applied sciences

MDPI

Article

Cross Domain Adaptation of Crowd Counting with Model-Agnostic Meta-Learning

Xiaoyu Hou [1,2], Jihui Xu [1,2], Jinming Wu [1,2] and Huaiyu Xu [1,*]

[1] Shanghai Advanced Research Institute, Chinese Academy of Sciences, Shanghai 201210, China; houxiaoyu@sari.ac.cn (X.H.); xujh@sari.ac.cn (J.X.); wujinming@sari.ac.cn (J.W.)
[2] University of Chinese Academy of Sciences, Beijing 100049, China
* Correspondence: xuhuaiyu@sari.ac.cn

Abstract: Counting people in crowd scenarios is extensively conducted in drone inspections, video surveillance, and public safety applications. Today, crowd count algorithms with supervised learning have improved significantly, but with a reliance on a large amount of manual annotation. However, in real world scenarios, different photo angles, exposures, location heights, complex backgrounds, and limited annotation data lead to supervised learning methods not working satisfactorily, plus many of them suffer from overfitting problems. To address the above issues, we focus on training synthetic crowd data and investigate how to transfer information to real-world datasets while reducing the need for manual annotation. CNN-based crowd-counting algorithms usually consist of feature extraction, density estimation, and count regression. To improve the domain adaptation in feature extraction, we propose an adaptive domain-invariant feature extracting module. Meanwhile, after taking inspiration from recent innovative meta-learning, we present a dynamic-β MAML algorithm to generate a density map in unseen novel scenes and render the density estimation model more universal. Finally, we use a counting map refiner to optimize the coarse density map transformation into a fine density map and then regress the crowd number. Extensive experiments show that our proposed domain adaptation- and model-generalization methods can effectively suppress domain gaps and produce elaborate density maps in cross-domain crowd-counting scenarios. We demonstrate that the proposals in our paper outperform current state-of-the-art techniques.

Keywords: crowd counting; domain adaptation; cross-domain; meta-learning; synthetic dataset

Citation: Hou, X.; Xu, J.; Wu, J.; Xu, H. Cross Domain Adaptation of Crowd Counting with Model-Agnostic Meta-Learning. *Appl. Sci.* **2021**, *11*, 12037. https://doi.org/10.3390/app112412037

Academic Editor: Fabio La Foresta

Received: 2 November 2021
Accepted: 14 December 2021
Published: 17 December 2021

1. Introduction

Crowd counting has become an essential component in crowd analysis, and attracts increasing attention in computer vision research [1,2]. It has many applications, including drone inspections, video surveillance, traffic flow analysis, and public safety [3]. Usually, crowd counting is regarded as a pixel-level estimation problem [4]. Deep convolutional networks first extract feature maps from images and the density value of each feature pixel is then predicted. By summing the densities of all the feature-map pixels, we can regress the final counting result [5]. Crowd counting is essential in these scenarios. Recently, supervision-based crowd analysis algorithms [6–8] have benefited from the power of deep learning to accomplish remarkable improvement. However, these algorithms have obvious limitations, and current popular crowd-counting datasets do not fully satisfy the demand [9].

In many real world scenarios, different image angles, exposures, location heights, and complex backgrounds, along with limited annotation data, lead to supervised learning methods that do not work satisfactorily, and many suffer from overfitting problems [10]. In addition, abundant labeled training data, which is costly and time-consuming to produce, is the basis of better performance. Furthermore, there are inevitably some incorrect annotations within the popular dataset, such as in Shanghai Tech [6] and UCF_CC [11] samples.

In this paper, we mainly focus on the problem of cross-domain crowd counting with limited labeled data. Generally speaking, the accuracy rate will drop drastically in cross-domain scenarios due to domain-shift issues. Therefore, many researchers pay more attention to synthetic data, hoping to utilize massive amounts of synthetic images with labeled data to adapt domains and train meta-learning models to reduce manual labeling in new scenarios. Multiple challenges lead to accurate and efficient crowd-counting results in this field.

One such challenge is extracting domain-invariant features to align the source and target domains at the feature level. Due to differences between synthetic data and real-world images, there are issues involving domain gaps, which significantly degrade performance. To align the domain gaps between synthetic and real-world datasets, Wang et al. [12] proposed a CycleGAN-based method, which transfers the image styles and extracts the domain-invariant features. Gao et al. [5] proposed an adaptive domain method for crowd counting, which focuses on transferring domain-invariant data from a source domain to a target domain.

Another challenge is improving the generalization of the crowd-counting meta-learning model and accelerate convergence. Theoretically, the more meta-learning scenarios exist for training, the higher the model accuracy. However, such model will consume more time in training. Reddy et al. [13] proposed a new approach for few-shot scenes, which improves the generalizable crowd-counting model, supporting the idea of learning to learn [14].

Most studies aim to explore effective methods with only a small amount of labeled training data needed to transfer the knowledge of crowd-counting models from source domains to target domains. Usually, source domains use a synthetic dataset, while target domains utilize a real-world dataset [1]. Therefore, we propose the method with model-agnostic meta-learning for cross-domain adaptation scenarios around the aforementioned key points. The paper's contribution is summarized below:

(1) To improve the model's generalization ability, in the density map estimation phase, we propose a meta-learning-based method, which accelerates the model's convergence in few-shot scenes with the dynamic meta-learning rate β.
(2) In cross-domain scenarios, domain-invariant feature extraction is essential to align the source and target domains. We propose an adaptive domain-invariant feature extracting module based on gradient reversal layer (GRL) to perform domain adaptation.
(3) To conclude, we discuss the effectiveness of domain adaptation with two critical model generalization phases in crowd-counting scenarios: feature-map extraction and density-map estimation. Experiments show that the methods we propose in this paper can improve performance over the baseline and achieve state-of-the-art performance.

2. Related Work

2.1. Crowd Counting

In crowd-analysis scenarios, crowd counting is the essential component when aiming to calculate the crowd number. In the last decade, several methods have emerged to solve the problem of crowd counting. Many traditional algorithms have applied hand-crafted features to detect people from images. Ref. [15] introduced the Hough forest to perform a generalized Hough transformation for object detection. Ref. [16] boosted several weak part detectors based on extracted features, and all detector responses were combined for counting. Ref. [17] combined mosaic images with a foreground segmentation module and head–shoulder detector to accurately estimate pedestrian counts. While early methods can satisfactorily solve the occlusion problem, they are conducted at the expense of spatial information. Various density estimation-based methods are proposed [18]. These methods do not need to detect every object, as they estimate the image density and calculate the area in the density map to obtain the quantity within that area. Ref. [19] proposed a patch-based method to learn patch features and the nonlinear mapping of corresponding objects in the patch. To improve estimation accuracy and speed, they used random forest

regression. Ref. [20] combined deep and shallow, fully convolutional networks, for which the high-level and low-level semantics complemented each other to predict higher-quality density maps. Ref. [21] proposed solving accuracy problems in the generation of density maps through multi-scale averaging. Ref. [6] attempted a multi-column-based architecture (MCNN) used with images of dense crowds and an angle of view. In a different approach, Ref. [22] presented a universal model for crowd counting across scenes and datasets. The model learns to obtain the optimal image rescaling factors for alignment, by minimizing the distances between their scale distributions. Ref. [23] proposed an unsupervised domain adaptation problem for video-based crowd counting.

2.2. Domain Adaptation

Domain adaptation is a representative method in transfer learning, which utilizes information-rich samples from the source domain to improve the performance of the model in the target domain. Many methods [24–28] have been proposed to reduce the domain gap. An unsupervised domain adaptation [29] has been proposed for semantic segmentation for the first time. Adversarial-based DA methods are becoming more and more popular in recent years. Sankaranarayanan et al. [30] propose a joint adversarial learning approach, preserving the learned embedding to represent the target distribution. Hoffman et al. Ref. [31] propose a novel discriminatively-trained cycle-consistent adversarial domain adaptation model (CyCADA) with cycle-consistency constraints. Ref. [32] presents a multi-level adversarial network in multi-level layers for semantic segmentation. Ref. [33] proposes a novel self-supervised framework to solve the distributed multi-source domain adaptation problem, referred as self-supervised federated domain adaptation (SFDA), which utilizes multi-domain model generalization balance (MDMGB) to aggregate the models from multiple source domains. To the domain-shift by learning domain-invariant representations, Ref. [34] designed a method for learning domain-invariant local feature patterns and jointly aligning holistic and local feature statistics. In our approach, we propose an adaptive domain-invariant features-extracting module based on gradient reversal layer (GRL) to perform domain adaptation.

2.3. Few-Shot Learning

Few-shot learning (FSL) aims to learn from very few labeled examples to complete a task. In terms of what prior knowledge is required, recent FSL work can be classified into three types: multitask learning [35,36], embedding learning [37–39], and generative modeling [40–42]. Ref. [43] builds a shared two-task network for general information and to learn task-specific information from different final layers. Luo et al. [44] propose the possibility of domain adaptation with a limited sample data. Ref. [45] propose training two different networks; one, being the source-domain training, and the other, being the target-domain training, which are then aligned through regularization to achieve the domain adaptation of the two networks. Ref. [46] used auto-regressive models to enable practical few-shot density estimation. This measurement-based method [47,48] usually uses the similarity and consistency between uniform category data points to learn the distance function and measure whether the data points are similar.

