Crowd Counting in the Frequency Domain

Weibo Shu[1], Jia Wan[1], Kay Chen Tan[2], Sam Kwong[1], Antoni B. Chan[1]
[1] Dept. of Computer Science, City University of Hong Kong
[2] Dept. of Computing, The Hong Kong Polytechnic University
weiboshu2-c@my.cityu.edu.hk, jiawan1998@gmail.com, kctan@polyu.edu.hk, {cssamk,abchan}@cityu.edu.hk

Abstract

This paper investigates crowd counting in the frequency domain, which is a novel direction compared to the traditional view in the spatial domain. By transforming the density map into the frequency domain and using the properties of the characteristic function, we propose a novel method that is simple, effective, and efficient. The solid theoretical analysis ends up as an implementation-friendly loss function, which requires only standard tensor operations in the training process. We prove that our loss function is an upper bound of the pseudo sup norm metric between the ground truth and the prediction density map (over all of their sub-regions), and demonstrate its efficacy and efficiency versus other loss functions. The experimental results also show its competitiveness to the state-of-the-art on five benchmark data sets: ShanghaiTech A & B, UCF-QNRF, JHU++, and NWPU. Our codes will be available at: wb-shu/Crowd_Counting_in_the_Frequency_Domain

1. Introduction

The research field of image-based crowd counting has been flourishing since the density map based method is proposed [12]. After the Multi-Column Neural Network (MCNN) shows the power of using the deep Convolution Neural Network (CNN) to generate the density map [46], the combination of deep learning and density map learning has led the state-of-the-art. Among current state-of-the-art, the Bayesian Loss (BL) distinguishes itself by only changing the loss function in the whole pipeline [21]. The BL used the ground truth dot map to calculate class conditional distributions (CCD) for each position rather than generating a discrete density map as supervision. This elegant method showed that how to exploit the ground truth to offer proper supervisory information (i.e., the loss function) has a large impact on the final performance.

The ground truth dot map in itself has a large amount of useful information. Therefore, how to fully utilize the ground truth to provide high-quality supervisory information becomes one of the active issues in crowd counting. This issue has yielded a number of prominent research works recently. Among them, the Distribution Matching (DMCount) [37] and the Generalized Loss (GL) [35] used the optimal transport (OT) distance as the loss function between predicted density maps and the ground truth dot maps. When the DNN adjusts one predicted pixel value according to the pixel-wise L2 loss, it only considers the influence on the same pixel in the ground truth. In contrast, when the DNN adjusts one predicted pixel value according to the OT loss, it must consider the influence of all nearby pixels in the ground-truth according to their distances – the OT problem is a global optimization problem that jointly considers the transport of all pixels. Therefore, the family of OT losses is able to better exploit the position information of the ground truth to provide high-quality supervision.

Another approach that better used the groundtruth's position information is the Purely Point-Based Framework (P2PNet) [31], which directly taught the network to predict people's head positions in the ground truth. The exact position information in the ground truth was used in training by calculating a one vs. one match between the prediction and the ground truth.

However, these SOTA methods also have some flaws. Firstly, both the OT loss [35, 37] and the P2PNet [31] require inefficient external algorithms to extract the spatial information from the ground truth in each training step. For the OT loss, the Sinkhorn algorithm [22] is executed to obtain the optimal transport matrix, while for P2PNet, the Hungarian algorithm [11] is required to get the one vs. one point matches. Both of these algorithms require a number of iterations and are carried out in each training step, which makes the OT/P2PNet training less efficient. Furthermore, the complex logic of the Hungarian algorithm makes it hard to use the advantage of parallelization in GPU. Indeed the official codes of P2PNet implemented it in CPU, which further decreases the efficiency compared with methods whose pipelines are fully implemented in GPU.

Secondly, although the position information is fully used in OT/P2PNet, the counting information of the ground truth

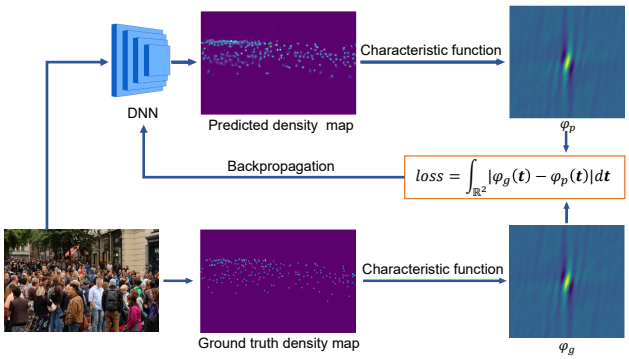


Figure 1. Our framework for crowd counting in the frequency domain. The dispersed spatial information in the predicted and ground-truth density maps is converted to compact information in the frequency domain by computing their characteristic functions. Then our loss is the L1-norm between the characteristic functions. We prove that our loss is an upper-bound to a pseudo sup norm metric between density maps (over all sub-regions), which makes it provide high-quality supervision for training crowd counting models, and consequently outperform other SOTA loss functions on crowd counting.

is underexploited. Hence, OT/P2PNet used different methods to remedy this issue: OT loss introduced extra loss terms; P2PNet required setting the maximum number of points in each patch of images, but this information is actually unknown in the test image. These additional remedies also created additional hyperparameters, which need to be tuned or balanced.

To address these problems in the current SOTA, we investigate a new method that fully harnesses both the position information and counting information of the ground truth. We hope the new method both provides the high-quality spatial supervisory information for training, and easily extracts it from the ground truth. Such properties may be hard to fulfill in the spatial domain, so instead we turn to an analysis in the frequency domain. Our solution is to use the characteristic functions of finite measures (i.e., unnormalized probability densities). It is intuitive that a density map is a finite measure on the 2D plane, and the position information and counting information are all in the finite measure. However, in the spatial domain, that information is spread out everywhere, and thus the global spatial information is hard to use without some external algorithms to extract it (e.g., the Sinkhorn algorithm [22] for the OT loss [35, 37], the Hungarian algorithm [11] for the P2PNet [11]). In contrast, if the finite measure is transformed into the frequency domain, then the spatial information is hierarchically organized in a compact range around the origin in the frequency domain. Values closer to the origin contain a larger proportion of global spatial information, while values further from the origin contain a larger proportion of local position information. Hence, a proper loss function on the frequency domain will adequately deliver all the information to the DNN for training (see Fig. 1).

