Learning from Synthetic Data for Crowd Counting in the Wild

Qi Wang, Junyu Gao, Wei Lin, Yuan Yuan
School of Computer Science and Center for OPTical IMagery Analysis and Learning (OPTIMAL),
Northwestern Polytechnical University, Xi'an, Shaanxi, P. R. China
{crabwq, gjy3035, elonlin24, y.yuan1.ieee}@gmail.com

Abstract

Recently, counting the number of people for crowd scenes is a hot topic because of its widespread applications (e.g. video surveillance, public security). It is a difficult task in the wild: changeable environment, large-range number of people cause the current methods can not work well. In addition, due to the scarce data, many methods suffer from over-fitting to a different extent. To remedy the above two problems, firstly, we develop a data collector and labeler, which can generate the synthetic crowd scenes and simultaneously annotate them without any manpower. Based on it, we build a large-scale, diverse synthetic dataset. Secondly, we propose two schemes that exploit the synthetic data to boost the performance of crowd counting in the wild: 1) pretrain a crowd counter on the synthetic data, then finetune it using the real data, which significantly prompts the model's performance on real data; 2) propose a crowd counting method via domain adaptation, which can free humans from heavy data annotations. Extensive experiments show that the first method achieves the state-of-the-art performance on four real datasets, and the second outperforms our baselines. The dataset and source code are available at https://gjy3035.github.io/GCC-CL/.

1. Introduction

Crowd counting is a branch of crowd analysis [17, 29, 18, 37], which is essential to video surveillance, public areas planning, traffic flow monitoring and so on. This task aims to predict density maps and estimate the number of people for crowd scenes. At present, many CNN- and GAN-based methods [43, 31, 32, 33, 7] attain a phenomenal performance on the existing datasets. The above methods focus on how to learn effective and discriminative features (such as local patterns, global contexts, multi-scale features and so on) to improve models' performance.

At the same time, The aforementioned mainstream deep learning methods need a large amount of accurately labeled and diversified data. Unfortunately, current datasets

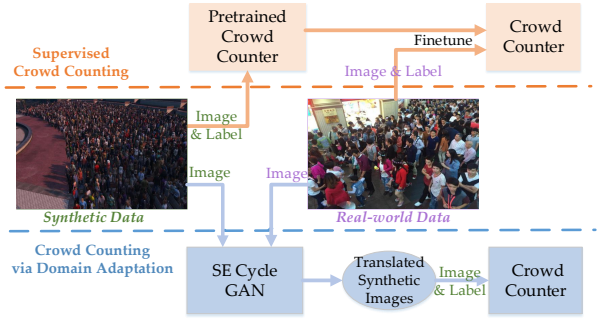


Figure 1. Two ways of using the proposed GCC dataset: supervised learning and domain adaptation.

[8, 9, 41, 43, 38, 14, 15] can not perfectly satisfy the needs, which also results in two intractable problems for crowd counting in the wild. Firstly, it causes that the existing methods cannot be performed to tackle some unseen extreme cases in the wild (such as changeable weather, variant illumination and a large-range number of people). Secondly, due to rare labeled data, many algorithms suffer from overfitting, which leads to a large performance degradation during transferring them to the wild or other scenes. In addition, there is an inherent problem in the congested crowd datasets: the labels are not very accurate, such as some samples in UCF_CC_50 [14] and Shanghai Tech A [43] ("SHT A" for short).

In order to remedy the aforementioned problems, we start from two aspects, namely data and methodology. From the data perspective, we develop a data collector and labeler, which can generate synthetic crowd scenes and automatically annotate them. By the collector and labeler, we construct a large-scale and diverse synthetic crowd counting dataset. The data is collected from an electronic game Grand Theft Auto V (GTA5), thus it is named as "GTA5 Crowd Counting" ("GCC" for short) dataset. Compared with the existing real datasets, there are four advantages: 1) free collection and annotation; 2) larger data volume and higher resolution; 3) more diversified scenes and 4) more accurate annotations. The detailed statistics are reported in Table 1.

From the methodological perspective, we propose two

ways to exploit synthetic data to improve the performance in the wild. Firstly, we propose a supervised strategy to reduce the overfitting phenomenon. To be specific, we firstly exploit the large-scale synthetic data to pretrain a crowd counter, which is our designed Spatial Fully Convolutional Network (SFCN). Then we finetune the obtained counter using the real data. This strategy can effectively prompt the performance on real data. Traditional models (training from scratch [43, 26, 7] or image classification model [5, 33, 15]) have some layers with random initialization or a regular distribution, which is not a good scheme. Compared with them, our strategy can provide more complete and better initialization parameters.

Secondly, we propose a domain adaptation crowd counting method, which can improve the cross-domain transfer ability. To be specific, we present an SSIM Embedding (SE) Cycle GAN, which can effectively translate the synthetic crowd scenes to real scenes. During the training process, we introduce the Structural Similarity Index (SSIM) loss. It is a penalty between the original image and reconstructed image through the two generators. Compared with the original Cycle GAN, the proposed SE effectively maintains local patterns and texture information, especially in the extremely congested crowd region and some backgrounds. Finally, we translate the synthetic data to photo-realistic images. Based on these data, we train a crowd counter without the labels of real data, which can work well in the wild. Fig. 1 demonstrates two flowcharts of the proposed methods.

In summary, this paper's contributions are three-fold:

1) We are the first to develop a data collector and labeler for crowd counting, which can automatically collect and annotate images without any labor costs. By using them, we create the first large-scale, synthetic and diverse crowd counting dataset.

2) We present a pretrained scheme to facilitate the original method's performance on the real data, which can more effectively reduce the estimation errors compared with random initialization and ImageNet model. Further, through the strategy, our proposed SFCN achieves the state-of-the-art results.

3) We are the first to propose a crowd counting method via domain adaptation, which does not use any label of the real data. By our designed SE Cycle GAN, the domain gap between the synthetic and real data can be significantly reduced. Finally, the proposed method outperforms the two baselines.

2. Related Works

Crowd Counting Methods. Mainstream CNN-based crowd counting methods [42, 43, 35, 36, 19, 22, 15, 7, 33, 26] yield the new record by designing the effective network architectures. [42, 35] exploit multi-task learning to explore the relation of different tasks to improve the counting performance. [43, 15, 7, 26] integrate the features of multi-stream, multi-scale or multi-stage networks to improve the quality of density maps. [36, 19] attempt to encode the large-range contextual information for crowd scenes. In order to tackle scarce data, [22] proposes a self-supervised learning to exploit unlabeled web data, and [33] presents a deep negative correlation learning to reduce the over-fitting.