2.4. Synthetic Dataset

Data collection and annotation is costly and time-consuming work, which limits most current deep learning approaches. Synthetic content generation [49–51] is considered a promising solution, since all labels are available in the graphics engine. Some excellent synthetic datasets have recently emerged, ranging from driving scenes [52], and crowd counting [53], to optical flow estimation [54]. Ref. [42] have offered an approach that can learn to modify the attributes of scene graphs obtained from probabilistic scene grammars. Ref. [55] propose using GTA-generated synthetic data as training samples for semantic segmentation training in urban scenes. Ref. [56] presented a novel large-scale human pose-estimating dataset, rendered from 3D sequences of human motion-capture data. Ref. [57]

provides a benchmark data set generated by specific low-level features to generate synthetic images for training the attention model.

3. Methods

In this paper, we break down the task of crowd counting into several parts: feature extraction, density estimation, and the counting map. We also study the different aspects of model generalization approaches. Many algorithms achieve excellent results when their training and testing data are in the same domain. However, in cross-domain scenarios [58], there are domain-shift issues, which is a result of training and testing on different domains. Generally, the basic steps of crowd-counting algorithms' flow [59] consists of three parts: a feature extractor, a density estimator, and a crowd-counting mapper, as shown in Figure 1. In most CNN-based algorithms, such as [6,60,61], the three parts are trained from end to end. Taking CSRNet [60] as an example, whose first ten network layers are utilized as feature extractors, and whose remaining parts are used as density estimators. Following the idea of unsupervised domain adaptation [28], we propose an adaptive domain-invariant feature extracting module to align the two domains along a feature level. Moreover, to make the density estimator more universal, we propose dynamic-β MAML, based on the idea of Alpha-MAML [62,63]. Finally, we study the counting map refiner, which transforms a density map from coarse to refined, regresses the crowd number based on a synthetic dataset, and applies the refiner to new domains.

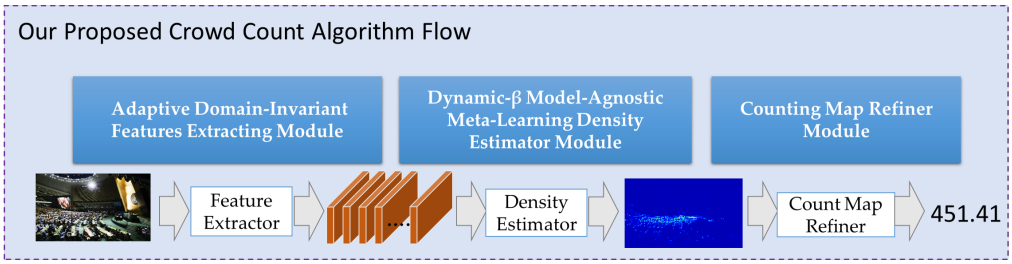


Figure 1. Comparison of the main proposed algorithm flow with general crowd-counting algorithm flow. The three modules proposed to enhance model generalization capability: Adaptive Domain-Invariant Features Extracting module, Dynamic-β Model-Agnostic Meta-Learning Density Estimator Module, and Counting Map Refiner Module.

3.1. Density-Map Estimator Module Based on Dynamic-β MAML

With the rapid development of computer science, powerful computing and significant data volume have greatly improved the accuracy of computer-vision algorithms, and with great success [64]. However, there are still many challenges, one of which is model generalization. When there is limited labeled data, it is essential to apply a training method that improves model generalization [52]. The supervised learning algorithm aims to learn a function between image data and labeled data. Moreover, the meta-learning algorithm is trained in different tasks, each containing a training set and a testing set. Originating from meta-learning, few-shot learning solves the predictions problem with limited training samples, while also enabling the model to adapt to new, unseen scenes with little or no labeled data. In most real-world scenarios there is limited training data, so, improving the model generalization to adapt more scenes quickly is an area worth studying, particularly in crowd-counting problems. There are naturally different domains because of the angle, location, exposure, and position of photos. Thus, we mainly focus on the meta-learning model generalization method for estimating density in this section. Inspired by the alpha model-agnostic meta-learning algorithm, we propose a meta-learning-based approach to generate a dynamic learning rate for faster convergence. Therefore, this will allow the model to quickly adapt to new scenes.

The MAML algorithm [62] is model-agnostic, which means that it is compatible with deep learning models trained with gradient descent. Therefore, we studied the approaches to adaptive crowd counting based on the MAML algorithm with a few samples. In this

paper, we chose the GCC dataset [53], consisting of 15,212 synthetic images with different scenes: rainy, cloudy, night, and so on.

We combined images in different scenes to organize multiple tasks for meta-learning [65]. The framework of the density-map estimator, based on model-agnostic meta-learning, is shown as Figure 2. The meta-learning aims to learn a mapping function $g(\cdot)$, trained on a set of tasks. Each task contains a training dataset and a testing dataset. In Figure 2, the meta-learning is divided into two phases; the meta-train phase, and the meta-test phase. We aimed to improve the model generalization ability based on model-agnostic meta-learning. Furthermore, as the feature extraction parameters are fixed, we will discuss an adaptive feature extraction method in the next section, but the density estimating parameters are trainable.

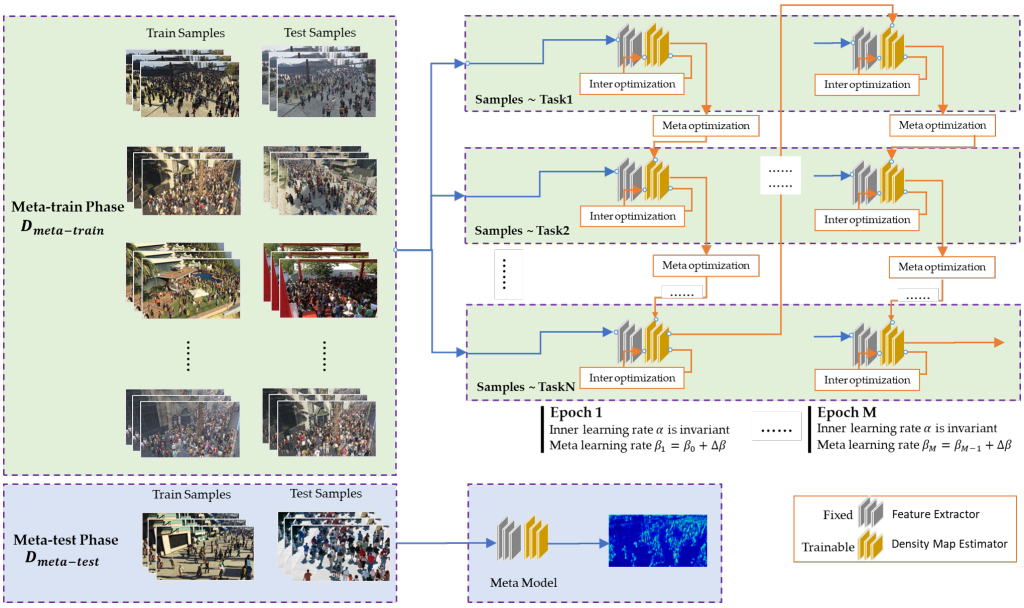


Figure 2. The details of meta-learning for the density estimator. There are two parts to optimize the model parameters: inner-optimization and meta-optimization. Inner-optimization is over each task, and the meta-optimization is across different tasks.

The MAML algorithm aims to adapt to a new task, $\mathcal{T}_t$, with SGD, given the model parameters θ. Originating from model-agnostic meta-learning, the key to our learning procedure is to generate the initial parameters, θ, to adapt to new scenes quickly. Available domains $\mathcal{D}_{meta-learning}$ are split into sets of meta-train domains $\mathcal{D}_{meta-train}$ and meta-test domains $\mathcal{D}_{meta-test}$. In our study, the feature extractor is defined as $f(\cdot)$ and the density-map estimator is defined as $g(\cdot)$. The $p(\mathcal{T})$ is the distribution over tasks. $\mathcal{T}_{train(t)}$ and $\mathcal{T}_{test(t)}$ denote the training and testing datasets, respectively, corresponding with task t. The basic MAML algorithm is formulated as below:

$$\hat{\theta}_t = \theta - \alpha \nabla_\theta \mathcal{L}_{\mathcal{T}_{train}}(f_\theta) \tag{1}$$

where, t is the task number, and α is the inter-learning rate. The tasks are sampled from the meta-train domain $\mathcal{D}_{meta-train}$. Moreover, the model aims to optimize the parameters θ such that with just one SGD step, it can adapt to the new task for optimization.

$$\theta_t = \theta - \beta \nabla_\theta \mathcal{L}_{\mathcal{T}_{test}}(f_{\hat{\theta}_t}) \tag{2}$$

where, β is the meta-learning rate, which produces an algorithm that learns an initialization of θ that is useful in efficiently adapting new tasks with a small number of iterations. In the MAML algorithm, there are two learning rates: α and β, which are updated with meta-training and meta-testing iteration. In this paper, we follow the idea of Alpha-