The characteristic function is exactly a representation of the finite measure on the frequency domain. Although it is originally defined for probability distributions, here we extend the definition to the finite measure so that some vital properties of the original definition are carried over. These properties play an important role throughout the analysis in the paper, and we will show their effects later. In summary, the contributions in the paper are:

- We extend the definition of the characteristic function from probability distributions to finite measures, as well as prove or intensify some of its vital properties. Thus, we transfer the learning problem from supervision with spatial density maps to supervision with frequency-domain characteristic functions, where the latter compactly summarizes the dispersed spatial information, which is more suitable for supervision. To the best of our knowledge, this is the first work investigating crowd counting in the frequency domain.
- Using properties of the characteristic function, we propose a simple, effective, and efficient loss function that provides high-quality supervisory information for training, and, in contrast to previous works, does not require external algorithms for extracting spatial information.
- We prove that minimizing our loss function will decrease the upper bound of a pseudo sup norm metric between the predicted and the ground truth density map (over all sub-regions), which is effective for crowd counting.
- The experimental results on five benchmark datasets show our method's competitiveness, and our loss function outperforms a large number of baseline and SOTA loss functions, while also being more efficient.

2. Related works

Image-based crowd counting. Research on image-based crowd counting can be divided into several stages. The early methods used various features to detect the heads/people in the image [8, 13, 15, 32, 41, 45, 47], and then the counting was based on the detection results. The second stage is based on "image to count", where methods directly regressed the people count from the input image [5–7, 16, 23, 25, 38]. The current prevalent methodology is regressing the density map from the image, which forms the basis of most recent works due to its effectiveness as an intermediate representation.

Density map regression. The density map method was first proposed in [12]. Afterwards, [46] reached a milestone by using the CNN to predict the density map from the image, and then the combination of deep learning and density map regression has led the trend in the crowd counting. The recent supervised learning methods can be roughly divided into two categories: improving network architecture

design [2, 4, 14, 17, 18, 26, 43, 44]; and improving loss functions for training [21, 31, 34, 35, 37]. Our method belongs to the second category.

Improving training and loss functions. Recent methods [21, 31, 34, 35, 37] aim to extract high-quality supervisory information from the ground truth to make training more effective (e.g., robust to spatial annotation noise or more accurate in position match). The representative works [31, 35, 37] achieved remarkable results, but they require inefficient external algorithms to extract spatial information from the ground truth on each training image. On the other hand, they lack the exploitation of the counting information, while they focus on the local position information of the ground truth. In contrast, by transforming the dispersed spatial information to compact frequency-domain information, our method can naturally use the counting information and position information simultaneously for supervision. Furthermore, our method does not require external algorithms for extracting this information.

3. Crowd counting in the frequency domain

In this section, we introduce our framework of crowd counting in the frequency domain, i.e., the crowd counting based on characteristic functions of density maps. First, we introduce the mathematical concepts of measure and characteristic function, and then extend the definition of the characteristic function from distribution to the density map. Second, we prove some useful properties of the characteristic function of density maps. Third, we elaborate on our loss function based on the characteristic function, and analyze its properties. Fourth, we discuss how to implement our method based on empirical and theoretical supports.

3.1. Characteristic function of the density map

In mathematics, the measure is a non-negative set function defined on a σ-algebra, which possesses the property of σ-additivity. The formal definition is as follows.

Definition 1 (Measure [33]) *A **measure** is a set function m defined on a measurable space $(\Omega, \mathcal{F})$, where Ω is the total space and the family of sets $\mathcal{F}$ is a σ-algebra (comprising subsets of Ω that are closed under union, intersection, and complement), that satisfies:*

(i) non-negativity: $m(A) \geq 0, \forall A \in \mathcal{F}$.

(ii) σ-additivity: $m(\emptyset) = 0$, where $\emptyset$ is the empty set, and $m(\bigcup_{i=1}^{\infty} A_i) = \sum_{i=1}^{\infty} m(A_i)$ for a countable set $\{A_i | A_i \in \mathcal{F}, A_i \cap A_j = \emptyset \text{ if } i \neq j\}$.

*If $m(\Omega) < \infty$, i.e., the total measure is finite, then it is a **finite measure**.*

Thus, the density map is a finite measure on the 2D plane – Ω is the 2D Euclidean space $\mathbb{R}^2$ and $\mathcal{F}$ are all Borel sets.

Definition 2 (Density Map) *A **density map** in crowd counting is a finite measure defined on $(\mathbb{R}^2, \mathcal{B}_{\mathbb{R}^2})$, where $\mathbb{R}^2$ is the 2D Euclidean space and $\mathcal{B}_{\mathbb{R}^2}$ is all the Borel sets on $\mathbb{R}^2$. The density map's total measure on $\mathbb{R}^2$ equals the total people count.*

A discrete density map *is a density map whose measure is only distributed on a set of finite points, i.e., if the density map m satisfies the following property:*

$$m(A) = \sum_{i=1}^{n} m(\{x_i\} \cap A), \forall A \in \mathcal{B}_{\mathbb{R}^2}, \tag{1}$$

*where $x_i \in \mathbb{R}^2$ are those points with non-zero measure, then m is a **discrete density map**.*

Next, we introduce the definition of the characteristic function for probability distributions, which is a class of special finite measures with total measure of 1.

Definition 3 (Characteristic Function for Distributions [3]) *Given a distribution d defined on $\mathbb{R}^n$, its **characteristic function** φ_d is a complex-valued function defined on $\mathbb{R}^n$:*

$$\varphi_d(\mathbf{t}) = \mathbb{E}_{\mathbf{X} \sim d}[e^{\mathrm{i}\langle \mathbf{t}, \mathbf{X} \rangle}], \tag{2}$$

where $\mathbf{t} \in \mathbb{R}^n$ is the independent variable of the frequency domain, $\mathbb{E}_{\mathbf{X} \sim d}$ is expectation under $\mathbf{X}$ with distribution d, and i is the imaginary unit.

Since the probability distribution is just the finite measure with the total measure of 1, we can naturally extend the definition of characteristic functions to finite measures (i.e., density maps).

Definition 4 (Characteristic Function for Measures) *Given a finite measure m defined on $\mathbb{R}^n$, its **characteristic function** φ_m is a complex-valued function defined on $\mathbb{R}^n$:*

$$\varphi_m(\mathbf{t}) = \int_{\mathbb{R}^n} e^{\mathrm{i}\langle \mathbf{t}, \mathbf{x} \rangle} dm(\mathbf{x}), \tag{3}$$

where $dm(\mathbf{x})$ means the integral is calculated based on measure m.

Thus, the characteristic function of a density map can be calculated by Def. 2 and Def. 4.

