Hybrid Perturbation Strategy for Semi-Supervised Crowd Counting

Xin Wang, *Graduate Student Member, IEEE*, Yue Zhan, Yang Zhao, Tangwen Yang, *Member, IEEE*, and Qiuqi Ruan, *Life Senior Member, IEEE*

***Abstract*— A simple yet effective semi-supervised method is proposed in this paper based on consistency regularization for crowd counting, and a hybrid perturbation strategy is used to generate strong, diverse perturbations, and enhance unlabeled images information mining. The conventional CNN-based counting methods are sensitive to texture perturbation and imperceptible noises raised by adversarial attack, therefore, the hybrid strategy is proposed to combine a spatial texture transformation and an adversarial perturbation module to perturb the unlabeled data in the semantic and non-semantic spaces, respectively. Moreover, a cross-distribution normalization technique is introduced to address the model optimization failure caused by BN layer in the strong perturbation, and to stabilize the optimization of the learning model. Extensive experiments have been conducted on the datasets of ShanghaiTech, UCF-QNRF, NWPU-Crowd, and JHU-Crowd++. The results demonstrate that the proposed semi-supervised counting method performs better over the state-of-the-art methods, and it shows better robustness to various perturbations.**

***Index Terms*— Crowd counting, semi-supervised learning, hybrid perturbation strategy, semantic and non-semantic perturbation, cross-distribution normalization.**

I. Introduction

CROWD counting aims to estimate people number and crowd density distribution in an image, which is generally formulated as the estimation of crowd density map [1]. Its ground truth is obtained by performing Gaussian kernel convolution on head location. The supervised learning methods [2], [3], [4], [5], [6] are frequently used in crowd counting, but it is time-consuming to manually label the people locations in images, especially for the images with thousands of people. Besides, we observe that most of the existing models work well only on the test dataset similar to training dataset. However, a real scene of varying scale, occlusion, nonuniform distribution and background clutter, may be quite different from training dataset. That is to say, new images need to be added and labeled manually when a model is used for new scenes. However, the labeling cost limits its application. It is necessary to reduce the tedious annotation work and to improve the efficiency of learning model with limited data. To this end, the synthetic data were used to train a counting model in [7]. However, the distribution shift between the synthetic and real data degrades the model performance in the real crowd scene. Therefore, the semi-supervised crowd counting (SSCC) methods [8], [9], [10], [11], [12], [13] utilize a large number of unlabeled data to train counting model, and surrogate tasks are introduced to leverage the insightful information of unlabeled data.

Semi-supervised learning (SSL) methods [14], [15] have been proposed to learn a model from unlabeled data under smoothness assumption, cluster assumption and manifold assumption. Among them, the consistency regularization based SSL methods have shown promising results in image classification [16], [17], [18] and semantic segmentation [19], [20]. They force a model to produce consistent predictions for the different augmented views of same input, so that the model can learn more generalized features from unlabeled data and not to overfit the limited labeled data. The density-based crowd counting is a pixel-level regression task, and is suitable for the consistency-based SSL framework under smoothness assumption. We tested the consistency-based SSL methods [16], [17], [18] to SSCC task, but obtained poor performance. Details will be presented in Section V. The main reason is that the data augmentations utilized in MT [16], VAT [17], UDA [18] are inadequate and unsuitable for the crowd counting task, but we know in [21], [22], and [23] that high-quality data augmentation is able to improve the performance of SSL. Therefore, a suitable perturbation strategy for SSCC will be studied in this paper.

Intuitively, unlabeled data can be perturbed by strong data augmentations of fully supervised learning [24], [25], [26], [27], [28]. However, these augmentations are not suitable perturbations for SSCC, because they are designed for image classification tasks and [26], [27] requires high computational cost. If prior knowledge about perturbations in the counting model is available, we can design a more efficient perturbation strategy for SSCC. This strategy can generate strong perturbation and reduce the computational costs, because it

Manuscript received 16 December 2021; revised 4 June 2023; accepted 21 January 2024. Date of publication 8 February 2024; date of current version 12 February 2024. This work was supported in part by the State's Key Project of Research and Development Plan under Grant 2022YFC3005200 and in part by the National Natural Science Foundation of China under Grant 62201355. The associate editor coordinating the review of this manuscript and approving it for publication was Dr. Benoit M. Macq. *(Corresponding author: Tangwen Yang.)*

Xin Wang, Yue Zhan, Tangwen Yang, and Qiuqi Ruan are with the Institute of Information Science and the Beijing Key Laboratory of Advanced Information Science and Network Technology, School of Computer and Information Technology, Beijing Jiaotong University, Beijing 100044, China (e-mail: 19112007@bjtu.edu.cn; 21112011@bjtu.edu.cn; twyang@bjtu.edu.cn; qqruan@bjtu.edu.cn).

Yang Zhao is with the College of Computer Science and Software Engineering, Shenzhen University, Shenzhen 518160, China (e-mail: zhaoyang1990@szu.edu.cn).

Digital Object Identifier 10.1109/TIP.2024.3361730

removes augmentations in the combined transformations [26], [27], [28] that are irrelevant to crowd counting. An effective perturbation should produce a significant consistency loss on unlabeled data. To acquire prior knowledge of crowd counting, a simple yet effective way is to analyze the robustness of the counting model against various perturbations. By doing this, we can know the perturbations to which the counting model is more vulnerable.

In this paper, the robustness analysis of the counting model is done with semantic and non-semantic perturbations. Their discrimination is to see whether their visual semantics are comprehensible. More details will be discussed in Section III, but we present the obtained prior knowledge of crowd counting in the following. 1) Counting model is sensitive to texture transformation in the crowd area and relatively robust to geometric transformation. 2) Counting model is sensitive to adversarial perturbations in the non-semantic space. This explains why we perturb the unlabeled crowd images in the semantic and non-semantic spaces, and the semantic perturbations are obtained by texture transformation while the non-semantic perturbations are generated by adversarial perturbation. Meanwhile, we observed that when semantic perturbations are too strong, counting model is difficult to converge due to Batch Normalization (BN) layer [29]. In the SSL, a mini-batch comprises labeled and unlabeled data. Perturbing unlabeled data may lead to a distinct sample distribution compared to labeled data, resulting in a mixture of distributions within the mini-batch. However, standard BN assumes that the samples in a batch follow the same data distribution. Furthermore, the inappropriate parameter updating mechanism during the training of teacher-student architecture [16], is prone to cause a mismatch between the BN statistics and the model weights.

Consequently, a novel hybrid perturbation strategy is proposed here with a cross-distributions normalization technique. It improves the learning performance of the counting model under a semi-supervised setting. Specifically, the hybrid perturbation strategy consists of spatial texture transformation (STT) for semantic perturbations (SP) and adversarial perturbation for non-semantic perturbations (NSP). STT draws inspiration from the first result 1) and performs strong texture transformation for the crowd foreground region and conflict texture transfer (CTT) for the background region, respectively. To reduce computation load, the texture transformation of the crowd foreground area is restricted to color jitter. Conflict texture information in CTT is added into the background region via a mixup-based technique. It can generate strong enough perturbation for the background. To promote perturbation diversity, we extend the perturbation space from the semantic to non-semantic space, where an unlabeled image is perturbed by adversarial perturbation. It iteratively explores incomprehensible noises within the ε-ball centered on an unlabeled image. These noises maximize the consistency loss between the unlabeled image and its perturbed version. In each iteration, the direction of noises varies, compensating for semantic perturbations that typically follow the same direction of data distribution [21]. Furthermore, a cross-distribution normalization (CDN) method is presented to address model optimization failure caused by the BN layer in the strong perturbation.

The contributions of this paper are summarized below.

- A new perspective to explore the prior knowledge of crowd counting is provided by studying the model robustness to various perturbations. With the prior of which perturbations the counting model is sensitive to, we can design more effective perturbations for SSCC.
- A hybrid perturbation strategy is proposed for SSCC under the consistency regularization framework. Semantic and non-semantic perturbations are put into unlabeled data with spatial texture transformation and adversarial perturbation.
- A cross-distribution normalization technique is introduced to the counting framework to address the model optimization failure caused by BN layer in the strong perturbation, and to stabilize the training of SSCC as well.
- The proposed method achieves leading performance on four crowd counting datasets and is more robust to various perturbations than other counting methods.

II. Related Works

The supervised methods have been used in crowd counting and achieved amazing performance on labeled datasets, while the semi-supervised crowd counting method is rarely investigated, particularly to the limited labeled datasets.

A. Fully Supervised Crowd Counting

Supervised methods have been proposed to the challenging issues of crowd counting. For example, [2], [30], [31], [32], [33], [34] focus on the various scale of people or head in images. To extract the multi-scale features, multi-columns networks are introduced in [2], [31], and [35]. However, multi-columns networks with bloated structures are prone to generate redundant information. Hence, the scale-aware modules with different receptive fields are used to improve the counting model performance [3], [30], [33]. Background clutter is also a challenging issue in crowd counting. To attend the useful information in the crowd rather than the background, attention mechanism is frequently used in the counting network [34], [36]. These methods have achieved good results on labeled dataset, but the cost of annotation limits their applications in the real scenes.

B. Semi-Supervised Crowd Counting

Semi-supervised learning has recently attracted a lot of attention. With limited labeled data, semi-supervised learning was used for crowd counting [8], [10], [11], [13], [37]. For example, Liu et al. [10] introduced self-supervised learning to obtain a strong feature extractor from unlabeled data by solving the ranking of unlabeled data, improving the performance of SSCC. Sindagi et al. [11] proposed a Gaussian process-based iterative learning mechanism to estimate the pseudo-ground truth for unlabeled data, and then used the supervised information to train the counting network. Liu et al. [8] designed a novel self-training strategy to improve the performance of SSCC by using the relationship of binary segmentation tasks. The noises in pseudo density maps may cause model degradation. Thus, Meng et al. [9] proposed a spatial uncertainty-aware teacher-student framework to estimate the spatial uncertainty maps with regularized surrogate tasks, alleviating the negative effect of inaccurate pseudo targets. Also, Zhu et al. [13] generated credible pseudo density maps by leveraging the correlation of multi-tasks, including density regression, binary segmentation, and confidence prediction. Previous methods mainly focus on how to construct auxiliary tasks and reduce the negative impact of noises in pseudo targets. However, their performance is not superior to fully supervised methods. Differently, we design a strong and diverse perturbation strategy to improve SSCC under the consistency-based SSL framework. Our method shows better performance on various crowd counting datasets.

C. Perturbation in Consistency-Based SSL

Perturbation is important in consistency-based SSL. Since the proposed HPS is performed on input images, we only discuss similar works. Early work [16] used additive Gaussian noises and simple image transformations such as flip and rotation to augment unlabeled data, and only obtained sub-optimal performance. The works [38], [39], [40] found that advanced data augmentation is useful for SSL. Mixmatch [38] employed the Mixup [24] to mix the images and their labels, and to improve the SSL. EnAET [22] proposed an AutoEncoding Transformation framework to learn a strong encoder under the ensemble of spatial and non-spatial transformations in a self-supervised manner. The learned discriminative feature representation can improve the performance of Mixmatch [38]. UDA [40] used a strong augmentation Randaugment [28] to perturb unlabeled data, obtaining a large performance improvement. During the same period, ReMixmatch [39] designed a strong augmentation CTAugment and an augmentation anchoring strategy to further improve SSL. FixMatch [41] used the combination of RandAugment [28], CTAugment [39] and Cutout [42] to apply aggressive perturbations to unlabeled data, and achieved excellent performance. In addition, the latest work CLSA [23] discussed the role of stronger augmentation in representation learning. The stronger augmentation is constructed by randomly combining 14 image transformations with random strengths. The above methods generate strong and diverse perturbations by combining as many image transformations as possible. Furthermore, the idea of adversarial training [43], [44] has been integrated into consistency-based SSL. The training of SSL is formulated as a min-max optimization. Miyato et al. [17] designed virtual adversarial training (VAT) for SSL, which used adversarial noises to perturb unlabeled data and minimized KL divergence between the output distributions of original data and perturbed data. However, VAT is sensitive to hyper-parameters and difficult to converge in SSCC. The reason may be that the optimization algorithm in VAT is not suitable and thus influences the generation of adversarial noise. Overall, the above perturbations are designed for image classification, and they are not suitable for SSCC due to differences in task data distributions. Hence, the hybrid perturbation strategy (HPS) is designed for SSCC based on the prior knowledge of the counting model. HPS produces strong and diverse perturbations in the semantic and non-semantic spaces. To our knowledge, we are the first to design a dedicated strong perturbation strategy to improve the performance of SSCC.

