)*, 2020. 6, 7, 8

[43] Andrew Wagner, John Wright, Arvind Ganesh, Zihan Zhou, Hossein Mobahi, and Yi Ma. Toward a practical face recognition system: Robust alignment and illumination by sparse representation. *IEEE transactions on pattern analysis and machine intelligence*, 34(2):372–386, 2011. 2

[44] Jia Wan and Antoni Chan. Adaptive density map generation for crowd counting. In *ICCV*, pages 1130–1139, 2019. 1, 3

[45] Jia Wan and Antoni Chan. Modeling noisy annotations for crowd counting. *Advances in Neural Information Processing Systems*, 33, 2020. 7

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

[47] Jia Wan, Wenhan Luo, Baoyuan Wu, Antoni B. Chan, and Wei Liu. Residual regression with semantic prior for crowd counting. In *CVPR*, June 2019. 3

[48] Boyu Wang. https://github.com/cvlab-stonybrook/DM-Count, 2020. 6

[49] Boyu Wang, Huidong Liu, Dimitris Samaras, and Minh Hoai. Distribution matching for crowd counting. *Advances in Neural Information Processing Systems*, 33, 2020. 6, 7, 8

[50] Qi Wang, Junyu Gao, Wei Lin, and Yuan Yuan. Learning from synthetic data for crowd counting in the wild. In *CVPR*, pages 8198–8207, 2019. 2, 3

[51] Qi Wang, Tao Han, Junyu Gao, and Yuan Yuan. Neuron linear transformation: Modeling the domain shift for crowd counting. *IEEE Transactions on Neural Networks and Learning Systems*, 2021. 3

[52] Yabin Wang, Zhiheng Ma, Xing Wei, Shuai Zheng, Yaowei Wang, and Xiaopeng Hong. Eccnas: Efficient crowd counting neural architecture search. In *ACM Transactions On Multimedia Computing, Communications, And Applications (ACM TOMM)*, 2021. 3

[53] Jiyang Xie, Zhanyu Ma, Dongliang Chang, Guoqiang Zhang, and Jun Guo. Gpca: A probabilistic framework for gaussian process embedded channel attention. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2021. 3

[54] Jiyang Xie, Zhanyu Ma, Jianjun Lei, Guoqiang Zhang, Jing-Hao Xue, Zheng-Hua Tan, and Jun Guo. Advanced dropout: A model-free methodology for bayesian dropout optimization. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2021. 3

[55] Haipeng Xiong, Hao Lu, Chengxin Liu, Liang Liu, Zhiguo Cao, and Chunhua Shen. From open set to closed set: Counting objects by spatial divide-and-conquer. In *ICCV*, pages 8362–8371, 2019. 1, 7, 8

[56] Chenfeng Xu, Kai Qiu, Jianlong Fu, Song Bai, Yongchao Xu, and Xiang Bai. Learn to scale: Generating multipolar normalized density maps for crowd counting. In *ICCV*, pages 8382–8390, 2019. 3, 7, 8

[57] Zhaoyi Yan, Yuchen Yuan, Wangmeng Zuo, Xiao Tan, Yezhen Wang, Shilei Wen, and Errui Ding. Perspective-guided convolution networks for crowd counting. In *ICCV*, pages 952–961, 2019. 3

[58] Yifan Yang, Guorong Li, Zhe Wu, Li Su, Qingming Huang, and Nicu Sebe. Reverse perspective network for perspective-aware object counting. In *CVPR*, pages 4374–4383, 2020. 3, 7

[59] Cong Zhang, Hongsheng Li, Xiaogang Wang, and Xiaokang Yang. Cross-scene crowd counting via deep convolutional neural networks. In *CVPR*, pages 833–841, 2015. 3

[60] Yingying Zhang, Desen Zhou, Siqin Chen, Shenghua Gao, and Yi Ma. Single-image crowd counting via multi-column convolutional neural network. In *CVPR*, 2016. 2, 3, 6