Table 1. Statistics of the seven real-world datasets and the synthetic GCC dataset.

Dataset	Number of Images	Average Resolution	Count Statistics			
			Total	Min	Ave	Max
UCSD [8]	2,000	158×238	49,885	11	25	46
Mall [9]	2,000	480×640	62,325	13	31	53
UCF_CC_50 [14]	50	2101×2888	63,974	94	1,279	4,543
WorldExpo'10[41]	3,980	576×720	199,923	1	50	253
SHT A [43]	482	589×868	241,677	33	501	3,139
SHT B [43]	716	768×1024	88,488	9	123	578
UCF-QNRF [15]	1,525	2013×2902	1,251,642	49	815	12,865
GCC	**15,212**	$\mathbf{1080 \times 1920}$	**7,625,843**	**0**	**501**	**3,995**

Crowd Counting Datasets. In addition to the algorithms, the datasets potentially promote the development of crowd counting. UCSD [8] is the first crowd counting dataset released by Chan *et al.* from University of California San Diego. It records the crowd in a pedestrian walkway, which is a sparse crowd scene. Chen *et al.* [9] propose a public Mall dataset which records a shopping mall scene. Idrees *et al.* [14] release the UCF_CC_50 dataset for highly congested crowd scenes. WorldExpo'10 dataset is proposed by Zhang *et al.* in [41], which is captured from surveillance cameras in Shanghai 2010 WorldExpo. Zhang *et al.* [43] present ShanghaiTech Dataset, including the high-quality real-world images. Idrees *et al.* [15] propose a large-scale extremely congested dataset. More detailed information about them is listed in Table 1.

Synthetic Dataset. Annotating the groundtruth is a time-consuming and labor-intensive work, especially for pixel-wise tasks (such as semantic segmentation, density map estimation). To remedy this problem, some synthetic datasets [28, 16, 27, 30, 6] are released to save the manpower. [28, 16, 27] collect synthetic scenes based on GTA5. To be specific, [28] develops a fast annotation method based on the rendering pipeline. Johnson-Roberson *et al.* [16] present a method to analyze the internal engine buffers according the depth information, which can produce the accurate object masks. [27] proposes an approach to extract data without modifying the source code and content from GTA5, which can provide six types groundtruth. [30, 6] build synthetic models based on some open-source game engine. [30] exploits Unity Engine [3] to construct the synthetic street scenes data for autonomous driving, which generates the pixel-wise segmentation labels and depth maps. [6] develops a synthetic person re-identification dataset based on Unreal Engine 4 [4].

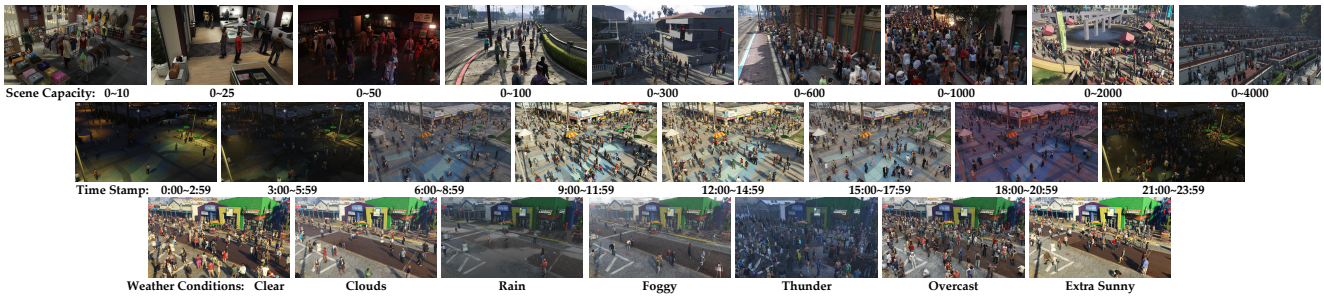


Figure 2. The display of the proposed GCC dataset from three views: scene capacity, timestamp and weather conditions.

Figure 3. The demonstration of image combination for congested crowd scenes.

3. GTA5 Crowd Counting (GCC) Dataset

Grand Theft Auto V (GTA5) is a computer game published by Rockstar Games [1] in 2013. In GTA5, the players can immerse themselves into the game in a virtual world, the fictional city of Los Santos, based on Los Angeles. GTA5 adopts the proprietary Rockstar Advanced Game Engine (RAGE) to improve its draw distance rendering capabilities. Benefiting from the excellent game engine, its scene rendering, texture details, weather effects and so on are very close to the real-world conditions. In addition, Rockstar Games allows the players to develop the mod for noncommercial or personal use.

Considering the aforementioned advantages, we develop a data collector and labeler for crowd counting in GTA5, which is based on Script Hook V [2]. Script Hook V is a C++ library for developing game plugins. Our data collector constructs the complex and congested crowd scenes via exploiting the objects of virtual world. Then, the collector captures the stable images from the constructed scenes. Finally, by analyzing the data from rendering stencil, the labeler automatically annotates the accurate head locations of persons.

Previous synthetic GTA5 datasets [28, 16, 27] capture normal scenes directed by the game programming. Unfortunately, there is no congested scene in GTA5. Thus, we need to design a strategy to construct crowd scenes, which is the most obvious difference with them.

3.1. Data Collection

This section describes the pipeline of data collection, which consists of three modules as follows.

Scene Selection. The virtual world in GTA5 is built on a fictional city, which covers an area of 252 square kilometers. In the city, we selected 100 typical locations, such as beach, stadium, mall, store and so on. For each location, the four surveillance cameras are equipped with different parameters (location, height, rotation/pitch angle). Finally, the 400 diverse scenes are built. In these scenes, we elaborately define the Region of Interest (ROI) for placing the persons and exclude some invalid regions according to common sense.

Person Model. Persons are the core of crowd scenes. Thus, it is necessary that we describe the person model in our proposed dataset. In GCC dataset, we adopt the 265 person models in GTA5: different person model has different skin color, gender, shape and so on. Besides, for each person model, it has six variations on external appearance, such as clothing, haircut, etc. In order to improve the diversity of person models, each model is ordered to do a random action in the sparse crowd scenes.

Scenes Synthesis for Congested Crowd. Due to the limitation of GTA5, the number of people must be less than 256. Considering this, for the congested crowd scenes, we adopt a step-by-step method to generate scenes. To be specific, we segment several non-overlapping regions and then place persons in each region. Next, we integrate multiple scenes into one scene. Fig. 3 describes the main integration process: the persons are placed in the red and green regions in turn. Finally, the two images are combined in the one.