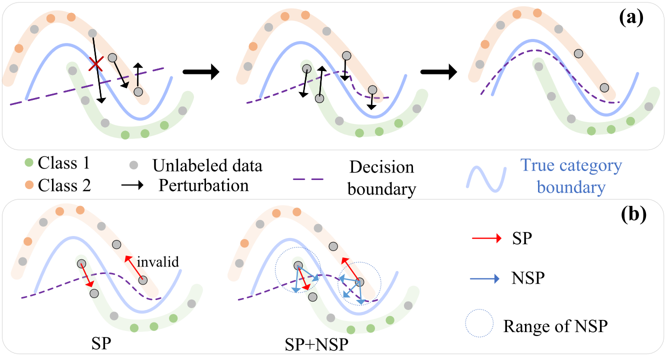


Fig. 1. The working mechanism of perturbation in SSL. (a) shows how perturbation brings the decision boundary closer to the true boundary for a binary classification task. (b) compares the difference between SP and NSP during SSL training.

III. Motivation

The objective of consistency-based SSL is to minimize the following loss:

$$\mathcal{L} = \mathcal{L}_s(X^l, Y^l) + \lambda_u \mathcal{L}_u(\mathcal{T}_1(X^u), \mathcal{T}_2(X^u)), \tag{1}$$

where $\mathcal{L}_s$ denotes the supervised loss on labeled data $\mathcal{D}_l = \{X_i^l, Y_i^l\}_{i=1}^{|\mathcal{D}_l|}$. $\mathcal{T}_1, \mathcal{T}_2$ are two transformations on a perturbation set. $\mathcal{L}_u$ is the consistency loss of the model predictions for the two transformations of unlabeled data $\mathcal{D}_u = \{X_i^u\}_{i=1}^{|\mathcal{D}_u|}$. If $\mathcal{T}_1$ and $\mathcal{T}_2$ are close to each other and weak, the perturbed images are similar, leading to a small $\mathcal{L}_u$, and the unlabeled data $\mathcal{D}_u$ may not be fully exploited. Fig. 1 (a) illustrates how a perturbation contributes to the consistency-based SSL. It can be seen that an effective perturbation must move a sample from one side of the decision boundary to the other, generating enough consistency loss to optimize the decision boundary. But, the perturbation can not cross the true category boundary, because it may destroy the label of the original sample. Therefore, how to generate strong, diverse, and reasonable perturbations is critical for SSCC. The prior knowledge of crowd counting can provide a solution to this question. To this end, we first acquire the prior knowledge of which perturbations the counting model is more vulnerable to, and then use the prior knowledge to design the perturbations for SSCC.

The perturbations to an image can be divided into semantic perturbation (SP) and non-semantic perturbation (NSP). SP refers to image transformations. Each changes a specific semantic of an image. NSP aims to add incomprehensible noises to the input image, such as random isotropic Gaussian noises and adversarial noises. To study the robustness of a counting model against various semantic and non-semantic perturbations, motivated experiments were carried out. The transformations of SP used in our experiments are described

TABLE I

IMAGE TRANSFORMATION OPERATION AND MAGNITUDE SETTING. FOLLOWING [28], THE NUMBER OF TRANSFORMATIONS, N, IS RANDOMLY CHOSEN FROM THE SET TO PERTURB EACH IMAGE AND STUDY THE ROBUSTNESS OF THE COUNTING MODEL AGAINST SP

Type	Operation (magnitude range)

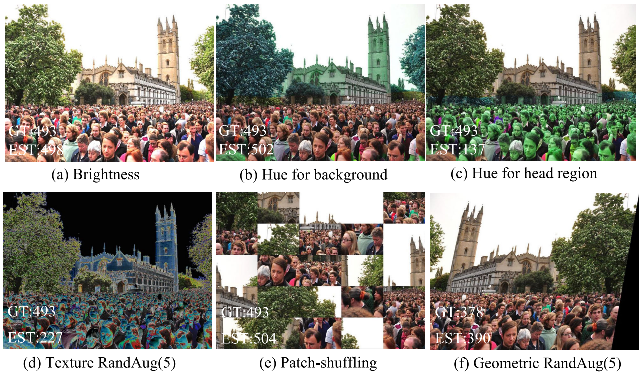
Texture transformations	Posterize: [0,4]; Solarize: [0,110];
	Contrast: [0.1,0.9]; Color: [0.1,1.9];
	Brightness: [0.1,1.9]; Sharpness: [0.1,1.9];
	AutoContrast: *None*; Equalize: *None*.

Geometric transformations	ShearX(Y): [0,0.3]; TranslateX(Y): [0,100]; Rotate: [0,30]; Flip: *None*.

TABLE II

THE PERFORMANCE OF THE CSRNET ON THE TEST SET OF SHHA UNDER DIFFERENT SEMANTIC TRANSFORMATIONS, AVERAGED OVER **FIVE** RUNS. PL REFERS TO THE PERTURBATION LEVEL I.E., THE NUMBER OF TRANSFORMATIONS TO EACH IMAGE. THE PIXELS LARGER THAN 0 IN THE CROWD DENSITY MAP IS CONSIDERED TO BE THE FOREGROUND, WHILE THE REST IS THE BACKGROUND

	Texture transformations						Geometric	
	All		Foreground		Background		All	
PL	MAE	RMSE	MAE	RMSE	MAE	RMSE	MAE	RMSE
N=0	61.1	93.3	61.1	93.3	61.1	93.3	61.1	93.3
N=1	80.1	142.4	77.2	129.5	60.9	93.4	62.6	95.4
N=3	95.2	164.8	101.3	174.6	60.7	92.3	68.4	114.9
N=5	122.5	218.5	137.0	254.8	61.1	92.5	68.8	111.9
N=7	133.2	241.9	154.9	257.5	60.1	90.8	70.2	111.8



(a) Brightness (b) Hue for background (c) Hue for head region

(d) Texture RandAug(5) (e) Patch-shuffling (f) Geometric RandAug(5)

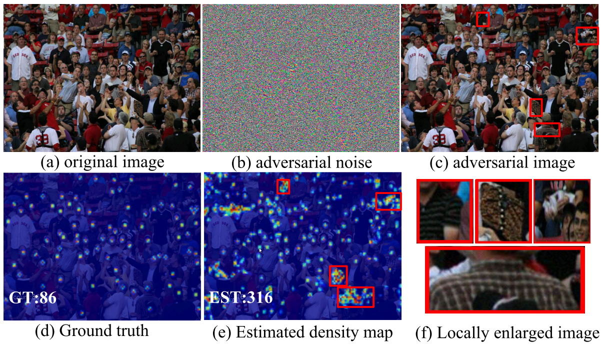
Fig. 2. Visualization of semantic perturbations for a crowd image, including the images (a)-(d) of texture transformations and the images (e)-(f) of geometric transformations. (a)-(c) apply a single texture transformation each, whereas (d) and (f) are the results obtained from texture combination and geometric combination, respectively. The value in $(\cdot)$ is the number of transformations within a combination. (e) shows the result of a 5×5 patch-shuffling. The lower left of each image shows the ground truth and estimated count of individuals.

in Table I. The performance of the counting model under different strengths of SP is evaluated on SHHA testing dataset, and the results are reported in Table II. We can find that the model performance undergoes a significant degradation (e.g., MAE increases from 61.1 to 133.2), when strong texture transformations ($\mathrm{N} = 7$) are made to input images. For strong geometric transformations, MAE increases slightly. Fig. 2 shows the images perturbed by SP (texture and geometric transformations) and the counting results. The texture changes in an image, particularly the hue of color, may cause the counting model to fail. What is more, the counting model appears to be more vulnerable to the texture perturbations in the head region, as shown in Fig. 2 (b)-(c). The experimental results show counting model is sensitive to texture perturbations in the crowd region and relatively robust to geometric perturbations. Therefore, we should pay more attention to the strong texture perturbation and the difference between the crowd and non-crowd regions under SSL setting. To this end, we construct a spatial texture transformation module in this study.

TABLE III

THE ROBUSTNESS OF THE COUNTING MODEL UNDER ADVERSARIAL ATTACKS (FGSM [43], PGD [45]). ϵ DENOTES THE PERTURBATION BUDGETS. K IS THE STEP SIZES OF PGD

Method	MAE	RMSE
None	61.1	93.3
FGSM($\epsilon = 2/255$)	172.1	249.6
FGSM($\epsilon = 4/255$)	302.3	430.5
PGD($\epsilon = 2/255$,K=2)	214.7	299.9
PGD($\epsilon = 2/255$,K=5)	293.8	391.4
PGD($\epsilon = 4/255$,K=5)	1127.8	1378.5



(a) original image (b) adversarial noise (c) adversarial image

(d) Ground truth (e) Estimated density map (f) Locally enlarged image

Fig. 3. Visualization of adversarial perturbation for a crowd image. The FGSM [43] with $\epsilon = 2/255$ is used to generate adversarial noise. The counting model produces incorrect results for highly textured regions of the adversarial image, such as plaid shirts and backpacks.

At the same time, we evaluate the robustness of counting model against untargeted adversarial perturbations (e.g., FGSM [43], PGD [45]), which generate the imperceptible noise of loss maximization through an optimization algorithm. The experimental results are given in Table III and Fig. 3. No surprise, the counting model is vulnerable to gradient-based adversarial perturbations. Moreover, it can be seen from Fig. 3 that the counting model overestimates in the highly textured regions of the perturbed crowd image. It means that the adversarial attack method may manipulate the texture information of a crowd image, as discussed previously. Thus, the non-semantic noises generated by adversarial attacks can be used as a strong perturbation for SSCC. Unlike semantic perturbation, adversarial perturbation is able to generate more diverse perturbations for unlabeled data, because it is generated in a larger perturbation space.