3.2. Properties of the characteristic function

Next we derive several vital properties of characteristic functions of finite measures. For clarity, we will directly present these properties for density maps, rather than finite measures. Thus, in the remaining, the terminology "density map" refers the finite measure defined on $(\mathbb{R}^2, \mathcal{B}_{\mathbb{R}^2})$ (see Def. 2). All proofs appear in the supplemental.

Property 1 (Uniqueness) *The characteristic function* ***uniquely*** *determines the density map and vice versa.*

Suppose that φ_{m_1} and φ_{m_2} are two characteristic functions derived from two density maps m_1 and m_2 respectively. Then, we have

$$\varphi_{m_1}(\mathbf{t}) = \varphi_{m_2}(\mathbf{t}) \ a.e. \tag{4}$$

if and only if

$$m_1(A) = m_2(A), \ \forall A \in \mathcal{B}_{\mathbb{R}^2}. \tag{5}$$

We denote this as $m_1 = m_2$. In (4), a.e. means $\mathcal{L}(\{\mathbf{t} \in \mathbb{R}^2 | \varphi_{m_1}(\mathbf{t}) \neq \varphi_{m_2}(\mathbf{t})\}) = 0$, where $\mathcal{L}$ is the Lebesgue measure.

Remark Intuitively, this property states that if two density maps' characteristic functions are the same, then the density maps are the same, and vice versa. This property mainly removes the problem of non-unique optimal solutions in the loss function, which is pointed out by [37] as a potential drawback of the BL [21].

Property 2 (Linearity) *Suppose that m_3 is a linear combination of two density maps m_1 and m_2,*

$$m_3 = \alpha m_1 + \beta m_2, \ \alpha, \beta \geq 0 \tag{6}$$

then

$$\varphi_{m_3}(\mathbf{t}) = \alpha \varphi_{m_1}(\mathbf{t}) + \beta \varphi_{m_2}(\mathbf{t}). \tag{7}$$

Remark This property helps to simplify the derivation of the characteristic functions of predicted and ground truth density maps, since they are actually the linear combinations of simple singleton measures or Gaussian distributions.

Property 3 (Inversion Formula) *For a density map m, suppose there is a box area $A = [a_1, b_1] \times [a_2, b_2]$ in $\mathbb{R}^2$ with zero measure boundary, i.e.,*

$$m(\partial A) = 0 \tag{8}$$

where ∂A means the boundary of A, then we have

$$m(A) = \lim_{T \to \infty} \frac{1}{(2\pi)^2} \int_{[-T,T]^2} \int_A \varphi_m(\mathbf{t}) e^{-\mathrm{i}\langle \mathbf{t}, \mathbf{x} \rangle} d\mathbf{x} d\mathbf{t} \tag{9}$$

where $d\mathbf{x}$ and $d\mathbf{t}$ mean both the first and second integral are calculated based on Lebesgue measure.[1]

[1]Note that when $d\mathbf{x}$ or $d\mathbf{t}$ appears in the next context, it also means the integral is calculated based on Lebesgue measure.

Remark This is a crucial property bridging the density map and its characteristic function. This property illustrates how the spatial domain's counting information and position information are together absorbed into the characteristic function. Although the integral in (9) is on the whole frequency domain $\mathbb{R}^2$, Fig. 2 shows that most information is concentrated on a very compact range in the frequency domain. Thus, the characteristic function of the density map has near-zero value outside this range, which makes little contributions to the integral. With only the information concentrated on a small range in the frequency domain, every area's people count and population distribution can be known by Property 3. Compared with the dispersed information in the spatial domain, the compact information in the frequency domain is more suitable to use for training.

Property 4 (Lipschitz Continuity) *If a density map m is a discrete density map (see Def. 2) or a discrete density map convolved with a Gaussian kernel, then the characteristic function $\varphi_m(\mathbf{t})$ is Lipschitz continuous.*

Remark This property plays an important role in the implementation of our method. Since there is no analytical solution to our method, we use an approximation method based on this property to calculate the loss.

3.3. Characteristic function loss

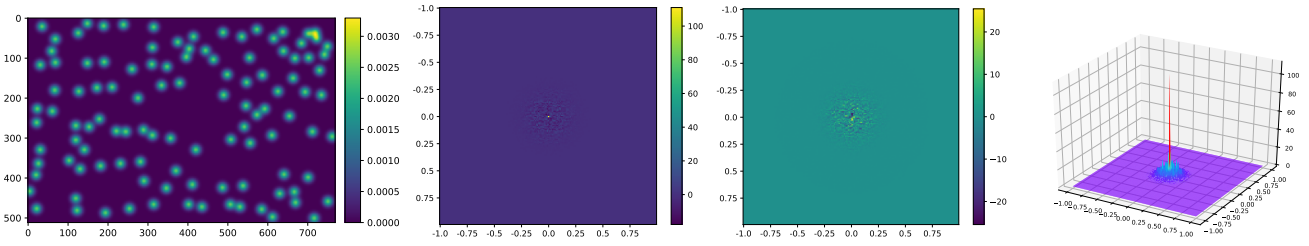
In this subsection, we propose our loss function based on characteristic functions and analyze it theoretically. Fig. 1 shows the flow chart of our method. Given the ground truth density map m_g and the predicted density maps m_p, our loss function is the L_1-norm metric between their characteristic functions φ_{m_g} and φ_{m_p}, i.e.,[2]

$$l_{\text{chf}}(m_g, m_p) = \int_{\mathbb{R}^2} |\varphi_{m_g}(\mathbf{t}) - \varphi_{m_p}(\mathbf{t})| d\mathbf{t} \tag{10}$$

We denote our loss l_{chf} as the **chf loss**.

To prove its effectiveness, we first show that the chf loss is not *underdetermined*, which is proposed in [37] to describe the case when the loss l can be zero when two density maps m_1 and m_2 are not equal, i.e., $\exists m_1 \neq m_2, \text{s.t. } l(m_1, m_2) = 0$. If a loss is underdetermined, than minimizing the loss may not make the prediction close to the ground truth [37]. Hence a good loss function should not be underdetermined, which is the case for our chf loss. (All proofs are in the supplemental.)

[2]Note here that we directly use the Lebesgue integral on $\mathbb{R}^2$, but in (9) we use a limit formula rather than the direct Lebesgue integral. As they are not always identical, some care is needed and we provide the mathematical details in the supplementary.