Summary. The flowchart of generation is described as follows. *Construct scenes*: a) select a location and setup the cameras, b) segment Region of interest (ROI) for crowd, c) set weather and time. *Place persons*: a) create persons in the ROI and get the head positions, b) obtain the person mask from stencil, c) integrate multiple images into one image, d) remove the positions of occluded heads. The demonstration video is available at: https://www.youtube.com/watch?v=Hvl7xWkIueo.

3.2. Properties of GCC

GCC dataset consists of 15,212 images, with resolution of 1080×1920, containing 7,625,843 persons. Compared with the existing datasets, GCC is a more large-scale crowd counting dataset in both the number of images and the number of persons. Table 1 compares the basic information of GCC and the existing datasets. In addition to the above advantages, GCC is more diverse than other real-

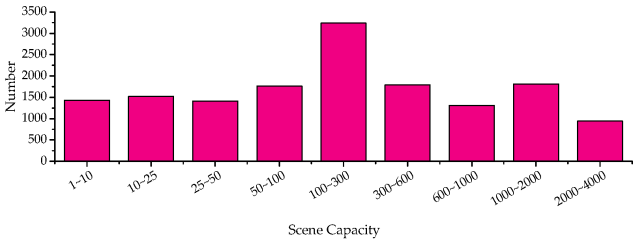
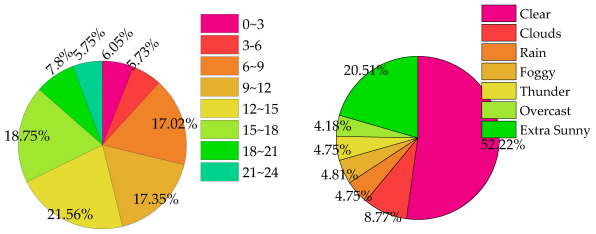


Figure 4. The statistical histogram of crowd counts on the proposed GCC dataset.



(a) Time stamp distribution. (b) Weather condition distribution.

Figure 5. The pie charts of time stamp and weather condition distribution on GCC dataset. In the left pie chart, the label "$0 \sim 3$" denotes the time period during $[0:00,\ 3:00)$ in 24 hours a day.

world datasets.

Diverse Scenes. GCC dataset consists of 400 different scenes, which includes multiple types of locations. For example, indoor scenes: convenience store, pub, etc. outdoor scenes: mall, street, plaza, stadium and so on. Further, all scenes are assigned with a level label according to their space capacity. The first row in Fig. 2 shows the typical scenes with different levels. In general, for covering the range of people, the larger scene has more images. Thus, the setting is conducted as follows: the scenes with the first/second/last three levels contain $30/40/50$ images. Besides, the images that contain some improper events should be deleted. Finally, the number of images in some scenes may be less than their expected value. Fig. 4 demonstrates the population distribution histogram of our GCC dataset.

Existing datasets only focus on one of sparse or congested crowd. However, a large scene may also contain very few people in the wild. Considering that, during the generation process of an image, the number of people is set as random value in the range of its level. Therefore, GCC has more large-range than other real datasets.

Diverse Environments. In order to construct the data that are close to the wild, the images are captured at a random time in a day and under a random weather conditions. In GTA5, we select seven types of weathers: clear, clouds, rain, foggy, thunder, overcast and extra sunny. The last two rows of Fig. 2 illustrate the exemplars at different times and under various weathers. In the process of generation, we tend to produce more images under common conditions. The two sector charts in Fig. 5 respectively show the proportional distribution on the time stamp and weather conditions of GCC dataset.

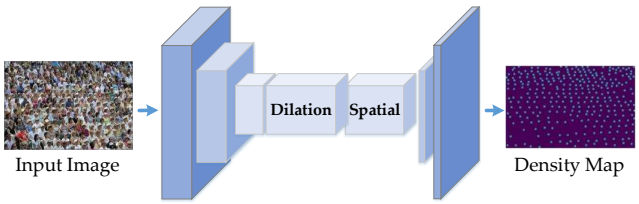


Figure 6. The architecture of spatial FCN (SFCN).

4. Supervised Crowd Counting

FCN-based methods [43, 24, 40, 19] attain good performances for crowd counting. In this section, we design an effective spatial Fully Convolutional Network (SFCN) to directly regress the density map, which is able to encode the global context information.

4.1. Network Architecture

Fully convolutional network (FCN) is proposed by Long *et al.* [23] in 2016, which focuses on pixel-wise task (such as semantic segmentation, saliency detection). FCN uses the convolutional layer to replace the fully connected layer in traditional CNN, which guarantees that the network can receive the image with an arbitrary size and produce the output of the corresponding size. For encoding the context information, Pan *et al.* [25] present a spatial encoder via a sequence of convolution on the four directions (down, up, left-to-right and right-to-left).

In this paper, we design a spatial FCN (SFCN) to produce the density map, which adopt VGG-16 [34] or ResnNet-101 [12] as the backbone. To be specific, the spatial encoder is added to the top of the backbone. The feature map flow is illustrated as in Fig. 6. After the spatial encoder, a regression layer is added, which directly outputs the density map with input's $1/8$ size. Here, we do not review the spatial encoder because of the limited space. During the training phase, the objective is minimizing standard Mean Squared Error at the pixel-wise level; the learning rate is set as 10^{-5}; and Adam algorithm is used to optimize SFCN.

4.2. Experiments

In this section, the two types of experiments are conducted: 1) training and testing within GCC dataset; 2) pretraining on GCC and fine-tuning on the real datasets.

4.2.1 Experiments on GCC Dataset

We report the results of the extensive experiments within GCC dataset, which verifies SFCN from three different training strategies: random, cross-camera and cross-location splitting. To be specific, the three strategies are explained as follows. 1) **Random splitting**: the entire dataset

Table 2. The results of our proposed SFCN and the three classic methods on GCC dataset.

Method	Random splitting				Cross-camera splitting				Cross-location splitting			
	MAE	MSE	PSNR	SSIM	MAE	MSE	PSNR	SSIM	MAE	MSE	PSNR	SSIM
MCNN [43]	100.9	217.6	24.00	0.838	110.0	221.5	23.81	0.842	154.8	340.7	24.05	0.857
CSR [19]	38.2	87.6	29.52	0.829	61.1	134.9	29.03	0.826	92.2	220.1	28.75	0.842
FCN	42.3	98.7	30.10	0.889	61.5	156.6	28.92	0.874	97.5	226.8	29.33	0.866
SFCN	**36.2**	**81.1**	**30.21**	**0.904**	**56.0**	**129.7**	**29.17**	**0.889**	**89.3**	**216.8**	**29.50**	**0.906**

is randomly divided into two groups as the training set (75%) and testing set (25%), respectively. 2) **Cross-camera splitting**: as for a specific location, one surveillance camera is randomly selected for testing and the others for training. 3) **Cross-location splitting**: we randomly choose 75/25 locations for training/testing. These scheme can effectively evaluated the algorithm on GCC. Table 2 reports the performance of our SFCN and two popular methods (MCNN [43] and CSRNet[19]) on the proposed GCC dataset.