MAML [63] and conduct experiments on the two learning rates. The task-inner-learning rate, α, is internal and affects the iteration result, while the meta-learning rate, β, is external and improves the result when applied with the alpha-MAML algorithm. We derived an updated rule for the meta-learning rate, β, which can be computed as below:

$$\begin{aligned}\frac{\partial \mathcal{L}_{\mathcal{T}_{test(i)}}(f_{\hat{\theta}_i})}{\partial \beta} &= \frac{\partial \mathcal{L}_{\mathcal{T}_{test(i)}}(f_{\hat{\theta}_i})}{\partial \theta_{i-1}} \cdot \frac{\partial \theta_{i-1}}{\partial \beta} \\ &= \nabla_{\theta_{i-1}} \mathcal{L}_{\mathcal{T}_{meta-test(i)}}(f_{\hat{\theta}_i}) \cdot (-\nabla_{\theta_{i-2}} \mathcal{L}_{\mathcal{T}_{meta-test(i-1)}}(f_{\hat{\theta}_(i-1)}))\end{aligned} \tag{3}$$

where i is the number of iterations. We can estimate the β_i as shown below:

$$\beta_i = \beta_{i-1} + \delta_{hyper} \nabla_{\theta_{i-1}} \mathcal{L}_{\mathcal{T}_{meta-test(i)}}(f_{\hat{\theta}_i}) \cdot \nabla_{\theta_{i-2}} \mathcal{L}_{\mathcal{T}_{meta-test(i-1)}}(f_{\hat{\theta}_i}) \tag{4}$$

We randomly divided the synthetic data into a set of N tasks, where each task consisted of both training data and testing data. For the i-th training iteration, we denote the sample number as $K\{1,5\}$. The algorithm we propose refers to using a small number of samples to learn a meta-learning model. The dynamic meta-learning rate makes the model faster, and the improved MAML algorithm increases the generalization of the model. The whole algorithm is shown in Algorithm 1 as below:

Algorithm 1 Dynamic-β MAML

Input:
α is the fixed inter learning rate
β_0 is the initial meta learning rate
δ_{hyper} is the hyper-gradient learning rates randomly initialize θ
M is the count of training iterations
Output:
θ is the parameters of meta-learning model

for i in range(0,M) **do**
  for each sample batch $\mathcal{T}_t \sim \mathcal{D}_{meta-learning}$ **do**
    evaluate $\nabla_\theta \mathcal{L}_{\mathcal{D}_{meta-train}}(f_\theta)$ with respect to K examples.
    Compute adapted parameters with gradient descent $\hat{\theta} = \theta - \alpha \nabla_\theta \mathcal{L}_{\mathcal{D}_{meta-train}}(f_\theta)$
  end for
  Compute the meta-learning rate β:
  $\beta_i = \beta_{i-1} + \delta_{hyper} \sum_{\mathcal{T}_t \in \mathcal{D}_{meta-learning}} \nabla_{\theta_{i-1}} \mathcal{L}_{\mathcal{T}_{meta-test(t)}}(f_{\hat{\theta}_t}) \cdot \nabla_{\theta_{i-2}} \mathcal{L}_{\mathcal{T}_{meta-test(t-1)}}(f_{\hat{\theta}_t})$
  $\theta_i = \theta_{i-1} - \beta_i \sum_{\mathcal{T}_t \in \mathcal{D}_{meta-test}} \nabla_{\theta_{i-1}} \mathcal{L}_{\mathcal{T}_{meta-test(t)}}(f_{\hat{\theta}_t})$
end for

In this section, we formulate the crowd-counting density estimation as a few-shot learning problem, given a set of datasets where $\mathcal{D}_{train}$ and $\mathcal{D}_{test}$ are the training and test sets, respectively. The CSRNet network contains several dilated convolutional layers to regress the density map according to the inputted images for different task-specific data. For the network architecture, we only trained the CSRNet [60] density-map estimator function model parameters, and the other function parameters were fixed. The density estimation model parameters are trainable in meta-learning iterations. The proposed algorithm can dynamically adjust the learning rate of meta-learning to, in turn, dynamically adjust the learning rate in each iteration. This will improve the algorithm convergence speed and help the model to adapt to new scenes with only a few labeled images.

3.2. Adaptive Domain-Invariant Feature-Extracting Module

The feature extraction module is an essential part of the vision algorithm [66]. In cross-domain scenarios, the training dataset is represented as the source domain for training, and testing or predicting is then performed in the target domain. Without domain adaptation, accuracy and performance will be significantly reduced [67]. The different

feature distribution between the source and target domains causes a decrease in accuracy in cross-domain scenarios. Consequently, aligning the two domains at the feature level will create a more adaptive model [68]. In this section, we study the adversarial training approaches used to extract domain-invariant features, and we apply the separate feature-extracting model in cross-domain scenes. For the source domain, we preferred the most popular synthetic dataset in crowd-counting scenarios, the GCC datasets, created for the GTA5 computer game, while, for the target domain, we chose real-world datasets, such as NWPU-crowd, Shanghai A, and UCF, etc. The GTA5 dataset exploits UE4 to construct synthetic street-scene data (different weather conditions, timestamps, and capacities) for crowd-counting tasks. The advantage of the synthetic dataset is that there is no need to manually label the data, and, when the image is synthesized, the objects in the image already have accurate location information [5,53].

In cross-domain scenarios, we sought to train a feature-extracting module to align the feature distribution and extract the domain-invariant features representation. Given the labeled source domain $\mathcal{D}_S = \{(x_i^S, y_i^S)\}_{i=1}^{N_S}$, x_i^S and y_i^S denote the i-th crowd image and corresponding label. Furthermore, we had access to the target domain $\mathcal{D}_T = \{(x_i^T)\}_{i=1}^{N_T}$ containing a set of unlabeled crowd images. We assumed that samples from the two domains are drawn from different distributions, and our goal was to align the two domains using the adversarial training method.

If the crowd-counting model is trained in different domains, the parameters of each model for extracting feature representations are different. In cross-domain scenarios, the domain-invariant feature representation needs to be extracted to achieve domain adaptation. We adhered to the idea of extracting domain-invariant feature representation and designing training algorithms in two domains by an adversarial method [69,70]. In both domains, we used $\mathcal{H}$-divergence to measure the distribution distance of the two sets of samples. As shown in Figure 3, we trained the adversarial discriminator module to distinguish whether the feature is generated from a source or target domain.

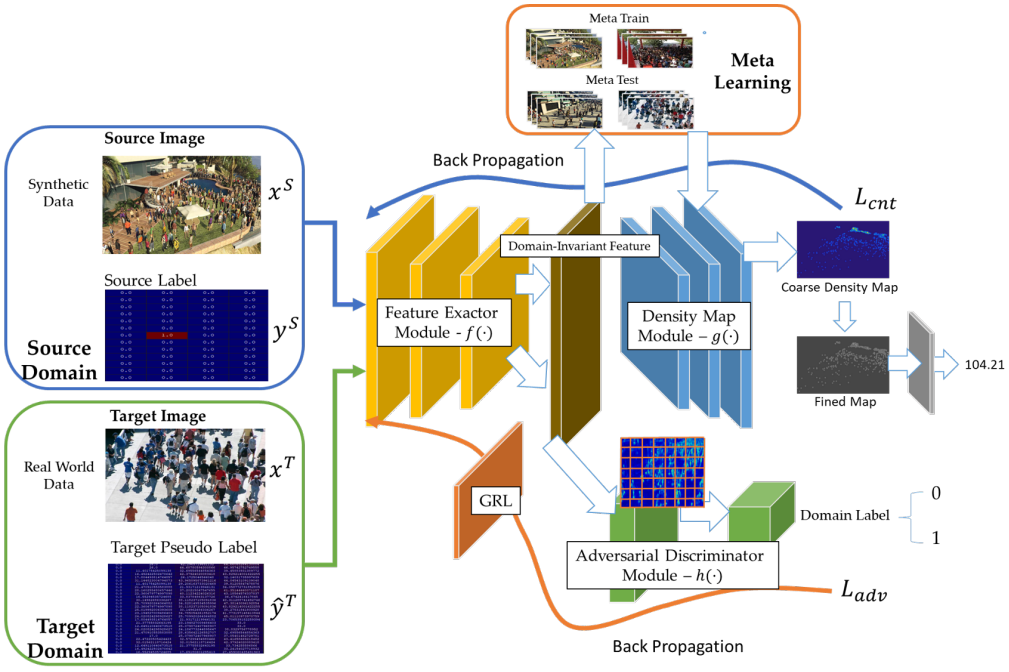


Figure 3. The domain invariant feature representation module framework, which allows models to extract domain-invariant features, to reduce the impact of domain gaps between the source and target domains.

In this feature-extraction module, the source domain contains synthetic data and x^S and y^S are the source image and source label, respectively. The target domain contains

real-world data and x^T and $\hat{y}^T$ are the target image and pseudo-target label, respectively. These are coarse and predicted by the model trained in the source domain. We define $h : x \to \{0, 1\}$ as the adversarial discriminator, which aims to distinguish samples from the source domain or target domain. We denote the sample of source domain x_i^S as 0, and the target domain sample x_i^T as 1. We denote the method $h(\cdot)$ as the domain classifier, and the $\mathcal{H}$-divergence distance between the source domain and target domain is shown below:

$$\begin{aligned} \epsilon_S(h) &= E_{x\sim\mathcal{D}_S}[|h(x) - 0|] \\ \epsilon_T(h) &= E_{x\sim\mathcal{D}_T}[|h(x) - 1|] \\ d_{\mathcal{H}}(S, T) &= 2(1 - \min_{h\in\mathcal{H}}(\epsilon_S(h) + \epsilon_T(h))) \end{aligned} \tag{5}$$

where, $\epsilon_S(h)$ and $\epsilon_T(h)$ denote the prediction errors of $h(\cdot)$, predicting the domain origin, i.e., whether source or target domain. If the prediction error of the domain classifier is high, the two domains become closer and are harder to distinguish, so the distance between the two domains $d_{\mathcal{H}}(S, T)$ is inversely proportional to the error rate of the domain classifier $h(\cdot)$.