(a) (b) (c) (d)

Figure 2. Comparison between the information distribution in the spatial domain and the frequency domain. (a) the density map m in the spatial domain $[0, 512] \times [0, 749]$; (b) the real part of the characteristic function φ_m of m, in range $[-1, 1]^2$; (c) the imaginary part of the characteristic function φ_m in range $[-1, 1]^2$; (d) the spectrum of the characteristic function, i.e., $|\varphi_m|$ in range $[-1, 1]^2$. The information is distributed everywhere in the spatial domain, while the information in the frequency domain is concentrated on a small compact range near the origin. By Property 3, that compact frequency information can recover the information anywhere in the spatial domain.

Proposition 1 *The chf loss l_{chf} is not underdetermined.*

Next we show what will happen to the predicted density map when the chf loss decreases w.r.t. the ground-truth.

Proposition 2 *For the ground truth density map m_g and the predicted density map m_p,*

$$|m_g(A) - m_p(A)| \leq (2\pi)^{-2} l_{\text{chf}}(m_g, m_p)\mathcal{L}(A), \quad (11)$$

for any open set $A \in \mathcal{B}_{\mathbb{R}^2}$. Here $\mathcal{L}$ means the Lebesgue measure, i.e., area of A.

This proposition reveals why the chf loss is effective. Rearranging the terms in (11), we obtain

$$(2\pi)^2 \frac{|m_g(A)-m_p(A)|}{\mathcal{L}(A)} \leq l_{\text{chf}}(m_g, m_p), \forall A \in \mathcal{B}_{\mathbb{R}^2}. \quad (12)$$

and thus the chf loss is an upper-bound to the normalized counting errors of all sub-regions A in the density map, $\frac{|m_g(A)-m_p(A)|}{\mathcal{L}(A)}$, where the normalization is based on the sub-region area $\mathcal{L}(A)$.

Next, we define the "sup norm" metric between two density maps, which is the largest normalized error over all sub-regions, as

$$\Delta(m_g, m_p) = \sup_{\partial A=\emptyset \wedge \mathcal{L}(A)\neq 0} \frac{|m_g(A) - m_p(A)|}{\mathcal{L}(A)}, \quad (13)$$

where $\partial A = \emptyset$ means A has an empty boundary (i.e., it is an open set), and $\mathcal{L}(A) \neq 0$ means it has non-trivial Lebesgue measure. Our sup norm in (13) has similar flavor to the MESA (Maximum Excess over SubArrays) loss from [12], except that MESA is defined using rectangular regions and is unnormalized, whereas ours is defined over all sub-regions and is normalized.

Finally, we obtain

$$(2\pi)^2 \Delta(m_g, m_p) \leq l_{\text{chf}}(m_g, m_p), \quad (14)$$

and thus minimizing the chf loss is equivalent to minimizing the upper bound of our sup norm metric $\Delta(m_g, m_p)$ between the prediction and the ground truth, i.e., *minimizing the largest normalized error over all sub-regions.* Using the chf loss for training will apply supervision more evenly on all region counts, which avoids individual pixel-wise fluctuations in the spatial domain (e.g., inherent with pixel-wise losses like L2). Specifically, (12-14) show that decreasing the chf loss will ensure the closeness of the prediction to the ground-truth for all areas in the spatial domain, i.e., both local and global counts are considered for supervision.

3.4. Implementation of the chf loss

Since the integral in (10) for the chf loss is not analytically solvable, we next propose an approximation to the chf loss in this subsection. The integral in the chf loss is approximated using two steps: 1) truncating the infinite integral range on a finite range; 2) using the Riemann sum to approximate the integral in this finite range.

Truncating the integral. As illustrated in Fig. 2, the characteristic function values outside a compact central range are typically very small. The empirical and theoretical evidence also support that the integral on the compact range has small difference from the integral on the whole domain. In theory, considering a discrete density map obtained by convolving a dot map with a Gaussian kernel, then the following proposition gives an upper bound to the average error between the original density map and the reconstructed density map.

Proposition 3 *Suppose the density map m is obtained by convolving a discrete dot map with a Gaussian kernel whose bandwidth is σ, and the reconstructed density map from its characteristic function φ_m restricted on the disk $B(0, r)$ is $\tilde{m}$. Let T be the total measure of m. Then on any non-empty box area A with trivial boundary, i.e., $m(\partial A) = 0$, we have*

$$\frac{|m(A)-\tilde{m}(A)|}{\mathcal{L}(A)} \leq \frac{T \exp\{\frac{-\sigma^2 r^2}{2}\}}{2\pi\sigma^2}. \quad (15)$$

Proposition 3 indicates that the the error between the original and the reconstructed ground truth density map can be well bounded by an exponentially decaying term if we take the dot map convolved with a Gaussian kernel as the

ground truth. In the concrete implementation, we use the Gaussian kernel with bandwidth 8 which is the conventional setting. If we confine the integral range from $\mathbb{R}^2$ to the disk $\{||\mathbf{x}||^2 < 0.5\}$ and suppose the total people count is at most 0.1 million people in a training image, then by Proposition 3 the error upper bound is approximately 0.08.

The above upper bound is loose and in practical situation the approximation is even better. Fig. 3 shows the comparison between the original density map and the reconstructed density map from the truncated characteristic function. They are nearly the same, which suggests not much information is lost when truncating the integral.

Approximating the integral. Although the integral is confined to a small range, the integral of chf loss still needs to be approximated with the Riemann sum. Property 4 shows the nice continuity of the characteristic function, which gives a firm theoretical guarantee for the Riemann sum approximation. Furthermore, some empirical results will be shown in Subsection 4.4.

The approximation introduces two hyperparameters in our method: 1) the granularity of the grid in the Riemann sum; 2) the integral range. One of the important functions of Property 4 is to decouple the two hyperparameters. Property 4 demonstrates a uniform continuity of the characteristic function, which means the intensity of the continuity is similar everywhere in the domain. Therefore, if the granularity of the Riemann sum approximation works fine on some integral range, then it also works on any integral range. Hence, the granularity of the Riemann sum approximation is independent of the integral range. Then the hyperparameter search is converted from a two-dimensional grid search to two one-dimensional linear searches, which are more efficient.