4.2.2 Experiments of Pretraining & Finetuning

Many current methods suffer from the over-fitting because of scarce real labeled data. Some methods ([5, 33, 15]) exploit the pre-trained model based on ImageNet Database [10]. However, the trained classification models (VGG [34], ResNet [12] and DenseNet [13]) are not a best initialization for the regression problem: the regression layers and the specific modules are still initialized at the random or regular distributions.

In this paper, we propose a new scheme to remedy the above problems: firstly, the designed model is pretrained on the large-scale GCC Dataset; then the model pre-trained on GCC is finetuned using the real dataset. In the last step, the overall parameters are trained, which is better than traditional methods. To verify our strategy, we conduct the MCNN, CSR and SFCN on the two datasets (UCF-QNRF and SHT B). Note that SFCN adopts VGG-16 as backbone, and SFCN† uses the ResNet101 backbone. Table 3 shows the results of the comparison experiments. From it, we find that using the pretrained GCC models is better than not using or using ImageNet classification models. To be specific, for MCNN from scratch, our strategy can reduce by around 30% estimation errors. For the SFCN using pretrained ImageNet classification model, our scheme also decrease by an average 12% errors in four groups of experiments.

We also present the final results of our SFCN† on five real datasets, which is fintuned on the pretrained SFCN† using GCC. Compared with the state-of-the-art performance, SFCN† refreshes the records on the four datasets. The detailed results comparison is listed in the Table 4.

Table 3. The effect of pretrained GCC model on finetuning real dataset (MAE/MSE). "*" denotes other researchers' results.

Method	PreTr	UCF-QNRF	SHHT B
MCNN*	None	277/426 [15]	26.4/41.3 [43]
MCNN	None	281.2/445.0	26.3/39.5
MCNN	GCC	199.8/311.2(↓ 29/30%)	18.8/28.2(↓ 29/29%)
CSR*	ImgNt	-	10.6/16.0 [19]
CSR	ImgNt	120.3/208.5	10.6/16.6
CSR	GCC	112.4/185.6(↓ 7/11%)	10.1/15.7(↓ 5/5%)
SFCN	ImgNt	134.3/240.3	11.0/17.1
SFCN	GCC	124.7/203.5(↓ 7/15%)	9.4/14.4(↓ 15/16%)
SFCN†	ImgNt	114.8/192.0	8.9/14.3
SFCN†	GCC	**102.0/171.4**(↓ 11/11%)	**7.6/13.0**(↓ 15/9%)

Table 4. The comparison with the state-of-the-art performance on real datasets.

Dataset	Results (MAE/MSE)	
	SOTA	SFCN†
UCF-QNRF [15]	CL[15]: 132/191	**102.0/171.4**
SHT A [43]	SA[7]: 67.0/**104.5**	**64.8**/107.5
SHT B [43]	SA[7]: 8.4/13.6	**7.6/13.0**
UCF_CC_50 [14]	SAN[21]:219.2/**250.2**	**214.2**/318.2
WorldExpo'10[41]	ACSCP[32]:**7.5**(MAE)	9.4(MAE)

5. Crowd Counting via Domain Adaptation

The last section proposes the supervised learning on synthetic or real datasets, which adopts the labels of real data. For extremely congested scenes, manually annotating them is a tedious work. Not only that, there are label errors in man-made annotations. Therefore, we attempt to propose a crowd counting method via domain adaptation to save manpower, which learns specific patterns or features from the synthetic data and transfers them to the real world. Through this thought, we do not need any manual labels of real data. Unfortunately, the generated synthetic data are very different from real-world data (such as in color style, texture and so on), which is treated as "domain gap". Even in real life, the domain gap is also very common. For example, Shanghai Tech Part B and WorldExpo'10 are captured in different locations from different cameras, which causes that the data of them are quite different. Thus, it is an important task that how to transfer effective features between different domains, which is named as a "Domain Adaptation" (DA) problem.

In this work, we propose a crowd counting method via a domain adaptation, which can effectively learn domain-invariant feature between synthetic and real data. To be specific, we present a SSIM Embedding (SE) Cycle GAN to transform the synthetic image to the photo-realistic image. Then we will train a SFCN on the translated data. Finally, we directly test the model on the real data. The entire process does not need any manually labeled data. Fig. 7

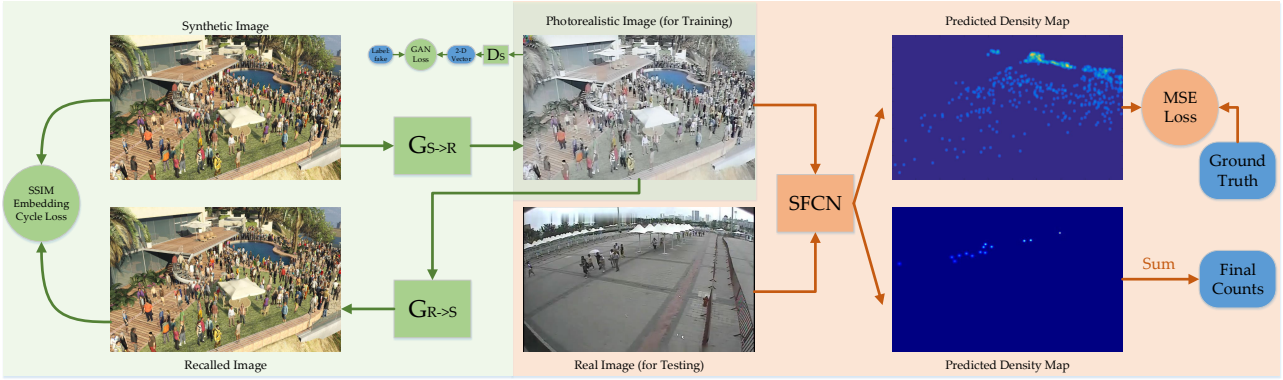


Figure 7. The flowchart of the proposed crowd counting via domain adaptation. The light green region is SSIM Embedding (SE) Cycle GAN, and light orange region represents Spatial FCN (FCN). Limited by paper length, we do not show the adaptation flowchart of real images to synthetic images (R→S), which is similar to S→R.

demonstrates the flowchart of the proposed method.

