

# Reverse Perspective Network for Perspective-Aware Object Counting

Yifan Yang<sup>1</sup>, Guorong Li<sup>\*1,2</sup>, Zhe Wu<sup>1</sup>, Li Su<sup>1,2</sup>, Qingming Huang<sup>1,2,3</sup>, Nicu Sebe<sup>4</sup>

<sup>1</sup>School of Computer Science and Technology, UCAS, Beijing, China

<sup>2</sup>Key Lab of Big Data Mining and Knowledge Management, UCAS, Beijing, China

<sup>3</sup>Key Lab of Intelligent Information Processing, Institute of Computing Technology, CAS, Beijing, China

<sup>4</sup>University of Trento, Trento, Italy

{yangyifan16, wuzhe14}@mailsucas.ac.cn, {liguorong, sul, qmhuang}@ucas.ac.cn, niculae.sebe@unitn.it

## Abstract

One of the critical challenges of object counting is the dramatic scale variations, which is introduced by arbitrary perspectives. We propose a reverse perspective network to solve the scale variations of input images, instead of generating perspective maps to smooth final outputs. The reverse perspective network explicitly evaluates the perspective distortions, and efficiently corrects the distortions by uniformly warping the input images. Then the proposed network delivers images with similar instance scales to the regressor. Thus the regression network doesn't need multi-scale receptive fields to match the various scales. Besides, to further solve the scale problem of more congested areas, we enhance the corresponding regions of ground-truth with the evaluation errors. Then we force the regressor to learn from the augmented ground-truth via an adversarial process. Furthermore, to verify the proposed model, we collected a vehicle counting dataset based on Unmanned Aerial Vehicles (UAVs). The proposed dataset has fierce scale variations. Extensive experimental results on four benchmark datasets show the improvements of our method against the state-of-the-arts.

## 1. Introduction

Counting objects is a hot topic in computer vision because of its wide applications in many areas. The most critical challenge of the counting task is scale variations. Significant effort has been devoted to addressing this issue [30, 19, 20, 32, 29, 4, 23, 13]. These approaches employ detection framework [20, 24] or regression framework [30, 19, 20].

Several approaches [39, 4, 18, 21] employ a network with multi-scale receptive fields to adapt to the various

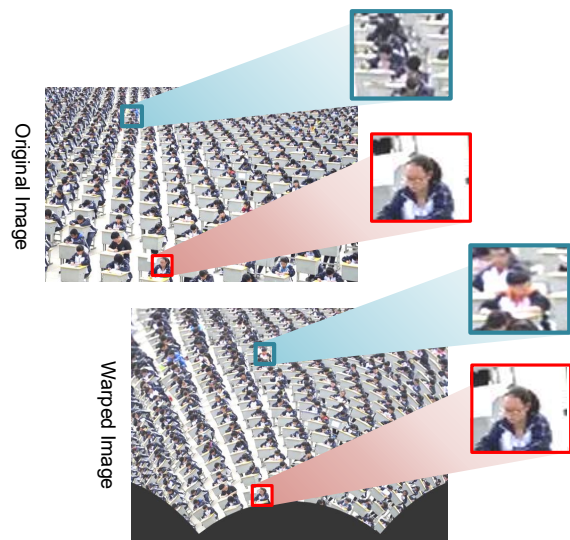


Figure 1: To diminish the scale variation, we uniformly expand the space near the horizon and shrink the space near the lens. The warped image with similar scales releases the burden of training a regression network.

scales. These methods significantly increase the computation cost brought by learning implicit perspective representations. However, Li et al. [19] prove that the various reception fields deliver similar results. Several other methods [31, 37, 36, 8] exploit perspective maps to normalize the final density maps and achieve accuracy improvements. However, these approaches learn from extra annotations or density maps to generate perspective maps in a supervised way. Besides, the generated perspective maps are highly noised. Therefore, smoothing the outputs with these perspective maps inevitably introduce noises.

As shown in Fig. 1, in the original image, there is significant scale variation caused by the perspective. Due to networks share the convolution kernels across space, it is challenging for them to adapt to the continuous scales. In-

\*Corresponding author.