During the training phase, we integrated a gradient reversal layer (GRL) [67] into the feature extracting module. The GRL minimizes the objective function, and adjustment in the negative gradient direction maximizes the objective function. If the feature is adaptive to two different domains, the GRL will make the two domains as indistinguishable as possible. The feature itself gradually inclines towards domain adaptation, and will become a domain invariant feature [10,67]. Furthermore, to reduce the domain shift between different samples in the source and target domains, as per previous studies, we divided the output features into blocks. This is helpful in alleviating the effects of domain shifts such as lighting, exposure, position, scale, image style, and so on.

Crowd counting is a compromise of feature extraction and density estimation, which are considered pixel-wise regression problems, and the domain discriminator is designed to distinguish each pixel of the extracted feature maps. We used four convolution layers for the domain discriminator to generate two-dimensional scores to indicate the confidence with which we can distinguish the source and target domain. Thus, the loss function can be formulated as below:

$$\mathcal{L}(x^S, y^S, x^T) = \mathcal{L}_{cnt}(x^S, y^S) + \lambda\mathcal{L}_{adv}(f(x^T)) \tag{6}$$

where $\mathcal{L}_{cnt}$ is the standard MSE loss, and $\mathcal{L}_{adv}$ is the adversarial loss. λ is the weight to balance the losses. For the feature maps $f(x^S), f(x^T)$, we trained one image-level discriminator $h(\cdot)$. Through $h(\cdot)$, we can obtain the pixel-wise domain labels for the source and target domains, denoted as O^S and O^T. We utilized binary cross-entropy loss to optimize the discriminator $h(\cdot)$, which is formulated as:

$$\mathcal{L}_{f(\cdot)}(x^S, x^T) = -\sum_{x^S\in D^S}\sum_{w\in W}\sum_{h\in H}\log(p(f(x^S))) - \sum_{x^T\in D^T}\sum_{w\in W}\sum_{h\in H}\log(1 - p(f(x^T))) \tag{7}$$

where $f(\cdot)$ is the feature extracting component, $f(x^S)$ and $f(x^T)$ are two-dimensional feature maps of size H × W. $f(x^S)$ is the source input, and $f(x^T)$ is the target input. At the pixel level, we utilized $p(\cdot)$ as a soft-max function. To confuse $h(\cdot)$, we also added the inverse adversarial loss into the training phase. The formulation is shown as below:

$$\mathcal{L}_{adv}(x^T) = -\sum_{x^T\in D^T}\sum_{w\in W}\sum_{h\in H}\log(p(f(x^T))) \tag{8}$$

We used the adversarial loss $\mathcal{L}_{adv}$ to guide $f(\cdot)$ to fool the discriminator $h(\cdot)$, by which we effectively alleviated the domain gaps in cross-domain scenarios. This section propose using the adversarial method to train domain-invariant feature-extracting modules for two

different domains. With the help of feature visualization tools, the effect can be shown as Figure 4.

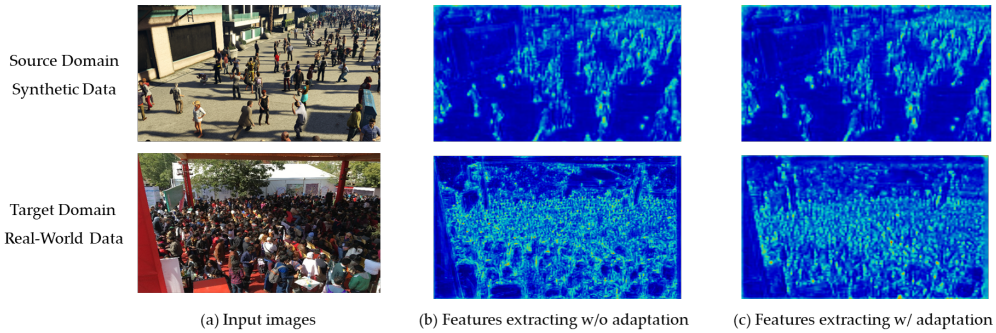


Figure 4. The comparison of feature extraction between the source and target domains. (**a**) shows the input images, (**b**) shows the feature extraction results without adaptation, and (**c**) shows the feature extraction results with adaptation. When using real-world images for testing, we compared the results in (**b**,**c**). Generated feature maps with adaptation will have less noise than those without adaptation.

3.3. Crowd-Counting Refined-Mapper Module

By introducing the feature-extraction and density-estimator modules above, we can generate a coarse density map. In this section, we mainly focus on refining the density map and regressing the accurate number. Coarse maps are always produced in cross-domain crowd counting. The first training was based on the GCC dataset and transformed the density map from coarse to refined, before predicting crowd counting in other real-world domains. The Figure 5 shows the structure of the counting-map refiner.

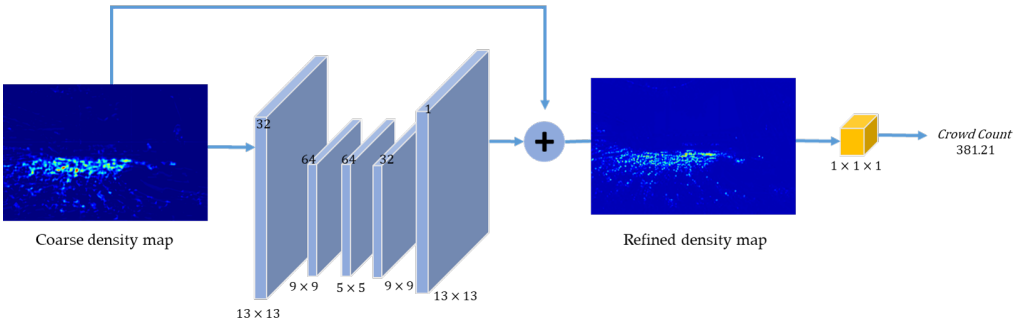


Figure 5. The mapping flow of the crowd-counting mapper from a coarse density map to a refined density map.

Given x^S and y^S, we trained the model using supervised learning, and the counting-map refiner was trained to predict the refined density map. We trained the crowd-counting refined mapper with y^S and $g(x^S)$, as shown in Figure 5. Next, we used the trained model on the target domain to generate the pseudo-labels $\hat{y}^T$ with x^T. To reduce noise in the density estimation, we utilized a 13×13 kernel to obtain the receptive fields. Following the idea of a single-column network, we subsequently designed a five-layer network as a regression layer of the same size as the original input-density map.

4. Experiments

This work studies two different model generalization approaches in crowd-counting tasks and proposes an adaptive crowd-counting framework for cross-domain scenarios. The crowd-counting algorithm, based on density estimation, mainly consists of three parts: feature extraction, density estimation, and count mapping. In cross-domain scenarios, the testing accuracy of the model trained in the source domain will drop considerably in the target domain without adaption. This issue is caused by domain shift. We took the

synthetic dataset as the source domain, as a synthetic dataset like GCC will generate data annotation points simultaneously when generating images, thus saving a lot of annotating work. Therefore, we propose several methods for adaptive crowd counting and finally conduct an ablation study analysis. This section will discuss the following aspects in detail:

- Verify whether our proposed density-map estimator, based on dynamic-β MAML, can accelerate convergence and improve crowd-counting performance in few-shot learning scenarios over the baseline and FSCC performances.
- Verify and evaluate the effectiveness of our proposed domain-invariant feature representation in cross-domain scenarios.
- Perform additional ablation studies on the efficacy of our proposed method, to verify the effectiveness of two key phases: feature extraction and density estimation.

We developed the crowd-counting algorithm based on the open-source crowd-counting project C3-Framework. The hardware environment we used was the Intel Core i7-6500k CPU 3.4 GHz with two TITAN RTX GPUs and 24gb of memory. We conducted the cross-domain adaption experiments from the GCC dataset to various real-world datasets, such as ShanghaiTech, UCF, NWPU-Crowd, and WorldExpo. Furthermore, in this paper, two metrics are used to evaluate accuracy: mean absolute error (MAE) and mean square error (MSE). They are defined as follows:

$$MAE = \frac{1}{N}\sum_{i}^{N}|z_i - \hat{z}_i| \tag{9}$$

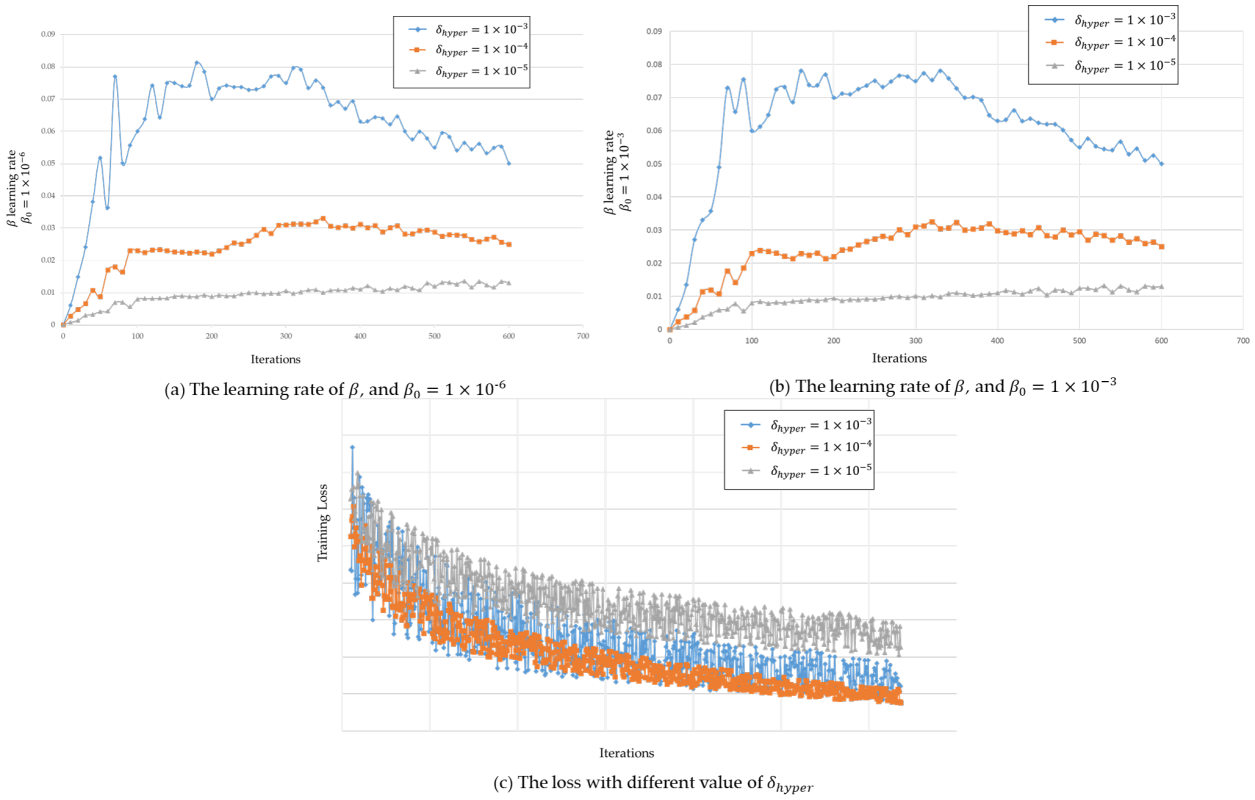
$$MSE = \sqrt{\frac{1}{N}\sum_{i}^{N}(z_i - \hat{z}_i)^2} \tag{10}$$

4.1. Evaluation of the Density-Map Estimator Based on Dynamic-β MAML

In this research, we used the synthetic data set GCC as the training set for meta-learning. Since GCC contains seven different weather scenes, we split these seven different scenes into different tasks. We fixed the feature extraction model parameters, and re-used the CSRNet feature-extracting function, or the component proposed in the previous section. The aim was to train the function $g(\cdot)$ to generate density maps by meta-learning, and we trained the density-map estimator with tasks containing a training set and testing set.

To evaluate the proposed dynamic-β MAML algorithm performance, we ran a series of training experiments to study the effect of meta-learning rate on density-estimator loss. As shown in Figure 6, for a fair comparison, we recorded the meta-learning rate β changes with the top 600 iterations. It was found that, no matter the initial value , under the influence of different hyperparameters δ_{hyper} the meta-learning rate β will show different results in the learning process. From Figure 6, our proposed dynamic-β MAML algorithm shows faster convergence with the meta-learning rate, specifically for $\nabla_{hyper} = 1e-4$. We utilized the standard crowd-counting model trained in a supervised setting as the baseline [60]. When the training was complete, the model was evaluated directly on target scenes without adaption. Simultaneously, we chose FSCC [13] for the comparative analysis. FSCC is a state-of-the-art algorithm in few-shot adaptive crowd-counting scenarios. Table 1 shows the experimental results.

From the table above, our proposed method can achieve MAE 16.13 and MSE 22.93 for 1-shot, and MAE 16.47 and MSE 23.48 for 5-shot. FSCC is state-of-the-art in few-shot crowd-counting problems. Our method exhibited better performance than FSCC.



(a) The learning rate of β, and $\beta_0 = 1 \times 10^{-6}$

(b) The learning rate of β, and $\beta_0 = 1 \times 10^{-3}$

(c) The loss with different value of δ_{hyper}

Figure 6. (**a**) shows the meta-learning rate β with initial value $\beta_0 = 1 \times 10^{-6}$, (**b**) shows the meta-learning rate β with initial value $\beta_0 = 1 \times 10^{-3}$. (**c**) shows the loss with different values of hyper-parameter δ_{hyper}.

Table 1. Verification of the proposed dynamic-β MAML algorithm with $\delta_{hyper} = 1 \times 10^{-4}$ on dataset GCC by comparing with baseline and FSCC [13] method.

Tasks	Method	1-Shot ($K = 1$)		5-Shot ($K = 5$)	
		MAE	MSE	MAE	MSE
Scene1: Clear	Baseline	23.11	34.16	22.54	33.24
	FSCC	19.33	28.08	19.33	28.08
	Ours	18.34	26.48	18.34	26.48
Scene2: Clouds	Baseline	21.99	32.35	21.81	32.06
	FSCC	17.12	24.52	17.37	24.92
	Ours	17.18	24.62	17.39	24.96
Scene3: Rain	Baseline	14.87	20.89	14.88	20.91
	FSCC	11.31	15.16	11.87	16.06
	Ours	11.32	15.17	11.91	16.12
Scene4: Foggy	Baseline	32.00	48.45	32.13	48.66
	FSCC	15.99	22.70	15.78	22.36
	Ours	15.71	22.25	15.99	22.70
Scene5: Thunder	Baseline	39.56	60.59	39.13	59.90
	FSCC	20.44	29.86	20.31	29.65
	Ours	19.23	27.91	20.01	29.17
Scene6: Overcast	Baseline	19.44	28.25	19.72	28.70
	FSCC	14.30	19.97	14.28	19.94
	Ours	14.22	19.85	14.65	20.54
Scene7: Extra Sunny	Baseline	24.52	36.43	24.37	36.19
	FSCC	17.49	25.11	17.47	25.08
	Ours	16.94	24.23	17.03	24.37
Average	Baseline	25.07	37.30	24.94	37.10
	FSCC	16.57	23.63	16.63	23.73
	Ours	**16.13**	**22.93**	**16.47**	**23.48**

4.2. Evaluation of Domain-Invariant Feature Representation in Cross-Domain Scenarios

In cross-domain scenario problems, domain adaptation aims to solve the issue that a model trained on one domain cannot generalize to another domain due to domain-shift issues. This paper follows [67]'s idea and proposes a domain-adaptation method, at the feature layer, to extract domain-invariant feature representations to reduce domain gaps. This section describes the experiments of the proposed domain-invariant feature-extracting method, on GCC and three real-world datasets. The GCC dataset is presented as the source domain and the remaining three real-world datasets are defined as the target domain. The results are shown in Figure 7.

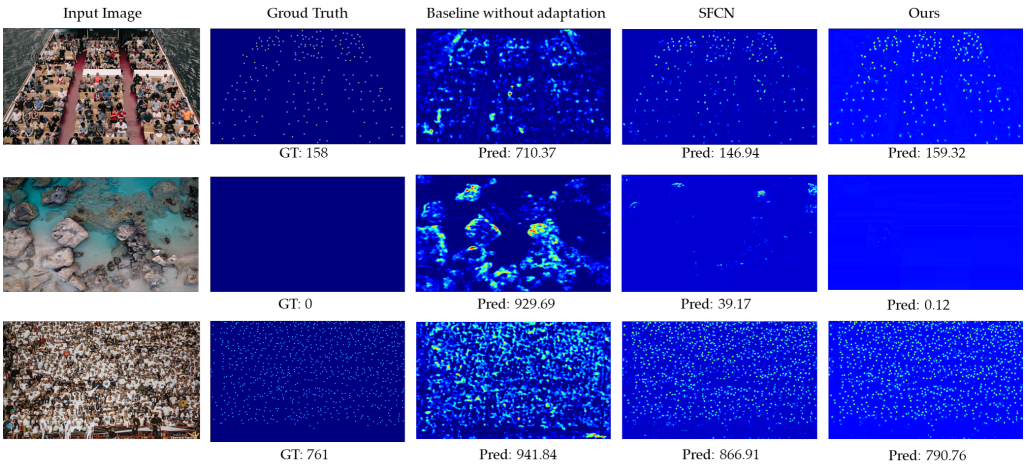


Figure 7. The comparison results between ground truth, baseline with no adaptation, SFCN [10], and our method.

We propose a new method for extracting domain-invariant features in cross-domain scenarios. First of all, we trained the crowd count models, which each consist of a feature extractor and density estimator. Then, we predicted the density maps, based on the previous phase, and generated pseudo-labels for the real-world datasets. Finally, the domain discriminator was trained, in adversarial mode, with a real-world dataset and GCC dataset. The discriminator was unable to distinguish whether the input images are from the GCC or real-world domain, and simultaneously, the domain-invariant feature extraction layer was reversed. In this section, we verify the efficacy of our proposed domain-invariant feature method. As shown in Figure 7, the methods we provide in this paper can be adapted for medium-sized crowd scenarios, as well as extremely large and empty scenarios. Therefore, we use CSRNet as the backbone of this module and test it on four different domains for satisfactory verification. Table 2 shows the results of the baseline without any adaptation, SFCN (state-of-the-art) [10], and our proposed cross-domain feature-extracting method. It is clear that our proposed method can improve performance in different real-world domains.