5.1. SSIM Embedding Cycle GAN

Here, we recall the crowd counting via domain adaptation by mathematical notations. The purpose of DA is to learn translation mapping between the synthetic domain $\mathcal{S}$ and the real-world domain $\mathcal{R}$. The synthetic domain $\mathcal{S}$ provides images $I_{\mathcal{S}}$ and count labels $L_{\mathcal{S}}$. And the real-world domain $\mathcal{R}$ only provides images $I_{\mathcal{R}}$. In a word, given $\mathrm{i}_{\mathcal{S}} \in I_{\mathcal{S}}$, $\mathrm{l}_{\mathcal{S}} \in L_{\mathcal{S}}$ and $\mathrm{i}_{\mathcal{R}} \in I_{\mathcal{R}}$ (the lowercase letters represent the samples in the corresponding sets), we want to train a crowd counter to predict density maps of $\mathcal{R}$.

Cycle GAN. The original Cycle GAN [44] is proposed by Zhu *et al.*, which focuses on unpaired image-to-image translation. For different two domains, we can exploit Cycle GAN to handle DA problem, which can translate the synthetic images to photo-realistic images. As for the domain $\mathcal{S}$ and $\mathcal{R}$, we define two generator $G_{\mathcal{S}\to\mathcal{R}}$ and $G_{\mathcal{R}\to\mathcal{S}}$. The former one attempts to learn a mapping function from domain $\mathcal{S}$ to $\mathcal{R}$, and vice versa, the latter one's goal is to learn the mapping from domain $\mathcal{R}$ to $\mathcal{S}$. Following [44], we introduce the cycle-consistent loss to regularize the training process. To be specific, for the sample $\mathrm{i}_{\mathcal{S}}$ and $\mathrm{i}_{\mathcal{R}}$, one of our objective is $\mathrm{i}_{\mathcal{S}} \to G_{\mathcal{S}\to\mathcal{R}}(\mathrm{i}_{\mathcal{S}}) \to G_{\mathcal{R}\to\mathcal{S}}(G_{\mathcal{S}\to\mathcal{R}}(\mathrm{i}_{\mathcal{S}})) \approx \mathrm{i}_{\mathcal{S}}$. Another objective is inverse process for $\mathrm{i}_{\mathcal{R}}$. The cycle-consistent loss is an L1 penalty in the cycle architecture, which is defined as follows:

$$\begin{aligned}&\mathcal{L}_{cycle}(G_{\mathcal{S}\to\mathcal{R}}, G_{\mathcal{R}\to\mathcal{S}}, \mathcal{S}, \mathcal{R}) \\ &\quad = \mathbb{E}_{i_{\mathcal{S}}\sim I_{\mathcal{S}}}[\|G_{\mathcal{R}\to\mathcal{S}}(G_{\mathcal{S}\to R}(i_{\mathcal{S}})) - i_{\mathcal{S}}\|_1] \\ &\quad + \mathbb{E}_{i_{\mathcal{R}}\sim I_{\mathcal{R}}}[\|G_{\mathcal{S}\to\mathcal{R}}(G_{\mathcal{R}\to S}(i_{\mathcal{R}})) - i_{\mathcal{R}}\|_1].\end{aligned} \tag{1}$$

Additionally, two discriminators $D_{\mathcal{R}}$ and $D_{\mathcal{S}}$ are modeled corresponding to the $G_{\mathcal{S}\to\mathcal{R}}$ and $G_{\mathcal{R}\to\mathcal{S}}$. Specifically, $D_{\mathcal{R}}$ attempts to discriminate that where the images are from ($I_{\mathcal{R}}$ or $G_{\mathcal{S}\to\mathcal{R}}(I_{\mathcal{S}})$), and $D_{\mathcal{S}}$ tries to discriminate the images from $I_{\mathcal{S}}$ or $G_{\mathcal{R}\to\mathcal{S}}(I_{\mathcal{R}})$. Take $D_{\mathcal{R}}$ for example, and the training objective is adversarial loss [11], which is formulated as:

$$\begin{aligned}&\mathcal{L}_{GAN}(G_{\mathcal{S}\to\mathcal{R}}, D_{\mathcal{R}}, \mathcal{S}, \mathcal{R}) \\ &\quad = \mathbb{E}_{i_{\mathcal{R}}\sim I_{\mathcal{R}}}[log(D_{\mathcal{R}}(i_{\mathcal{R}})] \\ &\quad + \mathbb{E}_{i_{\mathcal{S}}\sim I_{\mathcal{S}}}[log(1 - D_{\mathcal{R}}(G_{\mathcal{S}\to\mathcal{R}}(i_{\mathcal{S}}))].\end{aligned} \tag{2}$$

The final loss function is defined as:

$$\begin{aligned}&\mathcal{L}_{CycleGAN}(G_{\mathcal{S}\to\mathcal{R}}, G_{\mathcal{R}\to\mathcal{S}}, D_{\mathcal{R}}, D_{\mathcal{S}}, \mathcal{S}, \mathcal{R}) \\ &\quad = \mathcal{L}_{GAN}(G_{\mathcal{S}\to\mathcal{R}}, D_{\mathcal{R}}, \mathcal{S}, \mathcal{R}) \\ &\quad + \mathcal{L}_{GAN}(G_{\mathcal{R}\to\mathcal{S}}, D_{\mathcal{S}}, \mathcal{S}, \mathcal{R}) \\ &\quad + \lambda\mathcal{L}_{cycle}(G_{\mathcal{S}\to\mathcal{R}}, G_{\mathcal{R}\to\mathcal{S}}, \mathcal{S}, \mathcal{R}),\end{aligned} \tag{3}$$

where λ is the weight of cycle-consistent loss.

SSIM Embedding Cycle-consistent loss. In the crowd scenes, the biggest differences between high-density regions and other regions (low-density regions or background) is the local patterns and texture features. Unfortunately, in the translation from synthetic to real images, the original cycle consistency is prone to losing them, which causes that the translated images lose the detailed information and are easily distorted.

To remedy the aforementioned problem, we introduce Structural Similarity Index (SSIM) [39] into the traditional CycleGAN, which is named as "SE Cycle GAN". SSIM is an indicator widely used in the field of image quality assessment, which computes the similarity between two images in terms of local patterns (mean, variance and covariance). About the SSIM in crowd counting, CP-CNN [36] is the first to evaluate the density map using SSIM, and SANet [7] adopt SSIM loss to generate high-quality density maps.