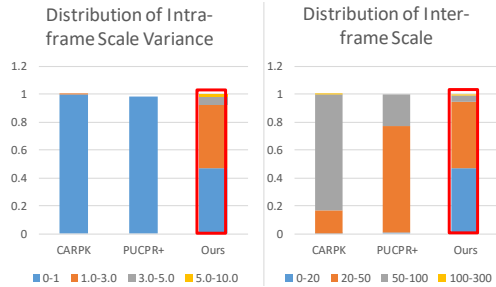


Figure 2: Compared with the existing datasets, the proposed dataset has more dramatic scale variations.

inspired by a drawing style named reverse perspective [35], in which all the objects have similar scales despite their various locations, we seek to transform the input images to obtain similar instance scales. While reversing the perspective, it is critical to maintain local structures to avoid introducing distortions. Therefore, we uniformly warp the images. As shown in Fig. 1, the warped image has similar instance scales. Therefore, unlike the multi-branch approaches [39, 4, 18, 21], the regression network doesn't need different receptive fields to adapt to the various scales. Thus we efficiently reduce the model complexity.

In this paper, we propose a reverse perspective network to diminish the scale variations of input images in an unsupervised way. Concretely, the proposed network consists of a perspective estimator and a coordinate transformer. The perspective estimator firstly evaluates the degrees of perspective distortions to obtain perspective factors. Then, guided by the perspective factors, the coordinate transformer warps the input images to obtain similar instance scales. After this transformation, we employ a single branch fully convolutional network to estimate the density maps. Before training the regressor, we pre-train the reverse perspective network to learn to correct the perspective distortions. As the perspective information is rarely available, we propose an objective function to optimize the reverse perspective network in an unsupervised way. Besides, as the reverse perspective network is a lightweight network and is easy to overfit, we look upon the perspective correction problem as a few-shot learning approach and train the proposed method through meta-learning.

The reverse perspective network still has limitations while encountering seriously crowded regions. To further solve the scale problem in these areas, we strengthen the ground-truth with the evaluation errors. Then the proposed framework forces the regressor to learn from the augmented density maps via an adversarial network.

To verify the ability of the reverse perspective network to solve scale variations, we collected a vehicle counting dataset with dramatic scale variations<sup>1</sup>, based on UAVs and

<sup>1</sup>To diminish the influence of the inter-scene scale variation, while eval-

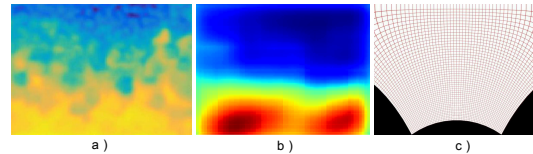


Figure 3: a) and b) are the perspective maps generated by PACNN [31] and PGCNet [36] respectively, which are highly noisy. The proposed network warps a uniform grid map, and c) is the warped image.

named UAVVC, which has more dramatic scale variations than the existing datasets (see Fig. 2). Our proposed method achieves state of the art results.

The main contributions of our method are summarized as follows:

- We propose a reverse perspective network to diminish the scale variations of input images in an unsupervised way. Therefore, we efficiently reduce the model complexity of the regression network.
- To improve the estimation accuracies in more congested areas, we strengthen the ground-truth with evaluation errors and force the regressor to learn from it via an adversarial network.
- To verify the ability of the proposed method to handle scale variations, we collect a vehicle counting dataset based on UAVs, which has dramatic scale variations.

## 2. Related Work

We categorize the object counting literature into scale-aware approaches and perspective-aware methods. In this section, we analyze the methods in both trends.

### 2.1. Scale-Aware Approaches

Most existing approaches solve the scale variations by employing a network with various receptive fields. Zhang et al. [39] and Sam et al. [28] employ multi-column networks with various kernel sizes, while Deb et al. [7] expand the multi-column network with different dilation rates. Otherwise, several other approaches utilize inception blocks to gain different receptive fields [4, 18, 21]. These algorithms significantly increase the computation cost brought by learning implicit perspective representations and want the receptive fields to match the corresponding scales automatically. However, Li et al. [19] prove that the various receptive fields deliver similar results. Besides changing the convolution kernels, a deep network can also obtain various receptive fields from its different layers. Many counting approaches utilize similar architectures with U-net [27] to count crowds and achieve promising performance

uating the intra-frame scale standard variances, we normalize the object scales in each frame with the smallest instance scale.