Based on these understandings, a hybrid perturbation strategy (HPS) is proposed to increase the diversity and strength of perturbations on unlabeled data. It is composed of the semantic perturbation of spatial texture transformation and the non-semantic perturbation of adversarial perturbation. Generally, SP is performed along the underlying data distribution of the same class [21], while NSP perturbs the input image along multiple directions. If only SP is used, some perturbed samples will not cross the decision boundary, making them

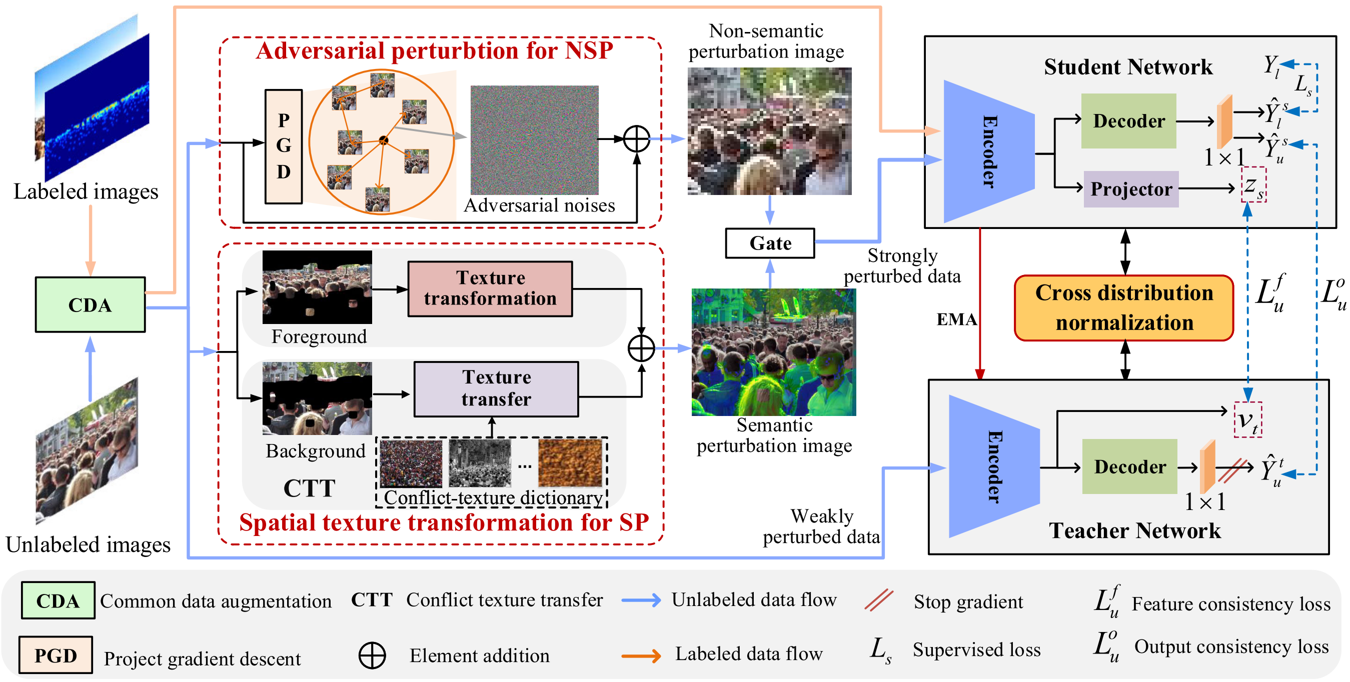


Fig. 4. The framework of the proposed semi-supervised crowd counting method. It mainly contains spatial texture transformation for semantics perturbation (SP), adversarial perturbation for non-semantic perturbation (NSP), and cross-distribution normalization (CDN). SP adopts a spatial texture transformation to perturb unlabeled images in the semantic space. NSP generates small yet efficient perturbations for unlabeled images via adversarial perturbations. CDN uses different BN modules for labeled and unlabeled images, and introduces a bidirectional statistics update mechanism to achieve accurate population statistics. CDA generates basic augmentations on all images, including random horizontal flipping, random cropping, and multi-scale training.

ineffective for model training, as shown in the first subplot in Fig. 1 (b). When SP and NSP are used together, the probability of samples crossing the decision boundary increases because they are perturbed in multiple effective directions, as shown in the second subplot in Fig. 1 (b).

IV. Proposed Method

A. Overview

Fig. 4 shows the overall framework of the proposed semi-supervised crowd counting method. It is based on the meanTeacher [16], including student network and teacher network. They have the same model architecture, and the teacher's parameters are updated using the Exponential Moving Average (EMA) of the student's parameters. Classical data augmentations, including random flip and multi-scale, are first used to generate weak perturbations on labeled and unlabeled images. Then, the weakly-perturbed unlabeled images are perturbed again by the proposed hybrid perturbation strategy to obtain strongly-perturbed images. The weakly-perturbed labeled images and the strongly-perturbed unlabeled images are input to the student, but the teacher only uses the weakly-perturbed unlabeled images as inputs, as done in [41]. This is because the predictions from the teacher are used as pseudo-labels for the student, and the strong unlabeled images may reduce the prediction quality of the teacher. Moreover, following BYOL [46], a latent feature constraint loss is used to avoid small consistency loss. It minimizes the similarity loss between the teacher representation v_t and the student projection z_s. A cross-distribution normalization technique is introduced to handle the data distribution shift between labeled and unlabeled images as well as the shift between the student and the teacher. In the framework, the student weights are updated with the supervised loss $\mathcal{L}_s$, the latent feature constraint loss $\mathcal{L}_u^f$, and the consistency loss of the model output $\mathcal{L}_u^o$. In the teacher network, each iteration the parameters (weights and BN statistics) are the EMA of the student network [47]. To alleviate the mismatch between the BN statistics and the learned student model, the statistics computed in the teacher are transferred to the student by the cross-distribution normalization module. During inference, the student is used to predict the crowd density maps of input images. Overall, the framework includes semantic perturbation on spatial texture transformation, non-semantic perturbation on adversarial perturbation and CDN modules. The details are presented below.

B. Spatial Texture Transformation for Semantic Perturbation

As discussed previously, we find that counting model is vulnerable to the texture perturbations of crowd area. Therefore, a spatial texture transformation (STT) is designed here to provide different perturbations on the different areas of input images. STT consists of a spatial mask generation (SMG) module and a hybrid texture transformation (HTT) module. They are given below.

1) Spatial Mask Generation: SMG generates the spatial mask of an unlabeled image. It is used to determine which pixels belong to crowd or background area. First, the teacher $F^t(\cdot;\theta^t)$ with parameters θ^t is used to estimate the pseudo density map $Y_u^t = F^t(X_u;\theta^t)$ of an unlabeled image X_u.

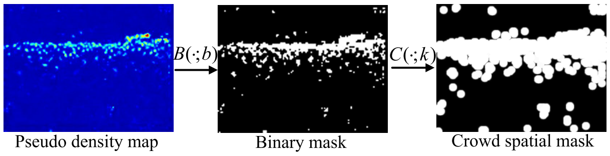


Fig. 5. Illustration of spatial mask generation. b and k are set to 0.01 and 40, respectively.

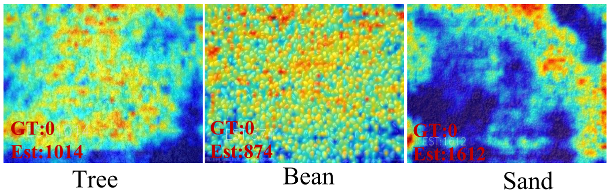


Fig. 6. Predicted density maps of the counting model for background images with significant textural features. It can be seen that the counting model incorrectly predicts a large number of people from the background images.

Then, Y_u^t is converted to a binary mask $B(Y_u^t; b)$ using a binary function $B(\cdot; b)$ with a threshold b. The mask is further refined through a morphology operation $C(\cdot; k)$. Here, k denotes the kernel size of the morphological operation. Formally, let SMG be $G(\cdot; \theta^t, b, k)$, then the spatial mask M_u of X_u can be written by

$$M_u = G(X_u; \theta^t, b, k) = C(B(F^t(X_u; \theta^t); b); k). \quad (2)$$

where $F(\cdot; \theta^t)$, $B(\cdot; b)$, $C(\cdot; k)$ are three functions to generate the spatial mask, namely, pseudo density map estimation, density map binarization, and binary mask post-processing, as shown in Fig. 5. During training, the crowd area of an image is obtained by setting the pixel values of the density map above the threshold b to 1 and the rest pixels to 0. The crowd area in the binary mask is small and reduces the influence of perturbation on SSCC. Therefore, a dilation operation with an appropriate kernel size k is performed to expand the size of the crowd area.

2) Hybrid Texture Transformation: CNN is biased to texture and is prone to lose global information [48]. Thus, the CNN-based crowd counting model is sensitive to the local pattern changes in the important image area, as discussed in Section III. This motivates us to consider texture transformation as a strong perturbation of the counting model. Hence, HTT is designed here to perturb the crowd and background areas of unlabeled images in the semantic space. Let the SP function be $T_{sp} : \mathcal{I} \times (\mathcal{P}_{\text{sp}}^1 \times \cdots \times \mathcal{P}_{\text{sp}}^q) \rightarrow \mathcal{I}$. $\mathcal{I}$ is the image space and $\mathcal{P}_{sp}^i$ is the feasible operation set of the i-th perturbation. q denotes the number of perturbations. Given an unlabeled image X_u, the semantic perturbation space $\Omega_{\text{sp}}(X_u)$ is defined by

$$\Omega_{\text{sp}}(X_u) = \{T_{\text{sp}}(X_u; P_{\phi^i}^i); P_{\phi^i}^i \in \mathcal{P}_{\text{sp}}^j, j = 1, 2, \cdots, q\}, \quad (3)$$

where $P_{\phi^i}^i$ is a specific operation in $\mathcal{P}_{\text{sp}}^i$. ϕ^i is the hyperparameters of $\mathcal{P}_{\text{sp}}^i$. Generally, the perturbations of the crowd foreground and background areas are different, and thus $\Omega_{\text{sp}}(X_u)$ is resolved to the foreground space $\Omega_{\text{sp}}^f(X_u)$ and the background space $\Omega_{\text{sp}}^b(X_u)$.

Foreground perturbation: $\Omega_{\text{sp}}^f(X_u)$ is determined based on the prior knowledge in Sec. III. The perturbation in the image foreground is used to change the texture information of crowd heads while preserving their original labels on X_u. During training, the crowd areas of unlabeled images are perturbed online by T_{sp}^f to obtain strong perturbations. First, T_{sp}^f generates iteratively a set of random parameters $\Phi^f = \{\phi^i\}_{i=1}^q$. Then, a perturbation list $\{P_{\phi^i}^i\}_{i=1}^q$ is constructed. Finally, the unlabeled image is perturbed by $T_{\text{sp}}^f = P_{\phi^1}^1 \circ P_{\phi^2}^2 \cdots \circ P_{\phi^q}^q$. To expand the semantic space, the order of each perturbation in T_{sp}^f is random. As mentioned previously, color-based transformation can significantly destroy the image texture. So, any perturbation that destroys the crowd texture can be used to construct the semantic perturbation space of foreground $\Omega_{sp}^f(X_u)$. For simplicity, two classical texture transformations, color-jitter $\mathcal{P}_{\text{sp}}^1$ and grayscale $\mathcal{P}_{\text{sp}}^2$, are used here to perturb the texture in the head area while remaining most of the shape information. $\mathcal{P}_{\text{sp}}^1$ can be parameterized by brightness, contrast, saturation and hue. They determine the feasible perturbation space of color transformation. In our experiments, brightness, contrast and saturation are allowed to vary from 0.5 to 1.5, while hue is set in the range of -0.5 to 0.5. $\mathcal{P}_{\text{sp}}^2$ converts a color image to a gray image.

Background perturbation: It is necessary to strongly perturb the background area of the crowd image since the texture transformation in the foreground is inadequate to produce enough consistency loss, particularly in a low-density crowd scene. The background objects with significant textural features (e.g., trees, sand, and rocks, etc.) are frequently considered as crowd heads in the counting model [49], because they look similar to the high-density crowd areas, as shown in Fig. 6. Essentially, the background objects, namely the hard samples, are strong perturbations to the background. Therefore, a conflict-texture transfer (CTT) module is designed to construct the background perturbation $\Omega_{\text{sp}}^b(X_u) = \{T_{\text{sp}}^b(X_u; \phi^i); \phi^i \in \Phi^b\}$. T_{sp}^b adds the texture information of hard samples into the background via a mixed-based technique. CTT consists of a conflict texture dictionary and a texture transfer technique, as shown in Fig. 7. The conflict texture dictionary includes images with significant texture characteristics that cause the counting model to overestimate the number of people. The textures in the image dictionary change the original information in the background area and are known as conflicting textures. The images in the dictionary include background and crowd images. The background images similar to the crowd are put into the dictionary, and they are collected by search engines with keywords like trees, beans, density objects, etc. The crowd images are collected from the high-density crowd images of the crowd dataset we use. We note that only part of texture influences the counting model, and crop the conflict texture region from the collected images. Thus, a well-trained CSRNet is used to predict the density map of the collected images. Afterward, the high-density region of each image is cropped with a fixed size 400 × 400. The cropped images are then saved to the texture dictionary $\mathcal{S} = \{S_i\}_{i=1}^{N_d}$. N_d is the number of images in the dictionary.