Implementation. Finally, the implementation of our chf loss is illustrated in Fig. 4. For a given image, let there be M people in the ground truth with locations $\{\boldsymbol{\mu}_j\}_{j=1}^M$. Convolving each person j with a Gaussian kernel with covariance matrix $\boldsymbol{\Sigma}_j$ yields a Gaussian distribution $\mathcal{N}(\boldsymbol{\mu}_j, \boldsymbol{\Sigma}_j)$. Then, the ground truth density map m_g is the stack of all of the M Gaussian distribution, i.e.,

$$m_g = \sum_{j=1}^{M} \mathcal{N}(\boldsymbol{\mu}_j, \boldsymbol{\Sigma}_j), \tag{16}$$

and by Property 2, we have

$$\varphi_{m_g}(\mathbf{t}) = \sum_{j=1}^{M} \varphi_{\mathcal{N}(\boldsymbol{\mu}_j, \boldsymbol{\Sigma}_j)}(\mathbf{t}) = \sum_{j=1}^{M} \exp\left(\mathrm{i}\boldsymbol{\mu}_j^T \mathbf{t} - \tfrac{\mathbf{t}^T \boldsymbol{\Sigma}_j \mathbf{t}}{2}\right). \tag{17}$$

Note that φ_{m_g} can be calculated directly from the positions and covariances $(\boldsymbol{\mu}_j, \boldsymbol{\Sigma}_j)$ without computing the ground-truth density map with convolution.

Let $P(\mathbf{x})$ be the values in 2D matrix corresponding to the predicted density map at spatial locations $\mathbf{x}$. The prediction density map m_p is also a stack of singleton measures, and by Property 2 again we have

$$\varphi_{m_p}(\mathbf{t}) = \sum_{\mathbf{x}} P(\mathbf{x}) \varphi_{\delta(\mathbf{x})} = \sum_{\mathbf{x}} \exp(\mathrm{i}\mathbf{x}^T \mathbf{t}) P(\mathbf{x}), \tag{18}$$

where $\delta(\mathbf{x})$ is the impulse function located at $\mathbf{x}$.

Suppose that we truncate the integral range in (10) to $\tilde{R}$, and use the Riemann sum approximation. Then $\tilde{R}$ is divided evenly into small square grids. Suppose the center points of all the grids construct the set R, and the edge size of the square grid is c, then the approximation to the integral in (10) is

$$\hat{l}_{\mathrm{chf}}(m_g, m_p) = c^2 \sum_{\mathbf{t} \in R} \left| \varphi_{m_g}(\mathbf{t}) - \varphi_{m_p}(\mathbf{t}) \right|. \tag{19}$$

Finally, substituting (17) and (18) into (19) gives the final form of our chf loss:

$$\hat{l}_{\mathrm{chf}}(m_g, m_p) \tag{20}$$
$$= c^2 \sum_{\mathbf{t} \in R} \left| \sum_{j=1}^{M} \exp(\mathrm{i}\boldsymbol{\mu}_j^T \mathbf{t} - \tfrac{\mathbf{t}^T \boldsymbol{\Sigma}_j \mathbf{t}}{2}) - \sum_{\mathbf{x}} \exp(\mathrm{i}\mathbf{x}^T \mathbf{t}) P(\mathbf{x}) \right|.$$

4. Experiments

In this section we present the experiment results validating the efficacy of our chf loss function, including comparisons with SOTA and ablation studies.

4.1. Experiment setup

The experiments are carried out on five benchmark data sets: ShanghaiTech A & B [46], UCF-QNRF [9], JHU++ [29, 30], and NWPU [39]. For UCF-QNRF, we resize the images such that the image's shortest length does not exceed 1536. For JHU++ and NWPU, similar resizing is performed for length 2048. The image crop window size is 384 for UCF-QNRF, JHU++, and NWPU, 128 for ShanghaiTech A, and 512 for ShanghaiTech B.

The density map regression network consists of the feature extraction layers of VGG19 [27] connected to a regression module composed of three convolution layers, which is the same architecture used in [21, 34, 35, 37]. Training uses our proposed chf loss in (20), denoted as "ChfL", and the optimizer is Adam [10] with the learning rate 1e-5 and the weight decay 1e-4.

For the ground-truth density map, we use a Gaussian kernel with the conventional bandwidth 8 pixels. Note that we do not need to calculate the ground-truth density map in the implementation, since its characteristic function can be directly obtained in closed-form from the annotated positions (see Eq. 17). For the other two hyperparameters of our chf loss: 1) the integral range is set to $[-0.3, 0.3]^2$ for all data sets; 2) the grid granularity in the Riemann sum approximation is set to 0.01 for all datasets.

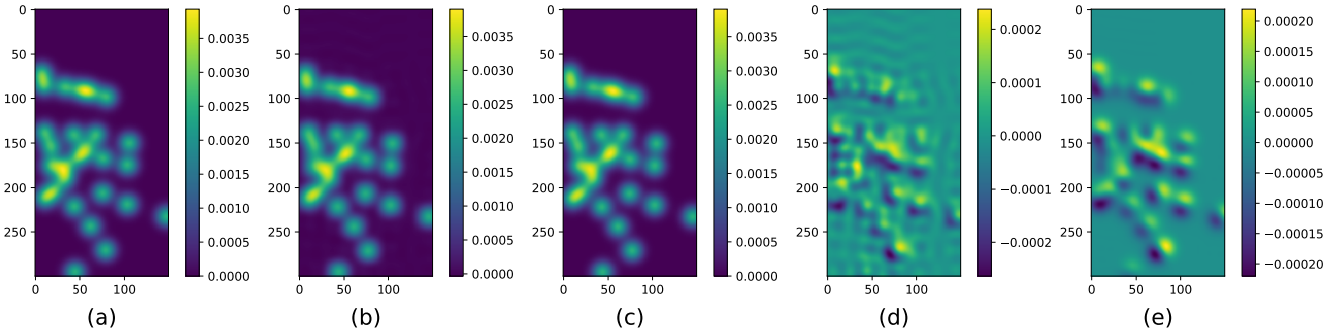


Figure 3. Comparison between the original density map and the reconstructed density map from the characteristic function confined on a small range. (a) the original density map; (b) the reconstructed density map from its characteristic function truncated on $[-0.3, 0.3]^2$, and on (c) $[-0.5, 0.5]^2$; (d) the difference between (a) and (b); (e) the difference between (a) and (c). The reconstructed density map and the original density map are nearly the same. Note the range of difference values in (d) and (e) is much smaller than the range of the density values. This indicates that the characteristic function confined in a small range carries nearly all the information in the spatial domain. Hence, it is appropriate to restrict the integral to a small range when we calculate the chf loss.