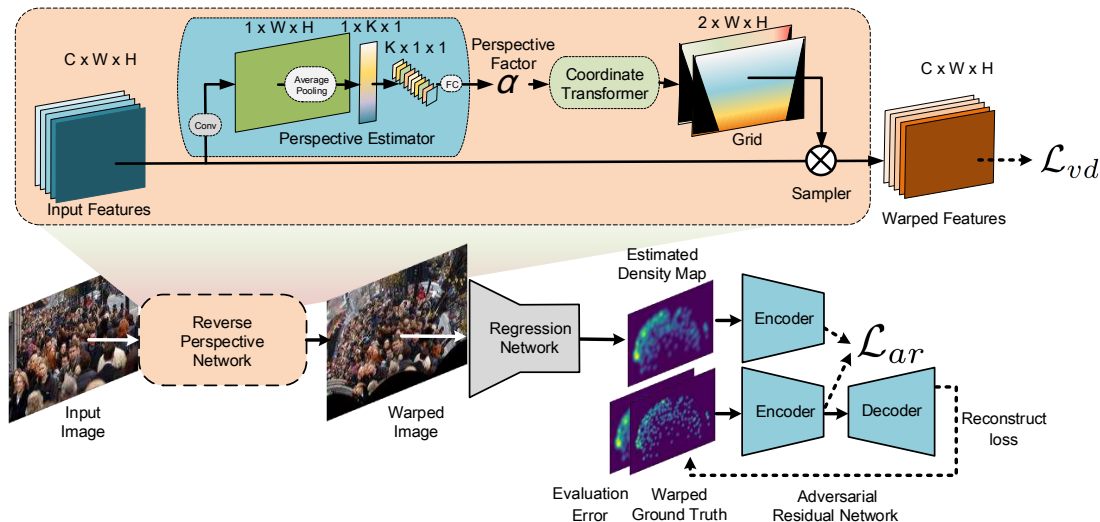


Figure 4: The framework of the proposed method. The reverse perspective network estimates the perspective factor of input images and warps the images with generated sampling grids. The regressor then estimates the crowd count in the warped images. We further deliver the estimation results and strengthen ground-truth to the adversarial residual network. The reverse perspective network is pre-trained through unsupervised meta-learning, while the regressor is trained with pixel-loss and the adversarial network.

[30, 4, 22, 18]. Moreover, some approaches employ multi-agents to handle various scales. Sam et al. [29] and Shi et al. [32] train a set of regressors to match the image patches with various densities. Liu et al. [20] use a detector and a regressor to estimate the crowd count and propose a decinet to merge the estimation results of two agents. However, these approaches achieve restricted scale diversity, comparing with the continuous scales.

In this work, we uniformly transform the images to obtain similar scales. The warped images reduce the model complexity as well as the burden of training the regressor to adapt to the continuous scales.

## 2.2. Perspective-Aware Approaches

Several algorithms have been proposed to handle the scale variations with perspective information. Shen et al. [30], Sam et al. [29] and Cao et al. [4] split an image into several patches and estimate the crowd count in them. A narrow view angle relieves the variation of target scale, yet the coarse scene splitting strategy has limited influence on the arbitrary perspectives. Besides, some approaches employ perspective maps to normalize the output density maps. Liu et al. [22] design a branch to integrate perspective maps into density maps. However, perspective maps are rarely available. Shi et al. [31] and Zhang et al. [37] generate perspective maps with extra annotations. Yan et al. [36] generate perspective maps from density maps. However, as shown in Fig. 3, the generated perspective maps are highly noised. Therefore, smoothing the outputs with these perspective maps inevitably introduce noises.

Different from them, we evaluate the perspective distortions in an unsupervised way, and uniformly correct the distortions to avoid introducing new deformities.

tion in an unsupervised way, and uniformly correct the distortions to avoid introducing new deformities.

## 3. Method

The framework of our proposed method is shown in Fig. 4. We employ a single branch regressor to evaluate the crowd count. Before the regressor, the reverse perspective network effectively diminishes the scale variations of input images. Thus the proposed network reduces the model complexity and releases the burden of training the regressor to adapt to the continuous scales. Concretely, the reverse perspective network first evaluates the perspective distortions of input images and generates grid maps for sampling. The proposed network then samples the original images to diminish the scale variations, and the regressor network evaluates the crowd count in the warped images. However, the reverse perspective network still has limitations while encountering regions with serious scale variations. We further address this issue by forcing the regressor to learn from augmented ground-truth via an adversarial network. In this section, we first elaborate on the reverse perspective network and the training method. In the last subsection, we explain the adversarial residual learning method.