The whole process of CTT is given in the following. First, the background area of an image is obtained by $X_u^b = X_u \odot (1 - M_u)$, where $\odot$ denotes the Hadamard product. Then,

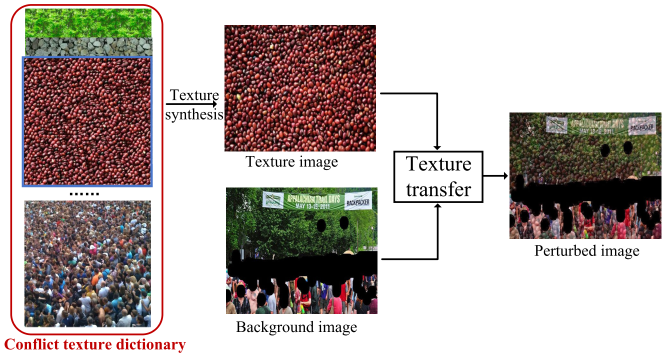


Fig. 7. Illustration of the conflict-texture transfer module. The dictionary contains the images which have the crowd texture pattern. The texture patterns are migrated by CTT from the dictionary to the background of input images.

a texture image $S_i \in \mathcal{S}$ is randomly selected. Since the size of S_i may not match X_u^b, the texture synthesis [50] is used here to adjust S_i. Finally, the texture pattern of S_i is transferred to X_u^b using texture transfer technique. Two ways are considered to transfer the texture.

a) Pixel space mixup: X_u^b is mixed by S_i with a weight factor α_p in the pixel space. It is set to 0.3 in our experiment. The mixup image $\hat{X}_u^b$ is thus expressed by

$$\hat{X}_u^b = \alpha_p \cdot S_i + (1 - \alpha_p) \cdot X_u^b. \tag{4}$$

b) Color space mixup: X_u^b and S_i are converted to the HSV space. Since the counting model is sensitive to the hue changes in color, they are mixed in the hue direction, and the mixed image is again wrapped back to the RGB space. This process can be formulated by

$$\hat{X}_u^b = H^{-1}(\alpha_h \cdot H(X_u^b)_{hue} + (1 - \alpha_h) \cdot H(S_i)_{hue}), \tag{5}$$

where $H(\cdot)$ is the mapping function from RGB to HSV, α_h is a hyperparameter to wrap the texture information of S_i to X_u^b. In the experiment, α_h is set to 0.85.

It should be noted that the high-density crowd image S_i is only suitable for the color space mixup method, but the other images of $\mathcal{S}$ are suitable for both.

Now, for an unlabeled image X_u, SMG is first used in STT to generate its spatial crowd mask, then HTT is adopted to perform strong texture transformations $T_{\text{sp}}^f(\cdot)$ for the crowd area and the conflict texture transfer $T_{\text{sp}}^b(\cdot)$ for the background area, respectively. The perturbed image $\hat{X}_u$ is given by

$$\hat{X}_u = \quad T_{sp}^f(G(X_u) \odot X_u) + T_{sp}^b((1 - G(X_u)) \odot X_u). \tag{6}$$

C. Adversarial Perturbation for Non-Semantic Perturbation

STT generates strong semantic perturbations to improve the performance of SSCC. However, the perturbation diversity of STT is not sufficient due to the limited prior knowledge and the crowd distribution difference of images. We found in [51] that the non-robust brittle features give rise to the vulnerability of neural networks, and they are small yet highly predictive non-semantic patterns. The finding is consistent with the prior knowledge regarding non-semantic perturbation, as mentioned in Section III. Therefore, we extend the perturbation space from semantic to non-semantic space, and to generate diverse perturbations and enough consistency loss on unlabeled data. Formally, the NSP space of an input image X is defined by $\Omega_{nsp}(X, \varepsilon) := \{X + \delta; \|\delta\|_p \leq \varepsilon\}$. δ is a perturbation value, whose p-norm is less than ε so that the perturbation does not change the label of X. For any input X, its noise $\rho(X)$ can be obtained through adversarial perturbation (AP), given by

$$\rho(X) := \arg\max_{\delta;\|\delta\|_p \leq \varepsilon} \mathcal{L}(X + \delta, Y, \theta), \tag{7}$$

where Y is the ground-truth target of X, θ is the model parameters. Thus, the perturbed input is $\hat{X} = X + \rho(X)$.

Eq. 7 is a constrained non-convex maximum optimization problem. The project gradient descent (PGD) method [45] is used to search the optimal perturbations of the problem under the constrained condition. It iterates in the rising direction of gradients and projects the adversarial sample into a feasible set, and yields

$$\hat{X}^{k+1} = f_{\Omega_{\text{nsp}}(X,\varepsilon)}(\hat{X}^k + \alpha_s \cdot \text{sign}(\triangledown_X \mathcal{L}(\hat{X}^k, Y, \theta))), \tag{8}$$

where $f_{\Omega_{\text{nsp}}(X,\varepsilon)}$ projects the perturbation sample X_p^k, obtained in the k-th iteration, to the ε-ball centered on input image X with the L_∞ distance metric. α_s is the perturbation coefficient of each iteration. However, under the SSL setting, the ground truth Y is unavailable, and thus it is replaced by the pseudo target $F(X;\theta^t)$. Define T_{nsp} to be non-semantic perturbation function, for each unlabeled image X_u, its non-semantic perturbation $T_{\text{nsp}}(X_u)$ is computed by

$$\hat{X}_u^{k+1} = \hat{X}_u^k + \alpha_s \cdot \text{sign}(\triangledown_{X_u} \|F(\hat{X}_u^k;\theta^s) - F(X_u;\theta^t)\|_2^2), \tag{9}$$

$$\hat{X}_u^{k+1} = \arg\min_{x\in\Omega_{\text{nsp}}(X_u,\varepsilon)} \|x - \hat{X}_u^{k+1}\|_\infty. \tag{10}$$

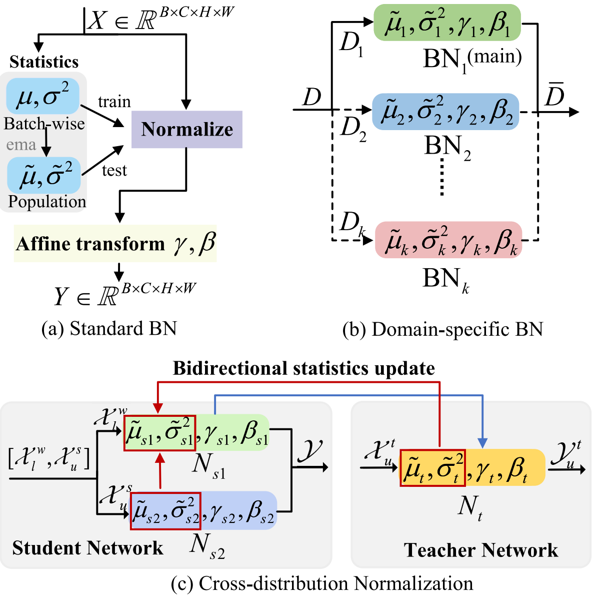
The details of NSP are summarized in Algorithm 1. In our experiment, the number of PGD iterations K is set to 5, and the adversarial perturbation is constrained into the ε-ball of norm L_∞ defined by Eq. (10). Here, the negative gradient of loss $\mathcal{L}$ with respect to X is computed by backpropagation, and the BN statistics updating is disabled during non-semantic perturbation. Since each sample needs to perform K forward and backward operations, we use the Free-AT training strategy [52] to reduce the computational cost of adversarial perturbation.

D. Cross-Distribution Normalization (CDN)

The BN layer of counting model normalizes each channel of input features to be zero-mean and unitary variance, as shown in Fig. 8 (a). Its cross-sample dependency property [53] requires that the samples in a batch have same distribution for accurate statistics. However, in SSL, strong perturbation may change the distribution of unlabeled data, resulting in a distribution shift between labeled and unlabeled data within a batch, as shown in Fig. 9. That is to say, there has distribution shift between training and testing sets. In [54], [55], and [56], BN modules are designed in terms of data distributions, and the main BN is used for each layer during inference, as shown in Fig. 8 (b). These methods only consider the differences in data distribution in a single network, and obtain a sub-optimal

Algorithm 1 Non-Semantic Perturbation

```
Input: Unlabeled data $\mathcal{D}_u$, student network weight $\theta^s$,
       teacher network weight $\theta^t$, perturbation budget
       $\varepsilon$, the number of iterations $K$, step size per
       iteration $\alpha_s$.
Output: Perturbation samples.
for mini-batch $X_u^{\mathcal{B}} \in \mathcal{D}_u$ do
    Generate pseudo targets $Y_u^{\mathcal{B}} \leftarrow F(X_u^{\mathcal{B}}; \theta^t)$;
    Assign initial noises $\rho(X_u^{\mathcal{B}}) \leftarrow \mathbf{U}(-\varepsilon, \varepsilon)$;
    $\hat{X}_u^{\mathcal{B}} \leftarrow X_u^{\mathcal{B}} + \rho(X_u^{\mathcal{B}})$;
    for $k$=1 to $K$ do
        $g \leftarrow \nabla_X \mathcal{L}(\hat{X}_u^{\mathcal{B}}, Y_u^{\mathcal{B}}, \theta^s)$;
        $\hat{X}_u^{\mathcal{B}} \leftarrow \hat{X}_u^{\mathcal{B}} + \alpha_s \cdot \mathrm{sign}(g)$;
        Project $\hat{X}_u^{\mathcal{B}}$ into $\Omega_{\mathrm{nsp}}(\hat{X}_u^{\mathcal{B}}, \varepsilon)$ by Eq. (10);
    end
    return perturbation samples $\hat{X}_u^{\mathcal{B}}$.
end
```


$X \in \mathbb{R}^{B\times C\times H\times W}$
Statistics
μ, σ^2
Batch-wise
train
ema
Normalize
$\tilde{\mu}, \tilde{\sigma}^2$
test
Population
Affine transform γ, β
$Y \in \mathbb{R}^{B\times C\times H\times W}$
D_1
$\tilde{\mu}_1, \tilde{\sigma}_1^2, \gamma_1, \beta_1$
BN$_1$(main)
D
D_2
$\tilde{\mu}_2, \tilde{\sigma}_2^2, \gamma_2, \beta_2$
BN$_2$
$\bar{D}$
D_k
$\tilde{\mu}_k, \tilde{\sigma}_k^2, \gamma_k, \beta_k$
BN$_k$
Bidirectional statistics update
$[\mathcal{X}_l^w, \mathcal{X}_u^s]$
$\mathcal{X}_l^w$
$\tilde{\mu}_{s1}, \tilde{\sigma}_{s1}^2, \gamma_{s1}, \beta_{s1}$
N_{s1}
$\mathcal{Y}$
$\mathcal{X}_u^s$
$\tilde{\mu}_{s2}, \tilde{\sigma}_{s2}^2, \gamma_{s2}, \beta_{s2}$
N_{s2}
Student Network
$\mathcal{X}_u^t$
$\tilde{\mu}_t, \tilde{\sigma}_t^2, \gamma_t, \beta_t$
N_t
$\mathcal{Y}_u^t$
Teacher Network

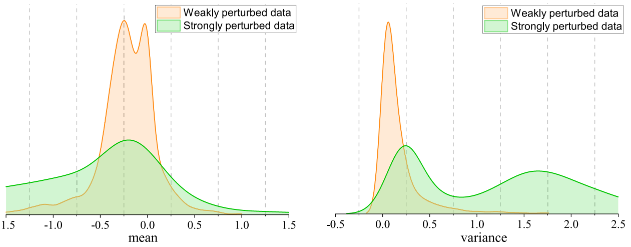
(a) Standard BN (b) Domain-specific BN

(c) Cross-distribution Normalization

Fig. 8. Comparison of different normalization methods. (a) Standard BN assumes that samples within a mini-batch come from a similar distribution. (b) Domain-specific BN (DSBN) uses different BN modules for different distributions. (c) Cross-distribution normalization uses a bidirectional statistics update mechanism to increase the accuracy of population statistics.

performance in the teacher-student framework due to train-test inconsistency. To this end, the cross-distribution normalization technique is introduced to stabilize model training in the teacher-student framework. Different BN modules are used for weakly and strongly perturbed data, and a bidirectional statistics update mechanism is used to match the BN parameters with the model weights across the two networks, as shown in Fig. 8 (c).