Similar to the traditional cycle consistency, our goal is: $G_{\mathcal{R}\to\mathcal{S}}(G_{\mathcal{S}\to\mathcal{R}}(\mathrm{i}_{\mathcal{S}})) \approx \mathrm{i}_{\mathcal{S}}$. To be specific, in addition to L1 penalty, the SSIM penalty is added to the training process. The range of SSIM value is in $[-1, 1]$, and larger SSIM means that the image has more higher quality. In particular, when the two images are identical, the SSIM value is

Table 5. The performance of no adaptation (No Adpt), Cycle GAN and SE Cycle GAN (ours) on the five real-world datasets.

Method	DA	SHT A				SHT B				UCF_CC_50			
		MAE	MSE	PSNR	SSIM	MAE	MSE	PSNR	SSIM	MAE	MSE	PSNR	SSIM
NoAdpt	✗	160.0	216.5	19.01	0.359	22.8	30.6	24.66	0.715	487.2	689.0	17.27	0.386
Cycle GAN[44]	✔	143.3	204.3	**19.27**	0.379	25.4	39.7	24.60	0.763	404.6	548.2	**17.34**	0.468
SE Cycle GAN (ours)	✔	**123.4**	**193.4**	18.61	**0.407**	**19.9**	**28.3**	**24.78**	**0.765**	**373.4**	**528.8**	17.01	**0.743**

Method	DA	UCF-QNRF				WorldExpo'10 (MAE)					
		MAE	MSE	PSNR	SSIM	S1	S2	S3	S4	S5	Avg.
NoAdpt	✗	275.5	458.5	20.12	0.554	4.4	87.2	59.1	51.8	11.7	42.8
Cycle GAN[44]	✔	257.3	400.6	20.80	0.480	4.4	69.6	49.9	29.2	9.0	32.4
SE Cycle GAN (ours)	✔	**230.4**	**384.5**	**21.03**	**0.660**	**4.3**	**59.1**	**43.7**	**17.0**	**7.6**	**26.3**

equal to 1. In the practice, we convert the SSIM value into the trainable form, which is defined as:

$$\begin{aligned}&\mathcal{L}_{SEcycle}(G_{\mathcal{S}\to\mathcal{R}}, G_{\mathcal{R}\to\mathcal{S}}, \mathcal{S}, \mathcal{R})\\&= \mathbb{E}_{i_\mathcal{S}\sim I_\mathcal{S}}[1 - SSIM(i_\mathcal{S}, G_{\mathcal{R}\to\mathcal{S}}(G_{\mathcal{S}\to R}(i_\mathcal{S})))]\\&+ \mathbb{E}_{i_\mathcal{R}\sim I_\mathcal{R}}[1 - SSIM(i_\mathcal{R}, G_{\mathcal{S}\to\mathcal{R}}(G_{\mathcal{R}\to S}(i_\mathcal{R})))],\end{aligned} \tag{4}$$

where $SSIM(\cdot,\cdot)$ is standard computation: the parameter settings are directly followed by [39]. The first input is the original image from domain $\mathcal{S}$ or $\mathcal{R}$, and the second input is the reconstructed image produced by the two generators in turns. Finally, the final objective of SE Cycle GAN is defined as:

$$\begin{aligned}&\mathcal{L}_{ours}(G_{\mathcal{S}\to\mathcal{R}}, G_{\mathcal{R}\to\mathcal{S}}, D_\mathcal{R}, D_\mathcal{S}, \mathcal{S}, \mathcal{R})\\&\quad= \mathcal{L}_{GAN}(G_{\mathcal{S}\to\mathcal{R}}, D_\mathcal{R}, \mathcal{S}, \mathcal{R})\\&\quad+ \mathcal{L}_{GAN}(G_{\mathcal{R}\to\mathcal{S}}, D_\mathcal{S}, \mathcal{S}, \mathcal{R})\\&\quad+ \lambda\mathcal{L}_{cycle}(G_{\mathcal{S}\to\mathcal{R}}, G_{\mathcal{R}\to\mathcal{S}}, \mathcal{S}, \mathcal{R})\\&\quad+ \mu\mathcal{L}_{SEcycle}(G_{\mathcal{S}\to\mathcal{R}}, G_{\mathcal{R}\to\mathcal{S}}, \mathcal{S}, \mathcal{R}),\end{aligned} \tag{5}$$

where λ and μ are the weights of cycle-consistent and SSIM Embedding cycle-consistent loss, respectively. During the training phase, the μ is set as 1, other parameters and settings are the same as Cycle GAN [44].

Density/Scene Regularization. For a better domain adaptation from synthetic to real world, we design two strategies to facilitate the DA model to learn domain-invariant feature and produce the valid density map.

Although we translate synthetic images to photo-realistic images, some objects and data distributions in the real world are unseen during training the translated images. As a pixel-wise regression problem, the density may be an arbitrary value in theory. In fact, in some preliminary experiments, we find some backgrounds in real data are estimated as some exceptionally large values. To handle this problem, we set a upper bound $MAX_\mathcal{S}$, which is defined as the max density in the synthetic data. If the output value of a pixel is more than $MAX_\mathcal{S}$, the output will be set as 0. Note that the network's last layer is ReLU, so the output of each pixel must be greater than or equal to 0.

Since GCC is large-counter-range and diverse dataset, using all images may cause the side effect in domain adaptation. For example, ShanghaiTech does not contain the thunder/rain scenes, and WorldExpo'10 does not have the scene that can accommodate more than 500 people. Training all translated synthetic images can decrease the adaptation performance on the specific dataset. Thus, we manually select some specific scenes for different datasets. The concrete strategies are described in the supplementary. In general, it is a coarse data filter not an elaborate selection.

5.2. Experiments

5.2.1 Performance on Real-world Datasets

In this section, we conduct the adaptation experiments from GCC dataset to five mainstream real-world datasets: ShanghaiTech A/B [43], UCF_CC_50 [14], UCF-QNRF [15] and WorldExpo'10[41]. For the best performance, all models adopt the Scene/Density Regularization mentioned in Section 5.1.

Table 5 shows the results of the No Adaptation (No Adpt), Cycle GAN and the proposed SSIM Embedding (SE) Cycle GAN. From it, we find the results after adaptation are far better than that of no adaptation, which indicates the adaptation can effectively reduce the domain gaps between synthetic and real-world data. After embedding SSIM loss in cycle GAN, almost all performances are improved on five datasets. There are only two reductions of PSNR on Shanghai Tech A and UCF_CC_50. In general, the proposed SE Cycle GAN outperforms the original Cycle GAN. In addition, we find the results on Shanghai Tech B achieve a good level, even outperforms some early supervised methods [43, 35, 31, 36, 20]. The main reasons are: 1) the real data is strongly consistent, which is captured by the same sensors; 2) the data has high image clarity. The two characteristics guarantee that the SE CycleGAN's adaptation on Shanghai Tech B is more effective than others.