### 3.1. Reverse Perspective Network

As shown in Fig. 4, the reverse perspective network consists of a perspective estimator and a coordinate transformer. Both components are trained end-to-end with unsupervised meta-learning.

A key observation is that the information about objects

is embedded in intermediate CNN features of classification networks [3, 6, 9]. Correspondingly, we evaluate the spatial capacity with the features extracted by the first ten convolutional layers of the pre-trained VGG-16 [33]. We denote the features as  $G(X; \psi) \in \mathbb{R}^{CWH}$ , where  $\psi$  denotes the parameters of the convolutional layers,  $X$  is the input images,  $C, W, H$  stand for the number of channels, width, and height respectively.

While training, the reverse perspective network warps the extracted features to estimate space capacities. In the reference stage, we transfer the input image instead.

**Perspective Estimation** As perspective information is rarely available, we estimate the spatial capacity to predict the perspective. The perspective estimator firstly regresses a class-agnostic density map from the extracted features via a convolutional network. The density map is formulated as:

$$Y^{CL} = f_d(G(X, \psi), \zeta), \quad (1)$$

where  $Y^{CL} \in \mathbb{R}^{1WH}$ ,  $f_d(\cdot)$  denotes dilated convolution operation, and  $\zeta$  is the parameters of the convolutional layers.

As  $Y^{CL}$  reveals the spatial capacity rather than the count of specific objects, we exploit the perspective information from it. Images are commonly captured with approaching horizontal viewpoints, so the spatial capacity linearly transforms vertically. Consequently, we embed the vertical capacity of an image into a capacity vector with an adaptive average pooling operation. The capacity vector is defined as:

$$c_v = \text{AdaptiveAveragePooling}(Y^{CL}, K), \quad (2)$$

where  $K$  is a pre-defined vertical dimension. As the horizon vector of  $Y^{CL}$  is pooled to a single element,  $c_v \in \mathbb{R}^{1K1}$ . The pooling operation embeds an image with an arbitrary resolution to a scale fixed capacity vector.

Afterward, we transfer the vector to capacity features with  $K$  channels, marked as  $c_v^T \in \mathbb{R}^{K11}$ . The perspective estimator then employs fully connected layers to regress the perspective factor  $\alpha_o$  as follow:

$$\alpha_o = f_p(c_v^T, \eta), \quad (3)$$

where  $f_p(\cdot)$  is the full connected operation, and  $\eta$  indicates the parameters. Since the perspective factor has a clear geometric meaning and guides the coordinate transformer to warp the images, we normalize the factor to a range of  $(0, 1]$ :

$$\alpha = \exp(-\text{relu}(\alpha_o)) + \epsilon, \quad (4)$$

where  $\epsilon$  is a constant close to zero.

**Coordinate Transformer** As the instance scales are related to the capacity of their belonging space, we shrink the space near the lens to diminish the perspective distortions. A toy example with a dramatic scale variation is shown in

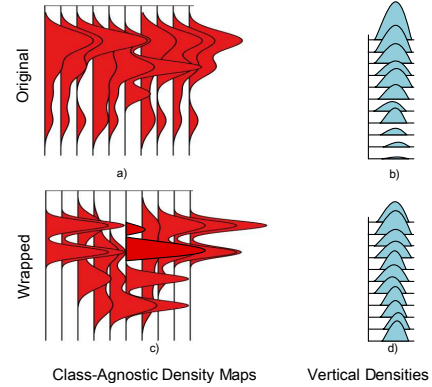


Figure 5: Sketch of a toy example, which simulates the changing of densities and capacities of a density map during coordinates transformation.

Fig. 5(a). The higher density of the upper area indicates smaller instance scales, while the down area with lower density contains relative bigger objects. When we compress the down area in Fig. 5(b), the density increases, and the scale variation is diminished as well.

While correcting the perspective distortions, it is critical to maintain local structures to avoid introducing new deformities. Therefore, we employ a coordinate transformation method to uniformly compress the vertical space of images and maintain the horizontal structure. Notably, the transformation method reconstructs the original image from elliptic coordinates to the cartesian coordinates.