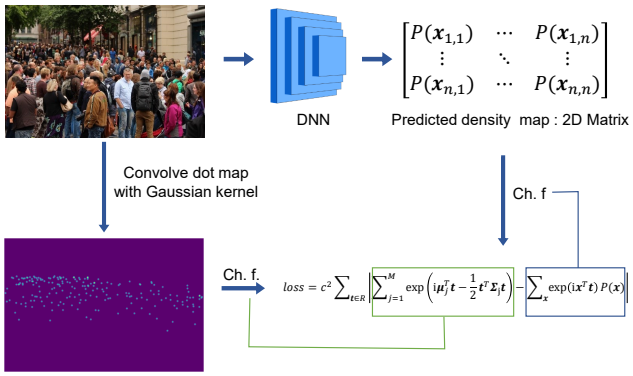


Figure 4. The implementation of our chf loss. The DNN's output is a density map represented as a 2D matrix, where each value $P(\mathbf{x})$ in the matrix corresponds to the singleton measure at a spatial position $\mathbf{x}$. The characteristic functions of the predicted density map is calculated numerically, while the characteristic function of the ground truth density map with Gaussian kernels is directly obtained in closed-form from the annotated positions. The L1 norm between characteristic functions is approximated using a Riemann sum over region $\tilde{R}$, which is based on the point set R.

The evaluation metrics follow the standard convention: the Mean Absolute Error (MAE) and the Root Mean Square Error (MSE) are adopted.

4.2. Comparison of loss functions

First we compare our chf loss with state-of-the-art loss functions in crowd counting in Table 1. All of the loss functions use the same network architecture proposed in [21]. Our chf loss outperforms the other losses on all datasets. Moreover, [37] and [35] require an external Sinkhorn algorithm [22] running dozens of even hundreds of iterations in each training batch, while [34] needs to invert large matrices in each training batch. Nevertheless, the chf loss does not require any other external algorithm, and the calculation can be quickly completed using standard tensor operations.

		NWPU		JHU++		UCF-QNRF		SHTC A		SHTC B	
		MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE
BL [21]	ICCV'19	105.4	454.2	75.0	299.9	88.7	154.8	62.8	101.8	7.7	12.7
NoiseCC [34]	NeurIPS'20	96.9	534.2	67.7	258.5	85.8	150.6	61.9	99.6	7.4	11.3
DM count [37]	NeurIPS'20	88.4	388.6	68.4	283.3	85.6	148.3	59.7	95.7	7.4	11.8
GL [35]	CVPR'21	79.3	346.1	59.9	259.5	84.3	147.5	61.3	95.4	7.3	11.7
ChfL (ours)		**76.8**	**343.0**	**57.0**	**235.7**	**80.3**	**137.6**	**57.5**	**94.3**	**6.9**	**11.0**

Table 1. Comparison with state-of-the-art loss functions. All losses use the same network architecture from [21].

Loss	time / per epoch	time / 500 epochs	number of related hyperparameters
BL [21]	15.2 s	2h 7m	2
NoiseCC [34]	16.4 s	2h 17m	6
DM count [37]	19.0 s	2h 38m	4
GL [35]	17.4 s	2h 25m	7
ChfL (ours)	15.4 s	2h 9m	3

Table 2. Efficiency and number of hyperparameters for different loss functions. The training time is measured using the training set (300 images) of ShanghaiTech A (with batch size 1 and crop size 512). Our implementation uses with *PyTorch* on an *RTX2080 TI*.

Table 2 shows the efficiency comparison among these loss functions. Since they use the same network architecture and the losses are only calculated in the training phase, the identical inference time is omitted here. From the table, BL [21] is the most efficient loss function among them, but BL also has the poorest performance. Our chf loss has 2nd highest efficiency, as well as the 2nd lowest number of hyperparameters, while also achieving best MAE. Note that there are only 300 training images in the timing test, and the efficiency advantage will increase as the training size and number of epochs increase.

4.3. Comparison with SOTA

Table 3 shows the comparison between our chf loss and the current SOTA. For fairness, this comparison only considers methods using a single model and trained on the individual datasets. Although our method is simple, our chf loss is competitive against current SOTA on large-scale datasets, obtaining lowest MAE/MSE on UCF-QNRF,

		NWPU		JHU++		UCF-QNRF		SHTC A		SHTC B	
		MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE
MCNN [46]	CVPR'16	232.5	714.6	188.9	483.4	277.0	426.0	110.2	173.2	26.4	41.3
SwitchCNN [1]	CVPR'17	-	-	-	-	228.0	445.0	90.4	135.0	21.6	33.4
CSRNet [14]	CVPR'18	121.3	387.8	85.9	309.2	110.6	190.1	68.2	115.0	10.6	16.0
SANet [4]	ECCV'18	190.6	491.4	91.1	320.4	-	-	67.0	104.5	8.4	13.6
CAN [18]	CVPR'19	106.3	386.5	100.1	314.0	107	183	62.3	100.0	7.8	12.2
SFCN [40]	CVPR'19	105.7	424.1	77.5	297.6	102.0	171.4	64.8	107.5	7.6	13.0
MBTTBF [28]	ICCV'19	-	-	81.8	299.1	97.5	165.2	60.2	94.1	8.0	15.5
BL [21]	ICCV'19	105.4	454.2	75.0	299.9	88.7	154.8	62.8	101.8	7.7	12.7
KDMG [36]	TPAMI'20	100.5	415.5	69.7	268.3	99.5	173.0	63.8	99.2	7.8	12.7
LSCCNN [24]	TPAMI'20	-	-	112.7	454.4	120.5	218.2	66.5	101.8	7.7	12.7
RPNet [43]	CVPR'20	-	-	-	-	-	-	61.2	96.9	8.1	11.6
AMRNet [19]	ECCV'20	-	-	-	-	86.6	152.2	61.6	98.4	7.0	11.0
NoiseCC [34]	NeurIPS'20	96.9	534.2	67.7	258.5	85.8	150.6	61.9	99.6	7.4	11.3
DM count [37]	NeurIPS'20	88.4	388.6	68.4	283.3	85.6	148.3	59.7	95.7	7.4	11.8
LA-Batch [48]	TPAMI'21	-	-	-	-	113.0	210.0	65.8	103.6	8.6	14.0
AutoScale [42]	IJCV'21	94.1	388.2	65.9	264.8	104.4	174.2	65.8	112.1	8.6	13.9
GL [35]	CVPR'21	79.3	346.1	59.9	259.5	84.3	147.5	61.3	95.4	7.3	11.7
P2PNet [31]	ICCV'21	77.4	362.0	-	-	85.3	154.5	**52.7**	**85.1**	**6.2**	**9.9**
SDA+BL [20]	ICCV'21	-	-	62.6	264.1	83.3	143.1	58.4	95.7	-	-
ChfL (ours)		**76.8**	**343.0**	**57.0**	**235.7**	**80.3**	**137.6**	57.5	94.3	6.9	11.0

Table 3. Comparison with state-of-the-art single-model methods trained on individual data sets.