Table 2. The results for our methods on three different real-world datasets. We compare with baseline without adaptation and SFCN [10] (state-of-art) [10].

Method	SH-B		MALL		UCSD	
	MAE	MSE	MAE	MSE	MAE	MSE
NoAdapt	22.4	31.3	5.11	5.98	16.23	18.22
SFCN [10]	**17.1**	**26.1**	2.56	3.88	2.09	2.42
Ours	17.4	26.8	**2.55**	**3.81**	**2.03**	**2.41**

4.3. Ablation Study

In this section, to demonstrate the effectiveness of these modules in our approach, we performed ablation studies on the NWPU-Crowd dataset with cross-domain scenarios.

More concisely, we used FE to represent the adaptive domain-invariant features-extracting module, DE to represent the density-map estimator module based on dynamic-β MAML, and CM to represent the crowd-counting refined-mapper module. We utilized the different modules on the source domain GCC dataset and verified the performance on the target domain NWPU-Crowd dataset. As shown in Table 3, compared with the baseline, we obtained a significant improvement, using only adaptation. Our proposed FE method, similar to CSRNet with adaptation, improved performance and reduced the MAE 4.12 and MSE 4.15, respectively. When DE was used to perform the model generalization module for density-map estimation, the improvement was significant, with a 0.75 and 1.12 improvement compared with FE only. When applying the FE + DE + CM module, the improvement was 1.65 and 1.84. The results indicate that the domain-alignment processing and model generalization, through performing feature extraction and density evaluation, proved effective in cross-domain scenarios. Finally, Figure 8 shows the visualization results of the real-world dataset. We selected different crowd-volume photos for the results visualization.

Table 3. Ablation Study: The performance of baseline with and without adaptation and our approach in cross-domain scenarios.

Method	GCC - > NWPU-Crowd	
	MAE	MSE
CSRNet w/o Adapt	86.12	148.32
CSRNet w/Adapt	45.84	91.12
Ours w/FE	41.72	80.97
Ours w/DE	40.97	78.85
Ours w/FE + CM	40.83	79.13
Ours w/FE + DE + CM	39.18	77.29

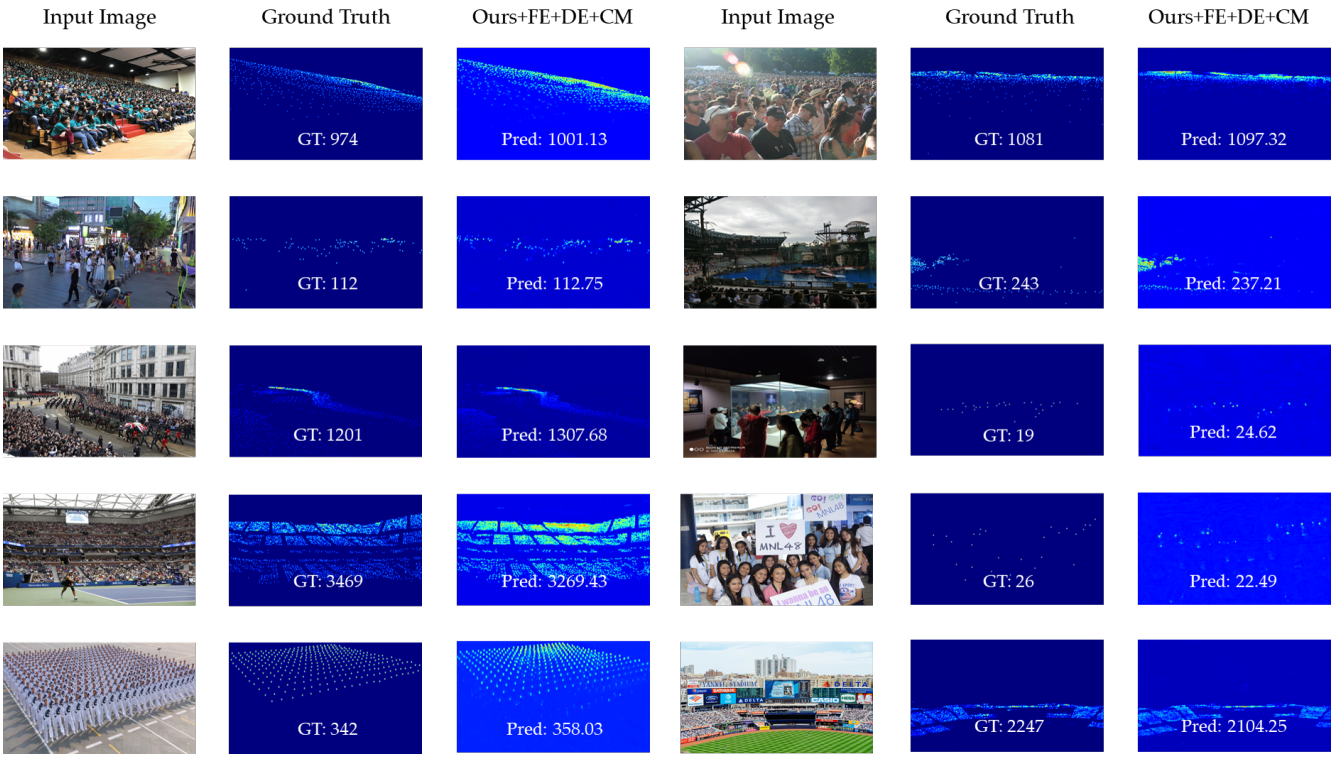


Figure 8. Visualization results of adaptation from GCC to real-world dataset.

In real-world scenarios, perspective is generally that of cameras on the ground or of drones in the sky. Nevertheless, the domain-shift issue affects performance due to weather, illumination, rotation, and scale changes. Our proposed method mainly focuses

on alleviating the issue above. Taking camera scenes from different angles, we first utilize UE4 to generate labeled annotations and synthetic images; the perspective, in the latter, was similar to a real-world position. Then, we used an adaptive domain-invariant feature-extracting module to extract the domain-invariant feature layer as a pre-training model. Next, we train the meta-learning model by using only a few labeled data. Finally, the network predicts the number of crowds.

4.4. Computational Cost Analysis

This section conducts the computational cost analysis on the whole work, in comparison with other methods. We divide the training phase of our method into two parts: training for cross-domain adaptation, to extract domain-invariant features, and training for the few-shot meta-learning model, to estimate the density map from the feature map. Thus, the whole training time consumed mainly concerns domain-invariant feature-extraction training and density-map-estimation training. In the adaptive domain-invariant feature-extracting module, we first pre-trained the feature extraction module on the synthetic dataset for 80 epochs and then generated pseudo labels for real-world images. We utilized synthetic and real images to train the domain-invariant feature-extracting layer for 80 epochs. In the following training, this layer can be integrated into the network. In the density-map-estimator module, we divided the synthetic data into multiple tasks and used the synthetic data to train a meta-learning model for 1000 epochs. As shown in Figure 9, we compare our proposed algorithm with other algorithms in terms of computational cost. In the domain adaptation phase, the number of epochs to convergence of our method was the same as that of SFCN and better than cycleGAN; and, in the phase of meta-learning training, the Dynamic-β MAML we have proposed can improve this convergence, such that the number of epochs to convergence of our method is better than SFCC and Reptile.

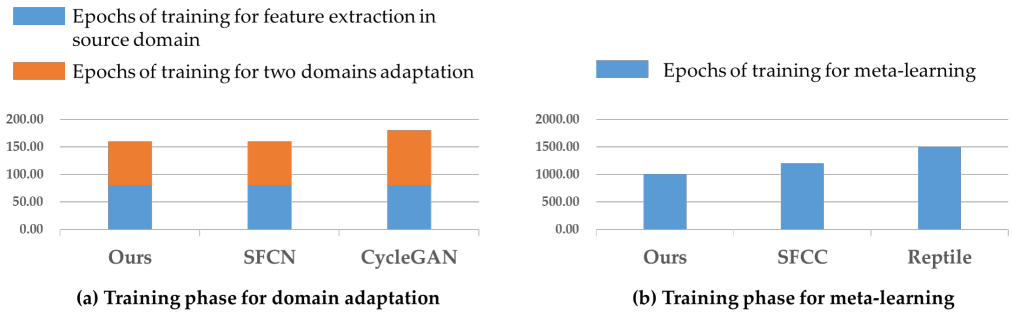


Figure 9. Comparison of Computational Cost Analysis with other methods.

If the whole work is compared with other crowd-counting algorithms, the time spent is much greater than others because of additional domain adaptation and meta-learning. However, we can train the domain-adaptation module as a pre-training model. The remaining training time of our algorithm is almost the same as that of other algorithms. In addition, in real-time crowd-counting estimation scenarios, as shown in Table 4, our algorithm can reach 1one∼two frames per second , which can satisfy the real-time density estimation.

Table 4. Real-time analysis in the inference phase.