Here, we first explain the elliptic coordinate. In geometry, the elliptic coordinate system is a two-dimensional orthogonal coordinate system in which the coordinate lines are confocal ellipses and hyperbolae. The two foci are fixed, and the focal length is denoted as  $2c$ . The most common coordinate system  $(\mu, \nu)$  for elliptic curves is defined as:

$$\begin{aligned} x &= c * \cosh(\mu) \cos(\nu) \\ y &= c * \sinh(\mu) \sin(\nu), \end{aligned} \quad (5)$$

where  $\mu$  is a nonnegative real number which denotes the distance from the pixel to the centre,  $\nu \in [0, 2\pi]$  is the corresponding eccentric angle. When transforming from elliptic coordinates to the cartesian coordinates, the space near the x-axis virtually stays the same, yet the space far from the x-axis is compressed.

The perspective factor is utilized as an adjustment term to guide the perspective correction, and is given the geometry meaning of ratio of the focal length to the image width. As shown in Fig. 6, as the radius is fixed, the perspective factor decides the curvature of confocal ellipses. In other word, it determines the intensity of deformations, and when the perspective factor is smaller, the deformation is more fierce. Moreover, we normalize the  $X$  coordinate to  $X_E \in [0, \pi]$ , and  $Y$  coordinate to  $Y_E \in [0, 1]$ . Therefore,

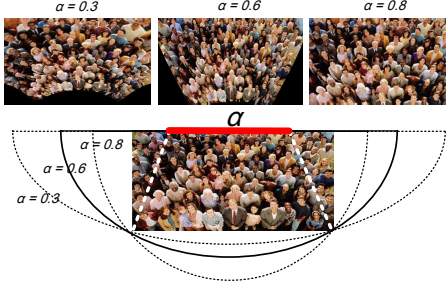


Figure 6: The influence of different perspective factors on the coordinates transformation.

the transformation is formulated as:

$$\begin{aligned} x_{CT} &= \frac{\alpha}{2} * \cosh(x_E) \cos(y_E) \\ y_{CT} &= \frac{\alpha}{2} * \sinh(x_E) \sin(y_E), \end{aligned} \quad (6)$$

where  $(x_{CT}, y_{CT})$  are the cartesian coordinates of the warped image.

We employ the transformation method to generate the grid maps of the new image, then employ the differentiable sampler, which was proposed by Jaderberg et al. [16], to warp the original image.

Ideally, the reverse perspective network can be trained end-to-end with the regression network. However, such a strategy ignores the internal structure of the perspective maps and results in inferior results. Otherwise, we warp the images rather than the features to maintain the semantic information in deeper features.

### 3.2. Training Method

**Unsupervised Loss** The ground-truth perspective factors are rarely available. Thus we propose an unsupervised loss in this subsection to train the reverse perspective network without extra annotations.

While an image row is compressed, the activated features are constrained to the central part, and the row density of the active region is changed as well. Based on this observation, we exploit the row densities of the warped features to reveal the global density distribution. We first formulate the average row density as:

$$D_j^H = \frac{1}{w_j} \sum_{i=1}^W \sum_{k=1}^C G_{warped}(i, j, k), \quad (7)$$

where  $G_{warped}(i, j, k)$  stands for warped features, and  $i, j, k$  are the indexes of column, row, channel respectively. Moreover,  $D_j^H$  is the average density of j-th row, and  $w_j$  is the number of effective pixels in this row. Accordingly, the row density vector of warped features is defined as  $D_H = [D_1^H, D_2^H, \dots, D_W^H]$ .

As shown in Fig. 5, subfigures (b) and (d) are the corresponding row density vectors of (a) and (c). A small variance of the row density vector indicates more uniform scale



Figure 7: The two example images with various perspectives. The first row is the outputs of the proposed network which is trained with SGD method. While the second row is the outputs of the proposed method, which trained with meta-learning.

distributions in the frames. Thus we devise an unsupervised regression loss to constrain the row density vector of the warped features. We formulate the objective function as:

$$\mathcal{L}_{vd} = Var(D^H). \quad (8)$$

**Meta Learning** It is desirable to employ the standard training process, for instance, SGD method, to train the reverse perspective network. However, the reverse perspective is a lightweight network and is easy to overfit. As shown in Fig. 7, the reverse perspective network, which is trained by the SGD, generates similar perspective factors.