Algorithm	training time / per epoch	inference time / per epoch	crop size of images in training
KDMG [36]	83.0 s	6.9 s	512
P2PNet [34]	60.8 s	11.8 s	128
ChfL (ours)	15.4 s	6.9 s	512

Table 4. Running time of recent algorithms. The inference time is measured using the test set (182 original images) of ShanghaiTech A. Other settings are the same as in Table 2.

JHU++, and NWPU. Our method also obtains 2nd lowest MAE on ShanghaiTech A and B (behind P2PNet), but these two datasets are smaller and less representative of genearlization ability. These comparative results demonstrate the potential of supervising crowd counting in the frequency domain. We believe that there is also room for improvement for facilitating the development of the crowd counting.

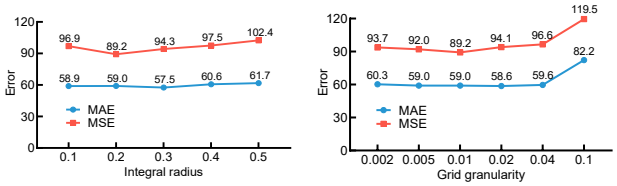
We also compare the efficiency of our method with other recent algorithms in Table 3. Our method is 4x faster than P2PNet (despite P2PNet using smaller image sizes) and 5.4x faster than KDMG in training. For inference, our method has the same running time as KDMG since they use the same architecture, and is $\sim$41% faster than P2PNet.

4.4. Ablation study

The approximation of the integral in the chf loss introduces two extra hyperparameters: the integral range and the grid granularity in the Riemann sum approximation. As mentioned in Section 3.4, Property 4 decouples these two hyperparameters, and thus the ablation study is carried out individually for each hyperparameter on ShanghaiTech A.

Fig. 5a shows the results for different integral ranges. Generally, the counting performance is robust to different integral ranges. When the range is above $[-0.3, 0.3]^2$, the performance gradually degenerates, which suggests that the frequency information beyond this range may make the model overfit. In practice, we fix the range at $[-0.3, 0.3]^2$.

Fig. 5b shows the counting result for different grid granularity. When the granularity is too coarse, i.e., 0.1 granularity, then the error increases significantly. When the granularity is below 0.04, the performance is not too sensitive to the granularity change. Since small granularity means more grids, which corresponds to more memory/computation, we set the granularity as 0.01 in practice.



(a) (b)

Figure 5. Ablation study on (a) the integral range $[-\alpha, \alpha]^2$ where α is the value in the x-axis; (b) the grid granularity in the Riemann sum approximation, where the granularity is the side length of the square grid and the integral range is fixed at $[-0.2, 0.2]^2$.

5. Limitations

By convention the density map is computed as the convolution between the dot map and the Gaussian kernel. Other works have shown that transforming the dot map into a smooth representation is also helpful to make training robust for counting [9, 21, 36, 46]. Indeed, in our framework, the Gaussian kernel acts like a low-pass filter to diminish high-frequency content, which allows for truncation of the integral for implementation. Therefore, convolving the dot map with the Gaussian kernel or other low-pass filter kernel is required in our framework.

6. Conclusions

In this paper, we have studied crowd counting using supervision in the frequency domain. By extending the definition of characteristic function to the density map (finite measures) and proving a series of key properties, we build the foundation of a new paradigm in supervision for training crowd counting models. Based on this foundation, we propose a simple, effective, and efficient method in the form of the chf loss function. The theoretical analysis plays an important role across the spectrum of the method's design, implementation, and hyperparameter selection. We elucidate why our chf loss is effective, through proving that it is an upper-bound to a sup-norm metric between two density maps (over all sub-regions). Experiment results demonstrate its superiority to other SOTA loss functions. We hope that our work will inspire future work on designing loss functions for crowd counting in the frequency domain so as to better exploit the ground-truth information.

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

References

[1] Deepak Babu Sam, Shiv Surya, and R Venkatesh Babu. Switching convolutional neural network for crowd counting. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 5744–5752, 2017. 8

[2] Shuai Bai, Zhiqun He, Yu Qiao, Hanzhe Hu, Wei Wu, and Junjie Yan. Adaptive dilated network with self-correction supervision for counting. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4594–4603, 2020. 3

[3] Patrick Billingsley. *Probability and measure*. John Wiley & Sons, 2008. 3

[4] Xinkun Cao, Zhipeng Wang, Yanyun Zhao, and Fei Su. Scale aggregation network for accurate and efficient crowd counting. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 734–750, 2018. 3, 8

[5] Antoni B Chan, Zhang-Sheng John Liang, and Nuno Vasconcelos. Privacy preserving crowd monitoring: Counting people without people models or tracking. In *2008 IEEE Conference on Computer Vision and Pattern Recognition*, pages 1–7. IEEE, 2008. 2

[6] Prithvijit Chattopadhyay, Ramakrishna Vedantam, Ramprasaath R Selvaraju, Dhruv Batra, and Devi Parikh. Counting everyday objects in everyday scenes. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1135–1144, 2017. 2

[7] Ke Chen, Shaogang Gong, Tao Xiang, and Chen Change Loy. Cumulative attribute space for age and crowd density estimation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2467–2474, 2013. 2

[8] Weina Ge and Robert T Collins. Marked point processes for crowd counting. In *2009 IEEE Conference on Computer Vision and Pattern Recognition*, pages 2913–2920. IEEE, 2009. 2

[9] Haroon Idrees, Muhmmad Tayyab, Kishan Athrey, Dong Zhang, Somaya Al-Maadeed, Nasir Rajpoot, and Mubarak Shah. Composition loss for counting, density map estimation and localization in dense crowds. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 532–546, 2018. 6, 8

[10] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014. 6

[11] Harold W Kuhn. The hungarian method for the assignment problem. *Naval research logistics quarterly*, 2(1-2):83–97, 1955. 1, 2

[12] Victor Lempitsky and Andrew Zisserman. Learning to count objects in images. *Advances in neural information processing systems*, 23:1324–1332, 2010. 1, 2, 5

[13] Min Li, Zhaoxiang Zhang, Kaiqi Huang, and Tieniu Tan. Estimating the number of people in crowded scenes by mid based foreground segmentation and head-shoulder detection. In *2008 19th international conference on pattern recognition*, pages 1–4. IEEE, 2008. 2

[14] Yuhong Li, Xiaofan Zhang, and Deming Chen. Csrnet: Dilated convolutional neural networks for understanding the highly congested scenes. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1091–1100, 2018. 3, 8