In the student network, its input batch includes the weakly-perturbed labeled data $\mathcal{X}_l^w$ and the strongly-perturbed unlabeled data $\mathcal{X}_u^s$. Each of its normalization layers consists of two parallel BN modules $N_{s1}(\cdot; \tilde{\mu}_{s1}, \tilde{\sigma}_{s1}^2, \gamma_{s1}, \beta_{s1})$ and $N_{s2}(\cdot; \tilde{\mu}_{s2}, \tilde{\sigma}_{s2}^2, \gamma_{s2}, \beta_{s2})$. $\tilde{\mu}_*, \tilde{\sigma}_*^2$ and γ_*, β_* denote the population statistics (mean and variance) of the whole training set, as well as the scaling and shifting parameters of the affine transform. During training, the two BN modules normalize the features of $\mathcal{X}_l^w$ and $\mathcal{X}_u^s$ with batch-wise statistics, respectively. In the teacher network, each normalization layer only has a BN module $N_t(\cdot; \tilde{\mu}_t, \tilde{\sigma}_t^2, \gamma_t, \beta_t)$, since the teacher only receives the weakly-perturbed unlabeled data $\mathcal{X}_u^w$. When the teacher parameters are updated, the statistics of N_t are updated simultaneously by



(a) Mean distribution (b) Variance distribution

Fig. 9. Comparison of BN statistics distributions for weakly-perturbed label data and strongly-perturbed unlabeled data.

$$\tilde{\mu}_t \leftarrow \alpha_{\mathrm{ema}} \tilde{\mu}_t + (1 - \alpha_{\mathrm{ema}}) \tilde{\mu}_{s1}, \tag{11}$$

$$\tilde{\sigma}_t^2 \leftarrow \alpha_{\mathrm{ema}} \tilde{\sigma}_t^2 + (1 - \alpha_{\mathrm{ema}}) \tilde{\sigma}_{s1}^2, \tag{12}$$

where $\tilde{\mu}_{s1}$ and $\tilde{\sigma}_{s1}^2$ are the student population statistics mixture of labeled and unlabeled data. Due to the domain gap between the student and teacher training data, $\tilde{\mu}_t, \tilde{\sigma}_t^2$ are not suitable for the teacher. Hence, N_t uses the batch-wise statistics μ_t, σ_t^2 to normalize the inputs in the pseudo-label prediction process. Each iteration $\tilde{\mu}_t, \tilde{\sigma}_t^2$ are updated by the EMA of μ_t, σ_t^2, and are integrated into N_{s1}. The statistics of N_{s1} is updated by the batch-wise statistics of N_{s2}, and yields

$$\tilde{\mu}_{s1} \leftarrow \alpha_1 \tilde{\mu}_{s1} + \alpha_2 \tilde{\mu}_t + \alpha_3 \mu_{s2}, \tag{13}$$

$$\tilde{\sigma}_{s1}^2 \leftarrow \alpha_1 \tilde{\sigma}_{s1}^2 + \alpha_2 \tilde{\sigma}_t^2 + \alpha_3 \sigma_{s2}^2, \tag{14}$$

where μ_{s2} and σ_{s2}^2 denote the batch-wise statistics in N_{s2}. N_{s1} is used to normalize the input features of the student during inference. $\alpha_1, \alpha_2, \alpha_3$ are the non-negative trade-off weights, and they are set respectively to 0.5, 0.3, 0.2 in our experiment.

E. Optimization Objective

The optimization objective is to minimize the weighted sum of the supervised loss $\mathcal{L}_s$ on labeled image $\mathcal{D}_l$ and the consistency loss $\mathcal{L}_u$ on unlabeled image $\mathcal{D}_u$. $\mathcal{L}_s$ is computed by the Euclidean distance between the predicted and ground-truth density maps by

$$\mathcal{L}_s = \sum_{i=1}^{N_l} \left\| F(X_l^i; \theta^s) - Y_l^i \right\|_2^2. \tag{15}$$

$\mathcal{L}_u$ is calculated by the prediction difference between teacher and student under the different perturbations of the same unlabeled data, at the output and feature levels, namely, $\mathcal{L}_u^o$ and $\mathcal{L}_u^f$. They are

$$\mathcal{L}_u^o = \sum_{j=1}^{N_u} \left\| F(T_s(X_u^j); \theta^s) - F(T_w(X_u^j); \theta^t) \right\|_2^2, \tag{16}$$

TABLE IV

THE DETAILS OF SHANGHAITECH [2], UCF-QNRF [58], JHU-CROWD++ [59], NWPU-CROWD [49]. THE TRAINING SET OF EACH DATASET IS DIVIDED INTO LABELED SET $\mathcal{D}_l$ AND UNLABELED SET $\mathcal{D}_u$

Dataset	Number of images	Avg Resolution	Count statistics				Train set		Val set	Test set
			Total	Min	Ave	Max	$\lvert\mathcal{D}_l\rvert$	$\lvert\mathcal{D}_u\rvert$		
SHHA [2]	482	589 × 868	241,677	33	501	3139	30	240	30	182
SHHB [2]	716	768 × 1024	88,488	9	123	578	40	320	40	316
UCF-QNRF [58]	1535	2013×2902	1,251,642	49	815	12865	120	961	120	334
JHU-Crowd++ [59]	4372	910×1430	1,515,005	0	346	25,791	227	2045	500	1600
NWPU-Crowd [49]	5109	2191×3209	2,133,375	0	418	20,033	311	2798	500	1500

$$\mathcal{L}_u^f = \sum_{j=1}^{N_u} \left\| \frac{v_t^j}{\|v_t^j\|_2} - \frac{z_s^j}{\|z_s^j\|_2} \right\|_2, \tag{17}$$

where $T_s = T_{\text{sp}} \circ T_{\text{nsp}}$ is a strong perturbation. T_w is a weak perturbation, generated by the CDA module. θ^t is the teacher weights, which are the EMA of student weights θ^s. v_t^j is the encoder output of the teacher. z_s^j represents the projector output of the student. The projector includes two 1 × 1 convolution layers with channels 256 and 512. $\mathcal{L}_u^f$ is used to prevent from training collapse [57].

Consequently, the total loss $\mathcal{L}_{total}$ of the counting model can be given by

$$\mathcal{L}_{total} = \lambda_1 \mathcal{L}_s + \lambda_2 \mathcal{L}_u^o + \lambda_3 \mathcal{L}_u^f, \tag{18}$$

where $\lambda_1, \lambda_2, \lambda_3$ are the trade-off weights, and set to 1.0, 0.2, 0.2 in our experiments.

V. EXPERIMENTS

To validate the proposed semi-supervised method, extensive experiments have been carried out on mainstream datasets, namely, ShanghaiTech [2], UCF-QNRF [58], JHU-Crowd++ [59], and NWPU-Crowd [49]. The experimental results are analyzed and compared with the other SSL-based crowd counting methods. Ablation studies are performed on ShanghaiTech Part A dataset to demonstrate the performance of the proposed components.

A. Implementation Details

To study SSL with limited labeled data, a strong counting baseline (SCB) is designed based on CSRNet [3]. It adopts the first thirteen layers of VGG16-BN as backbone. As done in [8], all images are augmented by the random horizontal flipping with a probability of 0.5, the multi-scale training with a scale factor ranging from 0.8 to 1.2, and the random cropping with a fixed size of 400 × 400. The above data augmentations are used by the CDA module to generate weakly perturbed data. To further obtain strongly perturbed data, the hybrid perturbation strategy proposed above is used to generate the SP and NSP perturbations. In SP, we set the parameters b, k of SMG to 0.01 and 40, and use the *ColorJitter* function in Pytorch to implement the texture transformation of the crowd area, but randomly change the order of the operations in *ColorJitter*. The conflict-texture dictionary is obtained in advance before the SSL training. It uses 20 background texture images obtained from the Internet and half of the crowd images in $\mathcal{D}_u$, namely, $N_d = 20 + 0.5 * |\mathcal{D}_u|$. In NSP, we set $\varepsilon = 0.01, \alpha_s = 0.005, K = 5$ to balance perturbation strength and computation cost. The Adam optimizer with a weight decay of 1e-4 is used to train the proposed framework. The learning rate is initially set to 1e-4, which is decayed by a factor of 0.995 every epoch. The size of mini-batch is set to be 12, including 6 labeled images and 6 unlabeled images. A factor $\alpha_{\text{ema}} = 0.99$ is used to update the teacher weights θ^t with the EMA of student weights θ^s. Following [3], the fixed Gaussian kernel with a size of 15 × 15 is used to generate the ground-truth density map. MAE and RMSE are used here to evaluate the performance of the counting model. We first train the model 20 epochs with $\mathcal{L}_s$ to obtain a good teacher, and then train the model 500 epochs with $\mathcal{L}_{total}$. Experiments are performed with PyTorch and two RTX-2080Ti GPUs.

B. Datasets

The mainstream datasets widely used are given in Table IV. **ShanghaiTech** [2] includes Part A and Part B, called here with SHHA and SHHB. The images of SHHA are crawled from the Internet, having different views and densities. The images of SHHB are captured on the street with a fixed camera, having limited diversity. Hence, it is more difficult to accurately count the crowd on SHHA. **UCF-QNRF** [58] contains 1535 images of congested crowd scenes, with a total of 1,215,642 people. **JHU-Crowd++** [59] comprises 4372 unconstrained crowd images with image-level and head-level annotations. These images are divided into training, validation and testing sets. **NWPU-Crowd** [49] is a congested crowd dataset with counting and localization annotation. Its images have an average resolution of 2191 × 3209. UCF-QNRF, JHU-Crowd++ and NWPU-Crowd are high-resolution images. Herein, the images with more than 1600 pixels in the lengthwise direction are downscaled to 1600 pixels, while maintaining the same aspect ratio. Since SHHA, SHHB, and UCF-QNRF have no validation sets, 10% of their training data is used as the validation set. The resultant training set of the four datasets is further divided into 10% labeled data and 90% unlabeled data, following the default partition protocol.