To address the above issues, we treat perspective correction as a few-shot learning problem and employ a meta-learning algorithm to optimize the proposed network. The meta-learning specializes in learning from small amounts of data, and the learned meta-model can fast adapt to indeterminate scenes. In the pre-training stage, we split the data into several small tasks, and employ Reptile [25], which is an efficient meta-learning algorithm, to train our regression network with only raw and unlabeled observations. Afterward, we fine-tune the learned meta-model to adapt to each image in both training and reference stages.

In short, Reptile averages several updates of the network, instead of updating the network on the average loss function. This strategy is guaranteeing to learn an initialization that is good for fast adaptations. The Reptile method  $\mathcal{M}(\cdot)$  randomly samples the training dataset into p-tasks  $\mathcal{T}_p$ . During training,  $\mathcal{M}(\cdot)$  inputs a set of tasks  $\mathcal{T}_i$ , and produces a learning procedure  $SGD(L, \theta, k)$ . The learning procedure performs  $k$  gradient steps on loss  $L$  starting with  $\theta$  and returns the final parameter vector. The training process on a single task  $\tau$  is formulated as:

$$\begin{aligned} W &= SGD(L_\tau, \theta, k) \\ \hat{\theta} &= \theta + \eta(\theta - W), \end{aligned} \quad (9)$$

where the  $\hat{\theta}$  is the updated parameters, and  $\eta$  is a scalar.

Benefiting from the sample and efficient Reptile method, we obtain a more sensitive meta-model, which is fine-tuned on each test image.



Figure 9: Representative frames of UAVVC, which have different perspectives, for instance, high latitude, front-view in low latitude, side-view, and top-view.

### 3.3. Adversarial Residual Learning

The reverse perspective network still has limitations while encountering seriously crowded regions. Therefore, we augment the ground-truth density map with the evaluation error, which highlights the congested area. The augmented ground-truth is formulated as:

$$Y_{Aug} = Y + \varphi(Y - \hat{Y}), \quad (10)$$

where  $Y$  is the ground-truth density map,  $\hat{Y}$  is the evaluation output, and  $\varphi$  is a scalar.

To force the regressor to learn from the augmented ground-truth, we employ an adversarial structure named Adversarial Structure Matching (ASM) [14] to extract the structures of congested areas. ASM employs an auto encoder-decoder framework. The encoder analyzes the structure of both estimated and the augmented density maps, while the decoder reconstructs the augmented density maps from the hidden features. While training, the regressor is trained to minimize the distance between hidden features of the evaluation and the augmented ground-truth, while the encoder is trained to enlarge the distance. The adversarial loss function is formulated as:

$$\max_A \min_R \mathcal{L}_{ar} = \mathbb{E}_{XY} [|A(R(X)) - A(Y_{Aug})|], \quad (11)$$

where  $R(\cdot)$  is the regression network,  $A(\cdot)$  is the encoder, and  $X$  is the input image. The adversarial approach forces the regression network to pay more attention to the objects that are easily overlooked.

## 4. UAV-based Vehicle Counting Dataset

There are limited datasets for the vehicle counting task. Among them TRANCOS\_v3 [10] is collected by fixed cameras, while CARPK [12] and PUCPR+ [1] contain images captured by UAVs. We show a representative frame of each dataset in Fig. 8. As shown, they only present constrained traffic circumstances with narrow viewpoints, which have limited scale variations, as such, are not suitable to verify the proposed method.

To fill this gap, we collected a UAV-based dataset with 50 different scenarios for vehicle counting, termed as UAVVC. The proposed dataset is manually annotated with 32,770 bounding boxes, instead of traditional point annotations. We show representative frames in Fig. 9. The proposed dataset has four kinds of perspectives, i.e., front-view in



Figure 8: Representative frame of TRANCOS\_v3, CARPK, and PUCPR+, which have limited perspectives.

Table 1: Summary of existing related datasets.