[15] Sheng-Fuu Lin, Jaw-Yeh Chen, and Hung-Xin Chao. Estimation of number of people in crowded scenes using perspective transformation. *IEEE Transactions on Systems, Man, and Cybernetics-Part A: Systems and Humans*, 31(6):645–654, 2001. 2

[16] Bo Liu and Nuno Vasconcelos. Bayesian model adaptation for crowd counts. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 4175–4183, 2015. 2

[17] Lingbo Liu, Hongjun Wang, Guanbin Li, Wanli Ouyang, and Liang Lin. Crowd counting using deep recurrent spatial-aware network. *arXiv preprint arXiv:1807.00601*, 2018. 3

[18] Weizhe Liu, Mathieu Salzmann, and Pascal Fua. Context-aware crowd counting. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5099–5108, 2019. 3, 8

[19] Xiyang Liu, Jie Yang, Wenrui Ding, Tieqiang Wang, Zhijin Wang, and Junjun Xiong. Adaptive mixture regression network with local counting map for crowd counting. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XXIV 16*, pages 241–257. Springer, 2020. 8

[20] Zhiheng Ma, Xiaopeng Hong, Xing Wei, Yunfeng Qiu, and Yihong Gong. Towards a universal model for cross-dataset crowd counting. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 3205–3214, 2021. 8

[21] Zhiheng Ma, Xing Wei, Xiaopeng Hong, and Yihong Gong. Bayesian loss for crowd count estimation with point supervision. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 6142–6151, 2019. 1, 3, 4, 6, 7, 8

[22] 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. 1, 2, 7

[23] David Ryan, Simon Denman, Clinton Fookes, and Sridha Sridharan. Crowd counting using multiple local features. In *2009 Digital Image Computing: Techniques and Applications*, pages 81–88. IEEE, 2009. 2

[24] Deepak Babu Sam, Skand Vishwanath Peri, Mukuntha Narayanan Sundararaman, Amogh Kamath, and Venkatesh Babu Radhakrishnan. Locate, size and count: Accurately resolving people in dense crowds via detection. *IEEE transactions on pattern analysis and machine intelligence*, 2020. 8

[25] Chong Shang, Haizhou Ai, and Bo Bai. End-to-end crowd counting via joint learning local and global count. In *2016 IEEE International Conference on Image Processing (ICIP)*, pages 1215–1219. IEEE, 2016. 2

[26] Zan Shen, Yi Xu, Bingbing Ni, Minsi Wang, Jianguo Hu, and Xiaokang Yang. Crowd counting via adversarial cross-scale consistency pursuit. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 5245–5254, 2018. 3

[27] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*, 2014. 6

[28] Vishwanath A Sindagi and Vishal M Patel. Multi-level bottom-top and top-bottom feature fusion for crowd counting. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 1002–1012, 2019. 8

[29] Vishwanath A Sindagi, Rajeev Yasarla, and Vishal M Patel. Pushing the frontiers of unconstrained crowd counting: New dataset and benchmark method. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 1221–1231, 2019. 6

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

[31] Qingyu Song, Changan Wang, Zhengkai Jiang, Yabiao Wang, Ying Tai, Chengjie Wang, Jilin Li, Feiyue Huang, and Yang Wu. Rethinking counting and localization in crowds: A purely point-based framework. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 3365–3374, 2021. 1, 3, 8

[32] Russell Stewart, Mykhaylo Andriluka, and Andrew Y Ng. End-to-end people detection in crowded scenes. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2325–2333, 2016. 2

[33] S. J. Taylor. Set functions. In *Introduction to Measure and Integration*, page 51–73. Cambridge University Press, Cambridge, 1973. 3

[34] Jia Wan and Antoni Chan. Modeling noisy annotations for crowd counting. *Advances in Neural Information Processing Systems*, 33, 2020. 3, 6, 7, 8

[35] Jia Wan, Ziquan Liu, and Antoni B Chan. A generalized loss function for crowd counting and localization. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 1974–1983, 2021. 1, 2, 3, 6, 7, 8

[36] Jia Wan, Qingzhong Wang, and Antoni B Chan. Kernel-based density map generation for dense object counting. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2020. 8

[37] Boyu Wang, Huidong Liu, Dimitris Samaras, and Minh Hoai. Distribution matching for crowd counting. *arXiv preprint arXiv:2009.13077*, 2020. 1, 2, 3, 4, 6, 7, 8

[38] Chuan Wang, Hua Zhang, Liang Yang, Si Liu, and Xiaochun Cao. Deep people counting in extremely dense crowds. In *Proceedings of the 23rd ACM international conference on Multimedia*, pages 1299–1302, 2015. 2

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

[40] Qi Wang, Junyu Gao, Wei Lin, and Yuan Yuan. Learning from synthetic data for crowd counting in the wild. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8198–8207, 2019. 8

[41] Xinlong Wang, Tete Xiao, Yuning Jiang, Shuai Shao, Jian Sun, and Chunhua Shen. Repulsion loss: Detecting pedestrians in a crowd. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 7774–7783, 2018. 2

[42] Chenfeng Xu, Dingkang Liang, Yongchao Xu, Song Bai, Wei Zhan, Xiang Bai, and Masayoshi Tomizuka. Autoscale: Learning to scale for crowd counting and localization. *arXiv preprint arXiv:1912.09632*, 2019. 8

[43] Yifan Yang, Guorong Li, Zhe Wu, Li Su, Qingming Huang, and Nicu Sebe. Reverse perspective network for perspective-aware object counting. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4374–4383, 2020. 3, 8

[44] Anran Zhang, Jiayi Shen, Zehao Xiao, Fan Zhu, Xiantong Zhen, Xianbin Cao, and Ling Shao. Relational attention network for crowd counting. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 6788–6797, 2019. 3

[45] Shifeng Zhang, Longyin Wen, Xiao Bian, Zhen Lei, and Stan Z Li. Occlusion-aware r-cnn: Detecting pedestrians in a crowd. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 637–653, 2018. 2

[46] Yingying Zhang, Desen Zhou, Siqin Chen, Shenghua Gao, and Yi Ma. Single-image crowd counting via multi-column convolutional neural network. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 589–597, 2016. 1, 2, 6, 8

[47] Tao Zhao and Ramakant Nevatia. Bayesian human segmentation in crowded situations. In *2003 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 2003. Proceedings.*, volume 2, pages II–459. IEEE, 2003. 2

[48] Joey Tianyi Zhou, Le Zhang, Du Jiawei, Xi Peng, Zhiwen Fang, Zhe Xiao, and Hongyuan Zhu. Locality-aware crowd counting. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2021. 8