C. Evaluation and Comparison

The semi-supervised crowd counting framework is established with PyTorch.[1] It integrates the SSL methods into the crowd counting, such as native self-training (N-ST) [40], MT [16], VAT [17], UDA [18]. They are based on SCB and use the same configuration, including batch size, common

[1] https://github.com/KingMV/SSCC-framework

TABLE V

Comparison Study of the Leading SSCC Methods on Various Crowd Count Datasets. _† Denotes the Reproduction Results in Our Experiment. Red Numbers Show the Performance of Ranking First and blue for Ranking Second Under 10% Partition Protocol. FS and SS Represent Fully-Supervised and Semi-Supervised Learning

	Method	SHHA [2]		SHHB [2]		UCF-QNRF [58]		NWPU-Crowd [49]		JHU-Crowd++ [59]	
		MAE↓	RMSE↓	MAE↓	RMSE↓	MAE↓	RMSE↓	MAE↓	RMSE↓	MAE↓	RMSE↓
FS	MCNN [2]	110.2	173.2	26.4	41.3	-	-	218.5	700.6	160.6	377.7
	CSRNet [3]	68.2	115.0	10.6	16.0	-	-	104.9	433.5	85.9	309.2
	SANet [31]	67.0	104.5	8.4	13.6	-	-	171.2	471.5	91.1	320.4
	CANNet [32]	62.3	100.0	7.8	12.2	107	183	93.6	489.9	118.8	362.5
	F-SCB†	62.9	100.8	6.8	11.2	112.4	210.9	71.8	456.8	87.5	311.6
	Label-only(10%)	96.3	167.3	15.2	26.7	156.3	278.3	146.0	654.5	127.5	428.9
SS	MT† [16]	91.5	155.1	16.1	34.2	154.2	262.8	126.3	651.7	107.5	385.6
	UDA† [18]	86.1	151.7	16.8	26.4	152.2	252.8	120.4	625.5	105.4	415.8
	N-ST† [60]	90.3	156.2	15.2	26.7	156.3	278.3	125.6	693.4	110.1	414.1
	VAT† [17]	92.1	150.2	15.1	24.9	147.1	252.9	124.2	682.5	114.7	430.9
	L2R [10]	90.3	153.5	15.6	24.4	148.9	249.8	-	-	-	-
	IRAST [8]	86.9	148.9	14.7	22.9	135.6	233.4	-	-	-	-
	MTCP [13]	81.3	130.5	14.5	22.3	-	-	-	-	-	-
	HPS	73.9	123.5	10.9	20.6	121.4	209.6	115.3	532.3	100.5	363.4
	GP(20%) [11]	91	149	-	-	147	226	-	-	-	-
	SUA(40%) [9]	68.5	121.9	14.1	20.6	130.3	226.3	111.7	443.2	80.7	290.8
	MTCP(40%) [13]	-	-	-	-	-	-	108.1	440.1	80.5	288.4
	HPS(40%)	64.7	106.3	7.1	11.4	114.4	196.5	82.7	521.1	88.3	315.6

TABLE VI

Ablation Study of Components and Losses on SHHA and SHHB Dataset. The Labeled and Unlabeled Data Are 10% and 90% of Train Data, Respectively

Method	$\mathcal{L}_s$	$\mathcal{L}_u^o$	$\mathcal{L}_u^f$	SHHA		SHHB	
				MAE↓	RMSE↓	MAE↓	RMSE↓
Label-only	✓	×	×	96.3	167.3	15.2	26.7
MT	✓	✓	×	91.5	155.1	14.9	28.4
BL†(MT+CDN)	✓	✓	✓	85.6	152.2	14.5	25.1
BL†+STT	✓	✓	✓	76.2	132.2	11.2	21.3
BL†+AP	✓	✓	✓	80.3	136.8	12.4	23.1
BL†+STT+AP	✓	✓	×	74.5	129.1	11.3	21.5
BL†+STT+AP	✓	✓	✓	73.9	123.5	10.9	20.6

data augmentation and learning rate, etc. Experimental results are given in Table V. Except for the N-ST, MT, UDA, and VAT methods, the results of the other methods come from the related papers. NWPU-Crowd is evaluated on its validation set because the test set is not available. We provide the results of Label-only which trains the SCB baseline with 10% labeled data, and F-SCB which uses 100% labeled data. Their results represent the lower and upper bounds of the performance of the baseline under the SSL setting. Overall, it can be seen that HPS is better than the leading semi-supervised crowd counting methods, such as L2R [10], IRAST [8], GP [11], SUA [9] and MTCP [13] under the same partition protocol, and significantly outperforms the classical SSL methods of MT [16], UDA [18], VAT [17], N-ST [40] on all the four datasets. The metrics of these four classical methods are a little bit better than Label-only. This suggests that the perturbations used by the SSL methods are not suitable for crowd counting, since they are specifically designed for image classification. Hence, the prior knowledge of crowd counting is used in the hybrid perturbation strategy to produce strong and diverse perturbations on crowd images, and to improve SSCC.

With the 10% partition protocol, HPS surpasses respectively latest MTCP by 9.1% and 24.8% on SHHA and SHHB, in the metric of MAE. Moreover, its performance slightly improves 10.4%, 4.2%, and 4.6% over the second-best method on complex datasets like UCF-QNRF, NWPU-Crowd, and JHU-Crowd++. In SUA [9], 40% and 10% of the training data are used respectively as labeled training data and validation data. Therefore, we conduct experiments with 40% labeled samples from the four datasets. The results in the bottom of Table V show that HPS outperforms SUA and MTCP in the metrics of MAE and RMSE on all datasets (except JHU-Crowd++). The framework of SUA is similar to HPS. That is, they are both based on a teacher-student framework, but their motivations are different. SUA utilizes an uncertainty-aware map to reduce the pseudo-label noise from unlabeled data, to improve the quality of pseudo-label generated by surrogate tasks. Unlike this, HPS concentrated on the strong perturbation design for unlabeled data so that the counting model can learn generalized features. As a whole, the performance of HPS is better than the other semi-supervised counting models when we set aside 40% and 10% of the training images as labeled datasets. It is even close to the performance of the fully-supervised counting models. Therefore, suitable perturbations are important in SSCC when a consistency-based SSL framework is used. Fig. 10 gives the visualization results of the counting methods. It can be seen that HPS is superior in the high-density and background areas, illustrated by red boxes.

D. Ablation Study

The following aspects of the proposed method are analyzed on ShanghaiTech test sets through ablation studies.

1) Analysis of Modules and Losses: HPS comprises the modules of STT, AP and CDN, and uses three losses ($\mathcal{L}_s$, $\mathcal{L}_u^o$, $\mathcal{L}_u^f$). Here, STT and AP are used to implement semantic and non-semantic perturbations, respectively. We carry out ablation studies to evaluate the proposed modules. Since STT and AP are related to $\mathcal{L}_u^o$ and $\mathcal{L}_u^f$, we discuss them together. The results

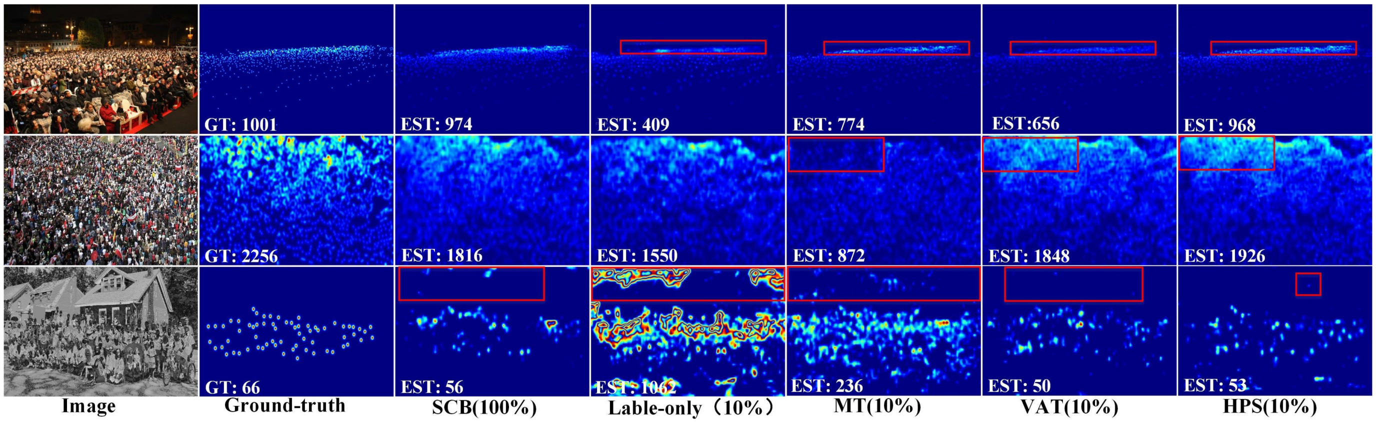


Fig. 10. The estimated density maps of different crowd counting methods. The first column gives the original images. The second column provides the ground-truth density maps. The rest columns show the estimated results by SCB, Label-only, MT [16], VAT [17] and HPS. The predicted counting results are given at the left-bottom corner of the images. Red boxes highlight the difference of estimated density maps. The quality of the original image in the third row is poor, while HPS still achieves better performance.

are given in Table VI. First, we train CSRNet with $\mathcal{L}_s$ on 10% labeled images of the training dataset. The performance of **Label-only** is poor due to the limited labeled data. Then, we add the unlabeled data and train the model with $\mathcal{L}_u^o + \mathcal{L}_s$, The metrics of MT [16] are slightly better than **Label-only**, but still unsatisfactory. When we put CDN to MT, the MAE metric of **BL**† varies from 91.5 to 85.6, on the SHHA test set. It means that CDN is useful to improve model optimization, and it is used in the following experiments. Then, STT is put to **BL**†**+STT** to generate semantic perturbations for unlabeled data, and we find it outperforms **BL**† in MAE by 11.9%. When we put only adversarial perturbation (AP) to **BL**†, the MAE of **BL**†**+AP** is improved by 6.2%. Although the STT and AP modules can both improve the performance of **BL**†, the former improves more than the latter. This indicates that the semantic perturbations designed with the prior knowledge of crowd counting are necessary for SSCC. So, we put STT and AP to unlabeled data, the MAE and RMSE metrics of **BL**†**+STT+AP** are 73.9 and 123.5 on the SHHA test set. It shows that they are complementary and their combination generates strong and diverse perturbations to unlabeled data. If $\mathcal{L}_u^f$ is discarded during model training, the performance of **BL**†**+STT+AP** decreases. It means that $\mathcal{L}_u^f$ is useful for consistency regularization and training collapse avoidance. With STT, AP and CDN and all the three losses, **BL**†**+STT+AP** increases respectively 13.7% and 18.9% in the MAE and RMSE metrics, compared with **BL**†, and outperforms the leading SSCC methods.

2) Analysis of Semantic Perturbation: Table VII summarized the results of different perturbations in the foreground and background areas. When we perform the texture perturbation on the foreground area of unlabeled images, the MAE metric decreases by nearly 8.4%. However, the performance of the counting model cannot get a big improvement when the whole image is perturbed with the same texture transformation. We also find that the counting model almost has no performance improvement when only the background region is perturbed. It means that the right texture transformation in the background area is needed to improve the SSCC performance.

TABLE VII

Counting Performance on SHHA Test Set With Different Perturbation Regions and Different Perturbation Strategies. The First Row Is the Baseline Performance. F and B Represent Foreground and Background Receptively. TP Means Texture Perturbation Including Color Jitter and Grayscale

Perturbation area		Perturbation strategy	MAE	RMSE
F	B			
-	-	-	85.6	152.2
✓	×	TP	78.4	133.4
✓	✓	TP	77.7	125.8
×	✓	TP	84.3	154.9
×	✓	CTT	79.1	135.7
×	✓	Cutmix [25]	80.8	140.8
✓	✓	HTT	76.2	132.2

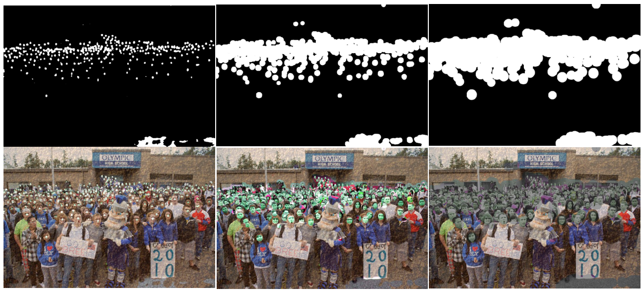


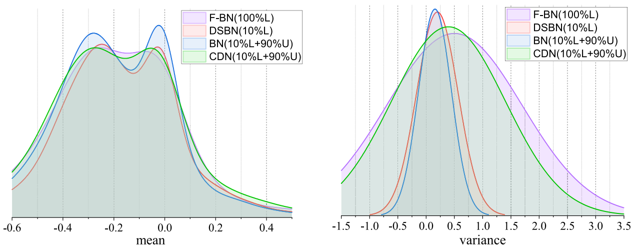
Fig. 11. The spatial masks and the corresponding perturbation images. Top: the spatial masks with dilated kernel sizes of 1 × 1, 40 × 40, 80 × 80. Bottom: the visualization of perturbation images.