Dataset	Resolution	Frames	View-Points	Scenes	Weather	Notation
TRANCOS	640x480	824	fixed	43	2	Points
CARPK	1,280x720	1,448	fixed	7	1	BoundingBox
PUCPR+	1,280x720	125	fixed	1	3	BoundingBox
<b>Ours</b>	1,024x540	738	<b>flexible</b>	<b>50</b>	<b>5</b>	BoundingBox

Table 2: The evaluation results on ShanghaiTech datasets

Method	ShanghaiTech Part A		ShanghaiTech Part B	
	MAE	MSE	MAE	MSE
MCMM [39]	110.2	173.2	26.4	41.3
SANet [4]	67.0	104.5	8.4	13.6
CSRNet [19]	68.2	115.0	10.6	16.0
IG-CNN [26]	72.5	118.2	13.6	21.1
TEDnet [18]	64.2	109.1	8.2	12.8
ADCrowdNet [21]	63.2	98.9	8.2	15.7
CG-DRCN [34]	64.0	98.4	8.5	14.4
ADMG [17]	64.7	97.1	8.1	13.6
ANF [2]	63.9	99.4	8.3	13.2
<b>Ours</b>	<b>61.2</b>	<b>96.9</b>	<b>8.1</b>	<b>11.6</b>

high latitude, front-view in low latitude, side-view, and top-view. With the unconstrained viewing points, there are dramatic scale variations. The frames are also collected in different weather conditions, for instance, sunny, raining, fog, night, and raining night. All these shooting conditions increase the diversity of the dataset and make it closer to the real traffic circumstances. More importantly, the dramatic scale variation can be used to verify our proposed method.

The statistics of our proposed dataset and existing datasets are shown in Table 1. The UAVVC is more diverse and provides more precise annotations. Moreover, three critical distributions are shown in Fig. 2. As shown, the proposed dataset has more dramatic scale variations, also is more balanced than others in all the criterions. The dataset and all the experimental results are available on <https://github.com/CrowdCounting>.

## 5. Experiments

We evaluate our approach on four datasets: ShanghaiTech [39], WorldExpo10 [15], UCSD [5], and the proposed UAVVC. In this section, we first evaluate and compare our method with the previous state-of-the-art approaches [39, 4, 26, 21, 19, 18, 34, 17, 2] on these datasets. Then we present ablation study results on ShanghaiTech

Table 3: The MAE evaluation results on the WorldExpo10 dataset.

Method	Scce.1	Scce.2	Scce.3	Scce.4	Scce.5	Avg.
MCMM [39]	3.4	20.6	12.9	13.0	8.1	11.6
SANet [4]	2.6	13.2	9.0	13.3	3.0	8.2
CSRNet [19]	2.9	11.5	8.6	16.6	3.4	8.6
IG-CNN [26]	2.6	16.1	10.15	20.2	7.6	11.3
TEDnet [18]	2.3	10.1	11.3	13.8	2.6	8.0
ADCrowdNet [21]	1.6	15.8	11.0	10.9	3.2	8.5
ADMG [17]	4.0	18.1	7.2	12.3	5.7	9.5
Ours	2.4	10.2	9.7	11.5	3.8	<b>8.2</b>



Figure 10: Examples in ShanghaiTech dataset. The first column is the input images, and the second column is the warped images, while the third column is the estimated density maps. The proposed method adaptively warps the images and delivers accurate estimations.

Part A dataset. In these experiments, we evaluate the performance with MAE and MSE metrics.

### 5.1. Implementation details

We employ CSRNet as our regression network. The most time-consuming operation of the proposed method is generating the grid maps. During transformation, we first generate initial grid maps with a resolution of  $10^*10$ , then enlarge them to the desired resolution. Therefore, the reverse perspective network introduces limited time complexity. The reverse perspective network is pre-trained on the training set of each dataset, respectively. For meta-learning, we set the inner batch size to 5, inner iterations to 20, and the learning rate to  $1e-6$ . When training the regression network, we also warp the ground-truth density maps with the generated grid maps. The learning rate is set to  $1e-7$ , and the  $\varphi$  is set to  $1e-6$ , while the  $K$  is set to 20.

### 5.2. Evaluation and comparison

**ShanghaiTech** dataset [37] focuses on the pedestrian counting. There are 1,198 images with different perspectives and resolutions. This dataset has two parts named Part A and Part B. We report the comparison between our method and state-of-arts in Table 2. Our method achieves the lowest MAE (the highest accuracy) on both parts. Fig. 10 shows samples of Part A and Part B. It can be seen that the proposed method is well adaptable to arbitrary perspectives.