Fig. 8 demonstrates three groups of visualized results on Shanghai Tech dataset. Compared with no adaptation, the map quality via Cycle GAN has a significant improvement. From Row 1, we find the predicted maps are very close to the groundtruth. However, for the extremely congested scenes (in Row 2 and 3), the results are far from the ground truth. We think the main reason is that the translated images lose the details (such as texture, sharpness and edge)

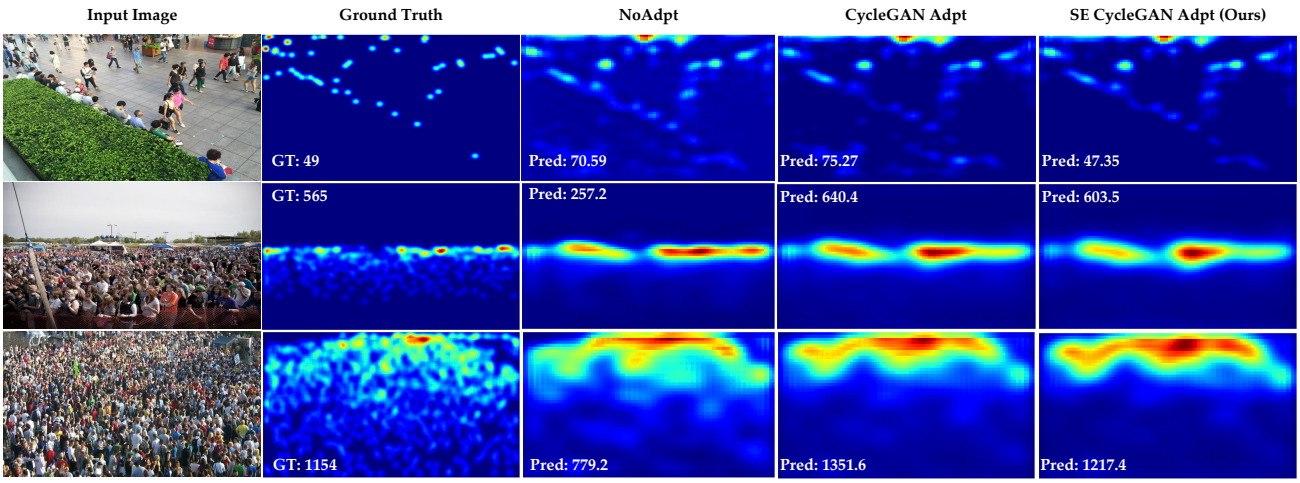


Figure 8. The demonstration of different methods on SHT dataset. "GT" and "Pred" represent the labeled and predicted count, respectively.

Figure 9. The comparison of Cycle GAN and SE Cycle GAN.

in high-density regions.

5.2.2 Analysis of SE & DSR

SSIM Embedding. SSIM Embedding can guarantee the original synthetic and reconstructed images have high structural similarity(SS), which prompts two generators' translation for images maintain a certain degree of SS during the training process. Fig. 9 illustrates the visualizations of two adaptations, where the first row is original images, the second and third row are translated images of Cycle GAN and SE Cycle GAN. Through comparison, the latter is able to retain local texture and structural similarity.

Density/Scene Regularization. Here, we compare the performance of three model (No Adpt, Cycle GAN and SE Cycle GAN) without Density/Scene Regularization (DSR) and with DSR. Table 6 reports the performance of with or without DSR on SHT A dataset. From the results in first column, we find these two adaptation methods cause some side effects. In fact, they do not produce the ideal translated images. When introducing DSR, the nonexistent synthetic scenes in the real datasets are filtered out, which improves the domain adaptation performance.

Table 6. The results under different configurations on SHT A.

Method	w/o DSR	with DSR
NoAdpt	**163.6**/244.5	160.0/216.5
Cycle GAN [44]	180.1/290.3	143.3/204.3
SE Cycle GAN	169.8/**230.2**	**123.4**/**193.4**

6. Conclusion

In this paper, we are committed to improving the performance of crowd counting in the wild. To this end, we firstly develop an automatic data collector/labeler and construct a large-scale synthetic crowd counting dataset. Exploiting the generated data, we then propose two effective ways (supervised learning and domain adaptation) to significantly improve the counting performance in the wild. The experiments demonstrate that the supervised method achieves the state-of-the-art performance and the domain adaptation method obtains acceptable results. In the future work, we will focus on the crowd counting via domain adaptation, and further explore that how to extract more effective domain-invariant features between synthetic and real-world data.

Acknowledgments. This work was supported by the National Natural Science Foundation of China under Grant U1864204 and 61773316, State Key Program of National Natural Science Foundation of China under Grant 61632018, Natural Science Foundation of Shaanxi Province under Grant 2018KJXX-024, and Project of Special Zone for National Defense Science and Technology Innovation.

References

[1] Rockstar games. https://www.rockstargames.com/. 3

[2] Script hook v. http://www.dev-c.com/gtav/scripthookv/. 3

[3] Unity engine. https://unity3d.com/. 2

[4] Unreal engine. https://www.unrealengine.com/. 2

[5] D. Babu Sam, N. N. Sajjan, R. Venkatesh Babu, and M. Srinivasan. Divide and grow: Capturing huge diversity in crowd images with incrementally growing cnn. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3618–3626, 2018. 2, 5

[6] S. Bak, P. Carr, and J.-F. Lalonde. Domain adaptation through synthesis for unsupervised person re-identification. *arXiv preprint arXiv:1804.10094*, 2018. 2

[7] X. Cao, Z. Wang, Y. Zhao, and F. Su. Scale aggregation network for accurate and efficient crowd counting. In *Proceedings of the European Conference on Computer Vision*, pages 734–750, 2018. 1, 2, 5, 6

[8] A. B. Chan, Z.-S. J. Liang, and N. Vasconcelos. Privacy preserving crowd monitoring: Counting people without people models or tracking. In *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition*, pages 1–7, 2008. 1, 2

[9] K. Chen, C. C. Loy, S. Gong, and T. Xiang. Feature mining for localised crowd counting. In *Proceedings of the British Machine Vision Conference*, volume 1, page 3, 2012. 1, 2

[10] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei. Imagenet: A large-scale hierarchical image database. In *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition*, pages 248–255, 2009. 5

[11] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio. Generative adversarial nets. In *Proceedings of the Advances in Neural Information Processing Systems*, pages 2672–2680, 2014. 6