Method	Frames per Second
CSRnet	8∼10
SFCC	3∼5
Ours	1∼2

5. Conclusions

Crowd counting is becoming increasingly popular in computer vision, as it is relevant to an extensive range of applications. In supervised learning, particularly, its performance

has dramatically improved. However, in many real-world scenarios, the different angles, exposures, location heights, and complex backgrounds of photos, along with limited annotation data, lead to supervised learning methods not working satisfactorily, and many suffer from overfitting problems. In this research, we focused on training synthetic crowd data and examined how to transfer knowledge to real-world datasets in two key phases: feature extraction and density estimation. The adaptive domain-invariant feature-extracting module aims to align the feature level with the source and target domains. In addition, the density-map-estimator module, based on dynamic-β MAML, trains the model in few-shot scenarios to improve generalization. Furthermore, we used a counting-map refiner to optimize the coarse density map into a fine density map and then regressed the crowd size. Finally, we compared our proposed method to the benchmark and achieved superior performance in cross-domain scenarios. The proposed method also has some limitations, such as more time to train domain-invariant features in the domain-adaptation phase and the need for more synthetic data to cover different scenarios. However, the advantage is that synthetic data is easier to generate and label annotations for in batches than is real-world data, which is equivalent to replacing manual annotation time with computational time.

Author Contributions: Conceptualization, X.H. and H.X.; methodology, X.H.; software, X.H. and J.X.; validation, J.W., J.X. and X.H.; formal analysis, X.H. and J.W.; investigation, X.H. and J.X.; resources, X.H.; data curation, X.H., J.X. and J.W.; writing original draft preparation, X.H. and J.X.; writing review and editing, H.X. and J.X.; visualization, X.H.; supervision, J.X.; project administration, X.H. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Data available on request due to restrictions e.g., privacy or ethical. The data presented in this study are available on request from the corresponding author. The data are not publicly available due to privacy.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Gao, G.; Gao, J.; Liu, Q.; Wang, Q.; Wang, Y. CNN-based Density Estimation and Crowd Counting: A Survey. *arXiv* **2020**, arXiv:2003.12783.
2. Cenggoro, T.W. Deep learning for crowd counting: A survey. *Eng. Math. Comput. Sci. J.* **2019**, *1*, 17–28. [CrossRef]
3. Shao, J.; Kang, K.; Change Loy, C.; Wang, X. Deeply learned attributes for crowded scene understanding. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Boston, MA, USA, 7–12 June 2015; pp. 4657–4666.
4. Gao, J.; Wang, Q.; Li, X. Pcc net: Perspective crowd counting via spatial convolutional network. *IEEE Trans. Circuits Syst. Video Technol.* **2019**, *30*, 3486–3498. [CrossRef]
5. Gao, J.; Han, T.; Wang, Q.; Yuan, Y. Domain-adaptive crowd counting via inter-domain features segregation and gaussian-prior reconstruction. *arXiv* **2019**, arXiv:1912.03677.
6. Zhang, Y.; Zhou, D.; Chen, S.; Gao, S.; Ma, Y. Single-image crowd counting via multi-column convolutional neural network. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, NV, USA, 27–30 June 2016; pp. 589–597.
7. Onoro-Rubio, D.; López-Sastre, R.J. Towards perspective-free object counting with deep learning. In Proceedings of the European Conference on Computer Vision, Amsterdam, The Netherlands, 8–16 October 2016; Springer: Berlin/Heidelberg, Germany, 2016; pp. 615–629.
8. Hossain, M.; Hosseinzadeh, M.; Chanda, O.; Wang, Y. Crowd counting using scale-aware attention networks. In Proceedings of the 2019 IEEE Winter Conference on Applications of Computer Vision (WACV), Waikoloa, HI, USA, 7–11 January 2019; IEEE: Piscataway, NJ, USA, 2019; pp. 1280–1288.
9. Han, T.; Gao, J.; Yuan, Y.; Wang, Q. Focus on semantic consistency for cross-domain crowd understanding. In Proceedings of the ICASSP 2020—2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), Barcelona, Spain, 4–8 May 2020; IEEE: Piscataway, NJ, USA, 2020; pp. 1848–1852.
10. Gao, J.; Wang, Q.; Yuan, Y. Feature-aware adaptation and structured density alignment for crowd counting in video surveillance. *arXiv* **2019**, arXiv:1912.03672.

11. Idrees, H.; Saleemi, I.; Seibert, C.; Shah, M. Multi-source multi-scale counting in extremely dense crowd images. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Portland, OR, USA, 23–28 June 2013; pp. 2547–2554.
12. Zhu, J.Y.; Park, T.; Isola, P.; Efros, A.A. Unpaired image-to-image translation using cycle-consistent adversarial networks. In Proceedings of the IEEE International Conference on Computer Vision, Venice, Italy, 27–29 October 2017; pp. 2223–2232.
13. Reddy, M.K.K.; Hossain, M.; Rochan, M.; Wang, Y. Few-shot scene adaptive crowd counting using meta-learning. In Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision, Snowmass Village, CO, USA, 1–5 March 2020; pp. 2814–2823.
14. Wortsman, M.; Ehsani, K.; Rastegari, M.; Farhadi, A.; Mottaghi, R. Learning to learn how to learn: Self-adaptive visual navigation using meta-learning. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Long Beach, CA, USA, 16–20 June 2019; pp. 6750–6759.
15. Gall, J.; Yao, A.; Razavi, N.; Van Gool, L.; Lempitsky, V. Hough forests for object detection, tracking, and action recognition. *IEEE Trans. Pattern Anal. Mach. Intell.* **2011**, *33*, 2188–2202. [CrossRef]
16. Wu, B.; Nevatia, R. Detection and tracking of multiple, partially occluded humans by bayesian combination of edgelet based part detectors. *Int. J. Comput. Vis.* **2007**, *75*, 247–266. [CrossRef]
17. Li, M.; Zhang, Z.; Huang, K.; Tan, T. Estimating the number of people in crowded scenes by mid based foreground segmentation and head-shoulder detection. In Proceedings of the 2008 19th International Conference on Pattern Recognition, Tampa, FL, USA, 8–11 December 2008; IEEE: Piscataway, NJ, USA, 2008; pp. 1–4.
18. Wan, J.; Luo, W.; Wu, B.; Chan, A.B.; Liu, W. Residual regression with semantic prior for crowd counting. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Long Beach, CA, USA , 16–20 June 2019; pp. 4036–4045.
19. Pham, V.Q.; Kozakaya, T.; Yamaguchi, O.; Okada, R. Count forest: Co-voting uncertain number of targets using random forest for crowd density estimation. In Proceedings of the IEEE International Conference on Computer Vision, Santiago, Chile, 7–13 December 2015; pp. 3253–3261.
20. Boominathan, L.; Kruthiventi, S.S.; Babu, R.V. Crowdnet: A deep convolutional network for dense crowd counting. In Proceedings of the 24th ACM International Conference on Multimedia, Amsterdam, The Netherlands, 15–19 October 2016; pp. 640–644.
21. Wang, Y.; Hu, S.; Wang, G.; Chen, C.; Pan, Z. Multi-scale dilated convolution of convolutional neural network for crowd counting. *Multimed. Tools Appl.* **2020**, *79*, 1057–1073. [CrossRef]
22. Ma, Z.; Hong, X.; Wei, X.; Qiu, Y.; Gong, Y. Towards a Universal Model for Cross-Dataset Crowd Counting. In Proceedings of the IEEE/CVF International Conference on Computer Vision, Montreal, QC, Canada, 11–17 October 2021; pp. 3205–3214.
23. Varol, G.; Romero, J.; Martin, X.; Mahmood, N.; Black, M.J.; Laptev, I.; Schmid, C. Learning from synthetic humans. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, HI, USA, 21–26 July 2017; pp. 109–117.
24. Nam, H.; Lee, H.; Park, J.; Yoon, W.; Yoo, D. Reducing domain gap via style-agnostic networks. *arXiv* **2019**, arXiv:1910.11645.
25. Pan, S.J.; Ni, X.; Sun, J.T.; Yang, Q.; Chen, Z. Cross-domain sentiment classification via spectral feature alignment. In Proceedings of the 19th International Conference on World Wide Web, Raleigh, NC, USA, 26–30 April 2010; pp. 751–760.
26. Pan, F.; Shin, I.; Rameau, F.; Lee, S.; Kweon, I.S. Unsupervised intra-domain adaptation for semantic segmentation through self-supervision. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Seattle, WA, USA, 13–19 June 2020; pp. 3764–3773.
27. Sohn, K.; Liu, S.; Zhong, G.; Yu, X.; Yang, M.H.; Chandraker, M. Unsupervised domain adaptation for face recognition in unlabeled videos. In Proceedings of the IEEE International Conference on Computer Vision, Venice, Italy, 27–29 October 2017; pp. 3210–3218.
28. Ganin, Y.; Lempitsky, V. Unsupervised domain adaptation by backpropagation. *arXiv* **2014**, arXiv:1409.7495.
29. Hoffman, J.; Wang, D.; Yu, F.; Darrell, T. Fcns in the wild: Pixel-level adversarial and constraint-based adaptation. *arXiv* **2016**, arXiv:1612.02649.
30. Sankaranarayanan, S.; Balaji, Y.; Jain, A.; Lim, S.N.; Chellappa, R. Learning from synthetic data: Addressing domain shift for semantic segmentation. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18–23 June 2018; pp. 3752–3761.
31. Hoffman, J.; Tzeng, E.; Park, T.; Zhu, J.Y.; Isola, P.; Saenko, K.; Efros, A.; Darrell, T. Cycada: Cycle-consistent adversarial domain adaptation. In Proceedings of the International Conference on Machine Learning, Stockholm, Sweden, 10–15 July 2018; pp. 1989–1998.
32. Tsai, Y.H.; Hung, W.C.; Schulter, S.; Sohn, K.; Yang, M.H.; Chandraker, M. Learning to adapt structured output space for semantic segmentation. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18–23 June 2018; pp. 7472–7481.
33. Wang, B.; Li, G.; Wu, C.; Zhang, W.; Zhou, J.; Wei, Y. A Framework for Self-Supervised Federated Domain Adaptation. *Eurasip J. Wirel. Commun. Netw.* **2021**. [CrossRef]
34. Wen, J.; Liu, R.; Zheng, N.; Zheng, Q.; Gong, Z.; Yuan, J. Exploiting local feature patterns for unsupervised domain adaptation. In Proceedings of the AAAI Conference on Artificial Intelligence, Honolulu, HI, USA, 27 January–1 February 2019; Volume 33, pp. 5401–5408.
35. Caruana, R. Multitask learning. *Mach. Learn.* **1997**, *28*, 41–75. [CrossRef]
36. Zhang, Y.; Yang, Q. A survey on multi-task learning. *arXiv* **2017**, arXiv:1707.08114.