TABLE VIII

The Counting Performance on SHHA Test Set With Different Crowd Spatial Mask Sizes. $* \times *$ Represents the Size of the Dilated Kernel

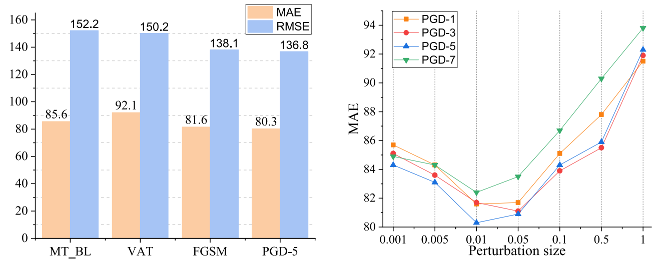
Size	–	5 × 5	10 × 10	30 × 30	40 × 40	50 × 50	80 × 80
MAE	85.9	80.8	80.3	76.7	76.2	76.1	78.9
RMSE	156.7	138.4	135.8	129.9	132.2	130.1	137.8

In Table VII, it can be seen that the model performance improves significantly when we use CTT for the background area. If the background of images is randomly mixed within a mini-batch [25], the model performance gets further improved,



(a) Mean distribution (b) Variance distribution

Fig. 12. Comparison of BN statistics distributions under different normalization settings. (a) and (b) represent mean and variance distributions of BN layers in a trained model, respectively. F-BN(100%L) denotes the statistics with 100% labeled data, which approximates the population statistics of the training set. DSBN(10%L) [56] is the statistics of main BN with 10% labeled data. BN(10%L+90%U) and CDN (10%L+90%U) represent the statistics with a mixture of 10% labeled data and 90% unlabeled data, respectively.



(a) (b)

Fig. 13. Results Comparison on SHHA test set for different non-semantic perturbations. (a) The performances of Baseline(MT-BL), VAT, FGSM and PGD. (b) the MAE metric with different perturbation sizes.

but lower than CTT. It shows that the hard samples can produce stronger perturbation with conflict textures. With the proposed hybrid texture transformation, the counting performance is significantly improved by 11.0%. Now, we discuss the effect of the spatial mask on the performance of SSCC. Fig. 11 shows the spatial masks and their corresponding perturbation images. The crowd foreground area is changed with the dilated kernel size. Table VIII reports the results of $\mathbf{BL}^{\dagger}$**+STT** with the different spatial masks. It can be seen that the model performance drops when the size of the crowd foreground mask is too small or too big. The metrics of $\mathbf{BL}^{\dagger}$**+STT** are almost the same as $\mathbf{BL}^{\dagger}$ when the size of the mask is small. The reasons are (i) the perturbation strength in the crowd area decreases, (ii) CTT adds high-density crowd images from the texture dictionary into the misclassified crowd area and thus destroys the original semantic labels of the crowd area. On the other side, the model performance gradually improves with the dilation size, and we get the best results with a dilated kernel size of 40×40. However, the performance drops afterward. This is because part of the background area is considered as the crowd area, and the perturbation strength decreases in the background area.

3) Analysis of Non-Semantic Perturbation: Adversarial perturbation is used to achieve NSP. Different adversarial perturbations are studied here, including FGSM [43] and PGD [45]. Specifically, we try two hyper-parameter settings: (a) the maximum perturbation size is within the 0.01-ball, and the number of attack iterations and step sizes is set to $K = 5$ and $\alpha_s = 0.005$ in PGD, (b) the perturbation size varies from 0.001 to 1, and the attack iterations $K = 1, 3, 5, 7$ in PGD. The results in Fig. 13 show that the performance of the counting model is improved with stronger adversarial perturbation in the allowed perturbation space. However, when the perturbation size exceeds the boundary of non-semantic, the performance gradually drops because too strong perturbation may lead to the distribution inconsistency between training and testing data.

TABLE IX

Comparison of Performance for SSCC Under Different Perturbation Strengths on SHHA Test Set. *None* Means That No Strong Perturbation Is Used

Strength level	0(None)	1	2	3
MAE	87.4	78.9	73.9	77.2
RMSE	162.1	142.6	123.5	136.1

4) Analysis of Perturbation Strength: Table IX shows the performance of SSCC with different perturbation strengths on SHHA. Since the predictions of the teacher are used as pseudo targets, its input should be weakly perturbed images. Following [23], we change the perturbation strength of the input images in the student, by adjusting the parameters of SP and NSP. Here, four levels of perturbation are used. We can see that the performance of SSCC improves with the increase of the perturbation strength. However, if the perturbation is too strong, the performance decreases instead. This is because the stronger perturbation destroys the counting label of image. Consequently, it is not advisable to align a strongly perturbed image of semantic distortion with a weakly perturbed one.

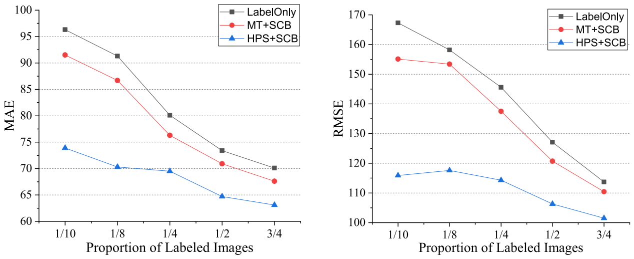
5) Analysis of Cross-Distribution Normalization: The crowd counting performance under the different normalization settings of BN [29], DSBN [56], and CDN, is summarized in Table X. When standard BN is used in the MT framework, its counting performance is poor. With the STT module added, **MT+STT+BN** only decreased MAE by 5.5%, due to no BN statistics update and the distribution shift between weakly and strongly perturbed data. In $\mathbf{MT{+}STT{+}BN}^{\dagger}$, the BN statistics of the teacher are the EMA of the student's BN statistics and are needed to match the weights of the teacher, as done in [47], but it is unable to converge. We argue that it is caused by the inconsistent distribution of training and testing images. Domain-specific BN (DSBN) [56] uses different BN modules for different distributions. As a result, **MT+STT+DSBN** decreases MAE by 4.5% with respect to **MT+STT+BN**. DSBN does not update the BN statistics in the teacher-student architecture, too. Instead, a bidirectional statistics update mechanism is used in CDN to update the BN parameters with the counting model weights across the teacher and student networks, and thus the performance of **MT+STT+CDN** is improved by 16.3%, 11.5% and 7.4% over the first three normalization settings. Fig. 12 shows the BN statistics distributions of the different normalization settings. We can see that the BN statistics distribution of CDN is closer to F-BN(100%L) than others. This proves that CDN can provide more accurate statistics.

6) Influence of the Labeled Images Proportion: To analyze the influence of the number of labeled images on the

TABLE X

THE COUNTING PERFORMANCES OF THE DIFFERENT NORMALIZATION METHODS ON SHHA TEST SET. BN† REPRESENTS THAT THE TEACHER NETWORK USES POPULATION STATISTICS FROM THE STUDENT NETWORK, INSTEAD OF BATCH-WISE STATISTICS

setting	MAE	RMSE
MT + BN	91.7	155.4
MT + STT + BN	86.7	157.3
MT + STT + BN† [47]	NAN	NAN
MT + STT + DSBN [56]	82.8	137.2
MT + STT + CDN	76.7	132.2



(a) (b)

Fig. 14. Counting results on SHHA dataset with different modules and different proportions of labeled images (1/10, 1/8, 1/4, 1/2, 3/4).

TABLE XI

THE ROBUSTNESS OF COUNTING MODELS AGAINST DIFFERENT ATTACKS ON SHHA TEST SET. LABEL-ONLY USES 10% LABELED DATA TO TRAIN THE COUNTING MODEL

Method	SHHA		VA-SHHA		FGSM-SHHA		PGD-SHHA	
	MAE	RMSE	MAE	RMSE	MAE	RMSE	MAE	RMSE
Label-only	97.6	172.3	135.6	254.5	207.8	337.2	289.1	479.4
CSRNet [3]	62.3	103.5	127.8	232.6	162.2	254.6	228.5	317.2
CANNet [32]	61.8	105.1	70.6	116.2	279.6	365.5	394.6	499.3
MT† [16]	91.5	161.3	150.0	245.9	219.2	339.3	229.4	410.4
VAT† [17]	92.1	150.2	108.4	167.6	194.9	309.8	281.0	425.6
HPS(Ours)	70.7	116.3	76.5	127.2	158.7	240.7	194.3	278.8

performance of SSCC, we train the model with various proportions of labeled images, $\frac{1}{10}, \frac{1}{8}, \frac{1}{4}, \frac{1}{2}, \frac{3}{4}$. The images are randomly selected from the training set, while the rest labels are removed and used as unlabeled images. Fig. 14 summarizes the different models performance *vs* the proportion of labeled images. It can be seen that HPS is superior over other methods under the small number of labeled images. With more labeled images added, HPS holds its superiority to the baseline method, however, the performance gap becomes smaller. In other words, the HPS performance can not be improved unlimitedly with the labeled images number.

7) Analysis of Counting Model Robustness: To our knowledge, there are few works about the robustness of semi-supervised crowd counting. Here, this issue is studied on SHHA with various attacks. The test dataset of SHHA is initially attacked with three attack modes, namely, visual augmentation, FGSM [43] and PGD [45]. For simplicity, they are referred to as VA-SHHA, FGSM-SHHA and PGD-SHHA. Table XI summarizes the performances of the counting models. No surprise, the leading fully supervised (CSRNet [3], CANNet [32]) and semi-supervised methods (MT [16], VAT [17]) are vulnerable to these attacks. However, HPS has better robustness against various attacks than the above methods. The reasons are: 1) HPS makes use of semantic and non-semantic perturbations, which generate diverse and hard training samples for the model to learn more discriminative features. 2) HPS is designed based on the robustness analysis of counting models, and it trains the counting model with the samples generated by the strong perturbations so that the model robustness is improved.

VI. CONCLUSION

A consistency-based semi-supervised crowd counting method is proposed with HPS to integrate the prior knowledge of crowd counting in the semantic and non-semantic spaces. HPS comprises three modules, namely, spatial texture transformation, non-semantic adversarial perturbation, and cross-distribution normalization. The texture transformations are used in semantic space to perturb the foreground area in images, while a conflict-texture transfer technique is used to augment the background area. Adversarial perturbation is used to generate diverse and hard perturbations in non-semantic space. Moreover, cross-distribution normalization is introduced to address counting model optimization by the normalization of each distribution sample in a mini-batch and the bidirectional statistics update mechanism. The proposed SSCC is validated on four mainstream datasets, and the experimental results show it significantly improves counting performance. In the future, we will employ other advanced SSL techniques (e.g., pseudo-label denoising) to further improve the model performance and extend HPS to other tasks.

REFERENCES

[1] G. Gao, J. Gao, Q. Liu, Q. Wang, and Y. Wang, "CNN-based density estimation and crowd counting: A survey," 2020, *arXiv:2003.12783*.

[2] Y. Zhang, D. Zhou, S. Chen, S. Gao, and Y. Ma, "Single-image crowd counting via multi-column convolutional neural network," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2016, pp. 589–597.

[3] Y. Li, X. Zhang, and D. Chen, "CSRNet: Dilated convolutional neural networks for understanding the highly congested scenes," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 1091–1100.

[4] Y. Tian, Y. Lei, J. Zhang, and J. Z. Wang, "PaDNet: Pan-density crowd counting," *IEEE Trans. Image Process.*, vol. 29, pp. 2714–2727, 2020.

[5] J. Gao, Y. Yuan, and Q. Wang, "Feature-aware adaptation and density alignment for crowd counting in video surveillance," *IEEE Trans. Cybern.*, vol. 51, no. 10, pp. 4822–4833, Oct. 2021.