[12] K. He, X. Zhang, S. Ren, and J. Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition*, pages 770–778, 2016. 4, 5

[13] G. Huang, Z. Liu, L. Van Der Maaten, and K. Q. Weinberger. Densely connected convolutional networks. In *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition*, pages 4700–4708, 2017. 5

[14] H. Idrees, I. Saleemi, C. Seibert, and M. Shah. Multi-source multi-scale counting in extremely dense crowd images. In *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition*, pages 2547–2554, 2013. 1, 2, 5, 7

[15] H. Idrees, M. Tayyab, K. Athrey, D. Zhang, S. Al-Maadeed, N. Rajpoot, and M. Shah. Composition loss for counting, density map estimation and localization in dense crowds. *arXiv preprint arXiv:1808.01050*, 2018. 1, 2, 5, 7

[16] M. Johnson-Roberson, C. Barto, R. Mehta, S. N. Sridhar, K. Rosaen, and R. Vasudevan. Driving in the matrix: Can virtual worlds replace human-generated annotations for real world tasks? In *Proceedings of the IEEE International Conference on Robotics and Automation*, pages 1–8, 2017. 2, 3

[17] J. C. S. J. Junior, S. R. Musse, and C. R. Jung. Crowd analysis using computer vision techniques. *Signal Processing Magazine IEEE*, 27(5):66–77, 2010. 1

[18] X. Li, M. Chen, F. Nie, and Q. Wang. A multiview-based parameter free framework for group detection. In *Proceedings of the Thirty-First AAAI Conference on Artificial Intelligence*, pages 4147–4153, 2017. 1

[19] Y. Li, X. Zhang, and D. Chen. Csrnet: Dilated convolutional neural networks for understanding the highly congested scenes. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1091–1100, 2018. 2, 4, 5

[20] J. Liu, C. Gao, D. Meng, and A. G. Hauptmann. Decidenet: Counting varying density crowds through attention guided detection and density estimation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 5197–5206, 2018. 7

[21] L. Liu, H. Wang, G. Li, W. Ouyang, and L. Lin. Crowd counting using deep recurrent spatial-aware network. *arXiv preprint arXiv:1807.00601*, 2018. 5

[22] X. Liu, J. van de Weijer, and A. D. Bagdanov. Leveraging unlabeled data for crowd counting by learning to rank. *arXiv preprint arXiv:1803.03095*, 2018. 2

[23] J. Long, E. Shelhamer, and T. Darrell. Fully convolutional networks for semantic segmentation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3431–3440, 2015. 4

[24] M. Marsden, K. McGuinness, S. Little, and N. E. O'Connor. Fully convolutional crowd counting on highly congested scenes. *arXiv preprint arXiv:1612.00220*, 2016. 4

[25] X. Pan, J. Shi, P. Luo, X. Wang, and X. Tang. Spatial as deep: Spatial cnn for traffic scene understanding. *arXiv preprint arXiv:1712.06080*, 2017. 4

[26] V. Ranjan, H. Le, and M. Hoai. Iterative crowd counting. *arXiv preprint arXiv:1807.09959*, 2018. 2

[27] S. R. Richter, Z. Hayder, and V. Koltun. Playing for benchmarks. In *Proceedings of the International conference on computer vision*, volume 2, 2017. 2, 3

[28] S. R. Richter, V. Vineet, S. Roth, and V. Koltun. Playing for data: Ground truth from computer games. In *Proceedings of the European Conference on Computer Vision*, pages 102–118, 2016. 2, 3

[29] M. Rodriguez, J. Sivic, I. Laptev, and J. Audibert. Data-driven crowd analysis in videos. In *IEEE International Conference on Computer Vision*, pages 1235–1242, 2011. 1

[30] G. Ros, L. Sellart, J. Materzynska, D. Vazquez, and A. M. Lopez. 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*, pages 3234–3243, 2016. 2

[31] D. B. Sam, S. Surya, and R. V. Babu. Switching convolutional neural network for crowd counting. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, volume 1, page 6, 2017. 1, 7

[32] Z. Shen, Y. Xu, B. Ni, M. Wang, J. Hu, and X. Yang. Crowd counting via adversarial cross-scale consistency pursuit. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 5245–5254, 2018. 1, 5

[33] Z. Shi, L. Zhang, Y. Liu, X. Cao, Y. Ye, M.-M. Cheng, and G. Zheng. Crowd counting with deep negative correlation learning. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 5382–5390, 2018. 1, 2, 5

[34] K. Simonyan and A. Zisserman. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*, 2014. 4, 5

[35] V. A. Sindagi and V. M. Patel. Cnn-based cascaded multi-task learning of high-level prior and density estimation for crowd counting. In *Proceedings of the IEEE International Conference on Advanced Video and Signal Based Surveillance*, pages 1–6, 2017. 2, 7

[36] V. A. Sindagi and V. M. Patel. Generating high-quality crowd density maps using contextual pyramid cnns. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 1879–1888, 2017. 2, 6, 7

[37] Q. Wang, M. Chen, F. Nie, and X. Li. Detecting coherent groups in crowd scenes by multiview clustering. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2018. 1

[38] Q. Wang, J. Wan, and Y. Yuan. Deep metric learning for crowdedness regression. *IEEE Transactions on Circuits and Systems for Video Technology*, 28(10):2633–2643, 2018. 1

[39] Z. Wang, A. C. Bovik, H. R. Sheikh, and E. P. Simoncelli. Image quality assessment: from error visibility to structural similarity. *IEEE transactions on image processing*, 13(4):600–612, 2004. 6, 7

[40] L. Zeng, X. Xu, B. Cai, S. Qiu, and T. Zhang. Multi-scale convolutional neural networks for crowd counting. In *Proceedings of the IEEE International Conference on Image Processing*, pages 465–469, 2017. 4

[41] C. Zhang, K. Kang, H. Li, X. Wang, R. Xie, and X. Yang. Data-driven crowd understanding: a baseline for a large-scale crowd dataset. *IEEE Transactions on Multimedia*, 18(6):1048–1061, 2016. 1, 2, 5, 7

[42] C. Zhang, H. Li, X. Wang, and X. Yang. Cross-scene crowd counting via deep convolutional neural networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 833–841, 2015. 2

[43] Y. Zhang, D. Zhou, S. Chen, S. Gao, and Y. Ma. Single-image crowd counting via multi-column convolutional neural network. In *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition*, pages 589–597, 2016. 1, 2, 4, 5, 7

[44] J.-Y. Zhu, T. Park, P. Isola, and A. A. Efros. Unpaired image-to-image translation using cycle-consistent adversarial networks. *arXiv preprint*, 2017. 6, 7, 8