37. Jia, Y.; Shelhamer, E.; Donahue, J.; Karayev, S.; Long, J.; Girshick, R.; Guadarrama, S.; Darrell, T. Caffe: Convolutional architecture for fast feature embedding. In Proceedings of the 22nd ACM International Conference on Multimedia, Orlando, FL, USA, 3–7 November 2014; pp. 675–678.
38. Oreshkin, B.N.; Rodriguez, P.; Lacoste, A. Tadam: Task dependent adaptive metric for improved few-shot learning. *arXiv* **2018**, arXiv:1805.10123.
39. Zhao, F.; Zhao, J.; Yan, S.; Feng, J. Dynamic conditional networks for few-shot learning. In Proceedings of the European Conference on Computer Vision (ECCV), Munich, Germany, 8–14 September 2018; pp. 19–35.
40. Edwards, H.; Storkey, A. Towards a neural statistician. *arXiv* **2016**, arXiv:1606.02185.
41. Rezende, D.; Danihelka, I.; Gregor, K.; Wierstra, D. One-shot generalization in deep generative models. In Proceedings of the International Conference on Machine Learning, New York, NY, USA, 20–22 June 2016; pp. 1521–1529.
42. Zhang, R.; Che, T.; Ghahramani, Z.; Bengio, Y.; Song, Y. MetaGAN: An Adversarial Approach to Few-Shot Learning. In Proceedings of the 32nd Conference on Neural Information Processing Systems (NIPS 2018), Montreal, QC, Canada, 2–8 December 2018; Volume 2, p. 1.
43. Zhang, Y.; Tang, H.; Jia, K. Fine-grained visual categorization using meta-learning optimization with sample selection of auxiliary data. In Proceedings of the European Conference on Computer Vision (ECCV), Munich, Germany, 8–14 September 2018; pp. 233–248.
44. Luo, Z.; Zou, Y.; Hoffman, J.; Fei-Fei, L. Label efficient learning of transferable representations across domains and tasks. *arXiv* **2017**, arXiv:1712.00123.
45. Fink, M. Object classification from a single example utilizing class relevance metrics. *Adv. Neural Inf. Process. Syst.* **2005**, *17*, 449–456.
46. Reed, S.; Chen, Y.; Paine, T.; Oord, A.v.d.; Eslami, S.; Rezende, D.; Vinyals, O.; de Freitas, N. Few-shot autoregressive density estimation: Towards learning to learn distributions. *arXiv* **2017**, arXiv:1710.10304.
47. Koch, G.; Zemel, R.; Salakhutdinov, R. Siamese neural networks for one-shot image recognition. In Proceedings of the ICML Deep Learning Workshop, Lille, France, 10–11 July 2015; Volume 2.
48. Vinyals, O.; Blundell, C.; Lillicrap, T.; Wierstra, D. Matching networks for one shot learning. *Adv. Neural Inf. Process. Syst.* **2016**, *29*, 3630–3638.
49. Ganin, Y.; Kulkarni, T.; Babuschkin, I.; Eslami, S.A.; Vinyals, O. Synthesizing programs for images using reinforced adversarial learning. In Proceedings of the International Conference on Machine Learning, Stockholm, Sweden, 10–15 July 2018; pp. 1666–1675.
50. Prakash, A.; Boochoon, S.; Brophy, M.; Acuna, D.; Cameracci, E.; State, G.; Shapira, O.; Birchfield, S. Structured domain randomization: Bridging the reality gap by context-aware synthetic data. In Proceedings of the 2019 International Conference on Robotics and Automation (ICRA), Montreal, QC, Canada, 20–24 May 2019; IEEE: Piscataway, NJ, USA, 2019, pp. 7249–7255.
51. Richter, S.R.; Vineet, V.; Roth, S.; Koltun, V. Playing for data: Ground truth from computer games. In Proceedings of the European Conference on Computer Vision, Amsterdam, The Netherlands, 8–16 October 2016; Springer: Berlin/Heidelberg, Germany, 2016; pp. 102–118.
52. Beery, S.; Liu, Y.; Morris, D.; Piavis, J.; Kapoor, A.; Joshi, N.; Meister, M.; Perona, P. Synthetic examples improve generalization for rare classes. In Proceedings of the Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision, Snowmass Village, CO, USA, 1–5 March 2020; pp. 863–873.
53. Wang, Q.; Gao, J.; Lin, W.; Yuan, Y. Learning from synthetic data for crowd counting in the wild. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Long Beach, CA, USA, 16–20 June 2019; pp. 8198–8207.
54. Krähenbühl, P. Free supervision from video games. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18–23 June 2018; pp. 2955–2964.
55. Ros, G.; Sellart, L.; Materzynska, J.; Vazquez, D.; Lopez, A.M. The synthia dataset: A large collection of synthetic images for semantic segmentation of urban scenes. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, NV, USA, 27–30 June 2016; pp. 3234–3243.
56. Zhang, C.; Kang, K.; Li, H.; Wang, X.; Xie, R.; Yang, X. Data-driven crowd understanding: A baseline for a large-scale crowd dataset. *IEEE Trans. Multimed.* **2016**, *18*, 1048–1061. [CrossRef]
57. Berga, D.; Fdez-Vidal, X.R.; Otazu, X.; Pardo, X.M. Sid4vam: A benchmark dataset with synthetic images for visual attention modeling. In Proceedings of the IEEE/CVF International Conference on Computer Vision, Seoul, Korea, 27–28 October 2019; pp. 8789–8798.
58. Zheng, Y.; Huang, D.; Liu, S.; Wang, Y. Cross-domain object detection through coarse-to-fine feature adaptation. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Seattle, WA, USA, 13–19 June 2020; pp. 13766–13775.
59. Loy, C.C.; Chen, K.; Gong, S.; Xiang, T. Crowd counting and profiling: Methodology and evaluation. In *Modeling, Simulation and Visual Analysis of Crowds*; Springer: Berlin/Heidelberg, Germany, 2013; pp. 347–382.
60. Li, Y.; Zhang, X.; Chen, D. CSRNet: Dilated Convolutional Neural Networks for Understanding the Highly Congested Scenes. *arXiv* **2018**, arXiv:1802.10062.
61. Liu, W.; Salzmann, M.; Fua, P. Context-Aware Crowd Counting. *arXiv* **2019**, arXiv:1811.10452.
62. Finn, C.; Abbeel, P.; Levine, S. Model-Agnostic Meta-Learning for Fast Adaptation of Deep Networks. *arXiv* **2017**, arXiv:1703.03400.

63. Behl, H.S.; Baydin, A.G.; Torr, P.H. Alpha maml: Adaptive model-agnostic meta-learning. *arXiv* **2019**, arXiv:1905.07435.
64. Jordan, M.I.; Mitchell, T.M. Machine learning: Trends, perspectives, and prospects. *Science* **2015**, *349*, 255–260. [CrossRef]
65. Vanschoren, J. Meta-learning: A survey. *arXiv* **2018**, arXiv:1810.03548.
66. Nixon, M.; Aguado, A. *Feature Extraction and Image Processing for Computer Vision*; Academic Press: Cambridge, MA, USA, 2019.
67. Ganin, Y.; Lempitsky, V. Unsupervised domain adaptation by backpropagation. In Proceedings of the International Conference on Machine Learning, Lille, France, 7–9 July 2015; pp. 1180–1189.
68. Wang, M.; Deng, W. Deep visual domain adaptation: A survey. *Neurocomputing* **2018**, *312*, 135–153. [CrossRef]
69. Hoffman, J.; Rodner, E.; Donahue, J.; Darrell, T.; Saenko, K. Efficient learning of domain-invariant image representations. *arXiv* **2013**, arXiv:1301.3224.
70. Inoue, N.; Furuta, R.; Yamasaki, T.; Aizawa, K. Cross-Domain Weakly-Supervised Object Detection Through Progressive Domain Adaptation. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Salt Lake City, UT, USA, 18–23 June 2018.