[6] X. Jiang et al., "Density-aware multi-task learning for crowd counting," *IEEE Trans. Multimedia*, vol. 23, pp. 443–453, 2021.

[7] Q. Wang, J. Gao, W. Lin, and Y. Yuan, "Pixel-wise crowd understanding via synthetic data," *Int. J. Comput. Vis.*, vol. 129, no. 1, pp. 225–245, Jan. 2021.

[8] Y. Liu, L. Liu, P. Wang, P. Zhang, and Y. Lei, "Semi-supervised crowd counting via self-training on surrogate tasks," in *Proc. Eur. Conf. Comput. Vis.* Cham, Switzerland: Springer, 2020, pp. 242–259.

[9] Y. Meng et al., "Spatial uncertainty-aware semi-supervised crowd counting," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Oct. 2021, pp. 15549–15559.

[10] X. Liu, J. V. D. Weijer, and A. D. Bagdanov, "Exploiting unlabeled data in CNNs by self-supervised learning to rank," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 41, no. 8, pp. 1862–1878, Aug. 2019.

[11] V. A. Sindagi, R. Yasarla, D. S. Babu, R. V. Babu, and V. M. Patel, "Learning to count in the crowd from limited labeled data," in *Proc. ECCV*. Cham, Switzerland: Springer, 2020, pp. 212–229.

[12] Y. Xu et al., "Crowd counting with partial annotations in an image," in *Proc. IEEE/CVF Int. Conf. Comp. Vis. (ICCV)*, Oct. 2021, pp. 15570–15579.

[13] P. Zhu, J. Li, B. Cao, and Q. Hu, "Multi-task credible pseudo-label learning for semi-supervised crowd counting," *IEEE Trans. Neural Netw. Learn. Syst.*, early access, pp. 1–13, Feb. 8, 2023, doi: 10.1109/TNNLS.2023.3241211.

[14] G.-J. Qi and J. Luo, "Small data challenges in big data era: A survey of recent progress on unsupervised and semi-supervised methods," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 44, no. 4, pp. 2168–2187, Apr. 2022.

[15] X. Yang, Z. Song, I. King, and Z. Xu, "A survey on deep semi-supervised learning," *IEEE Trans. Knowl. Data Eng.*, vol. 35, no. 9, pp. 8934–8954, Sep. 2023.

[16] A. Tarvainen and H. Valpola, "Mean teachers are better role models weight-averaged consistency targets improve semi-supervised deep learning results," in *Proc. Adv. Neural Inf. Process. Syst. (NeurIPS)*, 2017, pp. 1195–1204.

[17] T. Miyato, S.-I. Maeda, M. Koyama, and S. Ishii, "Virtual adversarial training: A regularization method for supervised and semi-supervised learning," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 41, no. 8, pp. 1979–1993, Aug. 2018.

[18] Q. Xie, Z. Dai, E. Hovy, T. Luong, and Q. Le, "Unsupervised data augmentation for consistency training," in *Proc. Adv. Neural Inf. Process. Syst. (NIPS)*, vol. 33, 2020, pp. 6256–6268.

[19] G. French, S. Laine, T. Aila, M. Mackiewicz, and G. Finlayson, "Semi-supervised semantic segmentation needs strong, varied perturbations," in *Proc. BMVC*, 2020, pp. 1–14.

[20] S. Mittal, M. Tatarchenko, and T. Brox, "Semi-supervised semantic segmentation with high- and low-level consistency," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 43, no. 4, pp. 1369–1379, Apr. 2021.

[21] A. Ghosh and A. H. Thiery, "On data-augmentation and consistency-based semi-supervised learning," in *Proc. ICLR*, 2021, pp. 1–13.

[22] X. Wang, D. Kihara, J. Luo, and G.-J. Qi, "EnAET: A self-trained framework for semi-supervised and supervised learning with ensemble transformations," *IEEE Trans. Image Process.*, vol. 30, pp. 1639–1647, 2021.

[23] X. Wang and G.-J. Qi, "Contrastive learning with stronger augmentations," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 45, no. 5, pp. 5549–5560, May 2023.

[24] H. Zhang, M. Cisse, Y. N. Dauphin, and D. Lopez-Paz, "mixup: Beyond empirical risk minimization," in *Proc. Int. Conf. Learn. Represent.*, 2018, pp. 1–13.

[25] S. Yun, D. Han, S. J. Oh, S. Chun, J. Choe, and Y. Yoo, "CutMix: Regularization strategy to train strong classifiers with localizable features," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Oct. 2019, pp. 6023–6032.

[26] E. D. Cubuk, B. Zoph, D. Mané, V. Vasudevan, and Q. V. Le, "AutoAugment: Learning augmentation strategies from data," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2019, pp. 113–123.

[27] D. Ho, E. Liang, X. Chen, I. Stoica, and P. Abbeel, "Population based augmentation: Efficient learning of augmentation policy schedules," in *Proc. Int. Conf. Mach. Learn.*, 2019, pp. 2731–2741.

[28] E. D. Cubuk, B. Zoph, J. Shlens, and Q. V. Le, "RandAugment: Practical automated data augmentation with a reduced search space," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. Workshops*, Seattle, WA, USA, Jun. 2020, pp. 702–703.

[29] S. Ioffe and C. Szegedy, "Batch normalization: Accelerating deep network training by reducing internal covariate shift," in *Proc. Int. Conf. Mach. Learn. (ICML)*, 2015, pp. 448–456.

[30] X. Chen, Y. Bin, N. Sang, and C. Gao, "Scale pyramid network for crowd counting," in *Proc. IEEE Winter Conf. Appl. Comput. Vis. (WACV)*, 2019, pp. 1941–1950.

[31] X. Cao, Z. Wang, Y. Zhao, and F. Su, "Scale aggregation network for accurate and efficient crowd counting," in *Proc. Eur. Conf. Comput. Vis. (ECCV)*, 2018, pp. 734–750.

[32] W. Liu, M. Salzmann, and P. Fua, "Context-aware crowd counting," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2019, pp. 5099–5108.

[33] D. Guo, K. Li, Z.-J. Zha, and M. Wang, "DADNet: Dilated-attention-deformable ConvNet for crowd counting," in *Proc. 27th ACM Int. Conf. Multimedia*, Oct. 2019, pp. 1823–1832.

[34] X. Wang, R. Lv, Y. Zhao, T. Yang, and Q. Ruan, "Multi-scale context aggregation network with attention-guided for crowd counting," in *Proc. 15th IEEE Int. Conf. Signal Process. (ICSP)*, vol. 1, Dec. 2020, pp. 240–245.

[35] Y. Yang, G. Li, D. Du, Q. Huang, and N. Sebe, "Embedding perspective analysis into multi-column convolutional neural network for crowd counting," *IEEE Trans. Image Process.*, vol. 30, pp. 1395–1407, 2021.

[36] N. Liu, Y. Long, C. Zou, Q. Niu, L. Pan, and H. Wu, "ADCrowdNet: An attention-injective deformable convolutional network for crowd understanding," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2019, pp. 3225–3234.

[37] Z. Zhao, M. Shi, X. Zhao, and L. Li, "Active crowd counting with limited supervision," in *Proc. Eur. Conf. Comput. Vis. (ECCV)*, Aug. 2020, pp. 565–581.

[38] D. Berthelot, N. Carlini, I. Goodfellow, N. Papernot, A. Oliver, and C. Raffel, "MixMatch: A holistic approach to semi-supervised learning," 2019, *arXiv:1905.02249*.

[39] D. Berthelot et al., "Remixmatch: Semi-supervised learning with distribution alignment and augmentation anchoring," in *Proc. ICLR*, 2019, pp. 1–13.

[40] Q. Xie, M.-T. Luong, E. Hovy, and Q. V. Le, "Self-training with noisy student improves ImageNet classification," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2020, pp. 10687–10698.

[41] K. Sohn et al., "FixMatch: Simplifying semi-supervised learning with consistency and confidence," in *Proc. Adv. Neural Inf. Process. Syst. (NIPS)*, 2020, pp. 596–608.

[42] T. DeVries and G. W. Taylor, "Improved regularization of convolutional neural networks with cutout," 2017, *arXiv:1708.04552*.

[43] I. J. Goodfellow, J. Shlens, and C. Szegedy, "Explaining and harnessing adversarial examples," in *Proc. ICLR*, 2015, pp. 1–11.

[44] A. Kurakin, I. J. Goodfellow, and S. Bengio, "Adversarial examples in the physical world," in *Proc. Workshop Track 5th Int. Conf. Learn. Represent. (ICLR)*, Apr. 2017, pp. 1–14.

[45] A. Madry, A. Makelov, L. Schmidt, D. Tsipras, and A. Vladu, "Towards deep learning models resistant to adversarial attacks," in *Proc. Int. Conf. Learn. Represent.*, Apr. 2018, pp. 1–10.

[46] J.-B. Grill et al., "Bootstrap your own latent—A new approach to self-supervised learning," in *Proc. Adv. Neural Inf. Process. Syst. (NeurIPS)*, vol. 33, 2020, pp. 21271–21284.

[47] Z. Cai, A. Ravichandran, S. Maji, C. Fowlkes, Z. Tu, and S. Soatto, "Exponential moving average normalization for self-supervised and semi-supervised learning," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2021, pp. 194–203.

[48] R. Geirhos, P. Rubisch, C. Michaelis, M. Bethge, F. A. Wichmann, and W. Brendel, "ImageNet-trained CNNs are biased towards texture; Increasing shape bias improves accuracy and robustness," in *Proc. ICLR*, 2019, pp. 1–11.

[49] Q. Wang, J. Gao, W. Lin, and X. Li, "NWPU-Crowd: A large-scale benchmark for crowd counting and localization," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 43, no. 6, pp. 2141–2149, Jun. 2021.

[50] A. A. Efros and W. T. Freeman, "Image quilting for texture synthesis and transfer," in *Proc. ACM Conf. Comp. Graph.*, 2001, pp. 341–346.

[51] J. M. Springer, M. Mitchell, and G. T. Kenyon, "Adversarial perturbations are not so weird: Entanglement of robust and non-robust features in neural network classifiers," 2021, *arXiv:2102.05110*.

[52] A. Shafahi et al., "Adversarial training for free!" in *Proc. Adv. Neural Inf. Process. Syst. (NeurIPS)*, vol. 32, 2019, pp. 1–11.

[53] Y. Wu and J. Johnson, "Rethinking 'batch' in BatchNorm," 2021, *arXiv:2105.07576*.

[54] X. Wang, Y. Jin, M. Long, J. Wang, and M. Jordan, "Transferable normalization: Towards improving transferability of deep neural networks," in *Proc. Adv. Neural Inf. Process. Syst. (NeurIPS)*, vol. 32, 2019, pp. 1–11.

[55] C. Xie, M. Tan, B. Gong, J. Wang, A. L. Yuille, and Q. V. Le, "Adversarial examples improve image recognition," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2020, pp. 819–828.

[56] J. Yuan, Y. Liu, C. Shen, Z. Wang, and H. Li, "A simple baseline for semi-supervised semantic segmentation with strong data augmentation," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2021, pp. 8229–8238.

[57] K. He, H. Fan, Y. Wu, S. Xie, and R. Girshick, "Momentum contrast for unsupervised visual representation learning," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2020, pp. 9729–9738.

[58] H. Idrees et al., "Composition loss for counting, density map estimation and localization in dense crowds," in *Proc. Eur. Conf. Comput. Vis.*, 2018, pp. 532–546.

[59] V. Sindagi, R. Yasarla, and V. Patel, "Pushing the frontiers of unconstrained crowd counting: New dataset and benchmark method," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2019, pp. 1221–1231.

[60] D.-H. Lee et al., "Pseudo-label: The simple and efficient semi-supervised learning method for deep neural networks," in *Proc. Int. Conf. Mach. Learn. (ICML)*, 2013, vol. 3, no. 2, p. 896.