**WorldExpo10** dataset [39] has 3,980 annotated frames.

Table 4: The evaluation results on the UCSD dataset

Method	MAE	MSE
MCMM [39]	1.07	1.35
SANet [4]	1.02	1.29
CSRNet [19]	1.16	1.47
ADCrowdNet [21]	<b>0.98</b>	1.25
Ours	1.32	<b>1.23</b>

Table 5: The evaluation results on UAVVC dataset, where IFSV stands for intra-frame scale variance and the second to the fifth columns are the corresponding MAE results.

Method	IFSV	IFSV	IFSV	IFSV	Total MAE	Total MSE
	$\in [0.0, 1.0]$	$\in [1.0, 3.0]$	$\in [3.0, 5.0]$	$\in [5.0, 10.0]$		
MCMM [39]	32.14	46.37	64.48	61.22	50.63	85.46
VGG-16 [38]	51.32	62.41	80.34	76.31	67.12	103.20
CSRNet [19]	13.11	19.61	28.02	25.65	18.32	32.27
Ours	<b>12.98</b>	<b>13.26</b>	<b>21.00</b>	<b>19.70</b>	<b>13.21</b>	<b>20.07</b>

We divide this dataset into a training set with 3,380 frames and a testing set with 600 frames. We list the result comparisons of MAE in Table 3, where our method achieves the best 8.2 average MAE against other methods. Some of the scenes are sparse, and the MAE is already quite low, being thus hard to achieve a significant improvement.

**UCSD** dataset [5] contains 2,000 frames captured by surveillance cameras, and the frames have the same perspective. The comparison between state-of-the-art crowd counting methods and our method is summarized in Table 4. Our approach overall performs comparably with the best algorithm, but much better than the others. This is because there are limited scale variations in the UCSD dataset, so our method achieves limited improvements.

**UAVVC** is our newly proposed dataset with 500 and 385 images for training / testing. Table 5 shows the comparisons of our method against MCNN [39], VGG-16 [33] and CSRNet [19], and our method achieves the best results. More importantly, according to intra-frame scale standard variance, we classify the testing images into four groups and respectively evaluate the MAE metric on each group. Our proposed method achieves improvement in each group. Especially on the image group with larger-scale variation, the performance improvements of our method are more prominent. These experiments verify the ability of the proposed algorithm to handle dramatic scale variations. We show three examples in Fig. 11, each of which has a unique perspective. In the dataset, we do not provide the region of interesting masks. Thus it is challenging for the regressors to recognize the objects in complicated scenes. As shown, compared with the baseline [19], our method delivers more accurate results, and is less likely to recognize backgrounds as vehicles. The similar scales in warped images facilitate the regressor to tell the vehicles from backgrounds.

### 5.3. Ablation study

In this section, we conduct several experiments to study the effect of different aspects of our method on ShanghaiTech Part A and show the results in Table 6.

**Baseline** Our method achieves 10.3% accuracy improve-

Table 6: The ablation study results on ShanghaiTech Part A. Where RAN stands for residual adversarial network.

Method	Baseline	Ours	w/o RAN	End-to-End Trained	Warp Features	Trained with SGD	New Baseline	Modified New Baseline
MAE	68.2	<b>61.2</b>	63.4	72.1	66.2	70.6	109.6	94.3

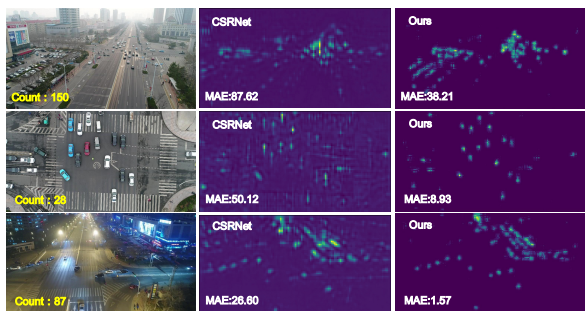


Figure 11: Examples in UAVVC. In each row, the first image is the input image, the second one is the output of CSRNet, and the third one is the output of our algorithm.



Figure 12: We show two examples in ShanghaiTech Part A. The first column is the input images, the second column is the outputs of baseline, while the third column is the warped images, and the last column is the outputs of proposed method.

ment compared with the baseline. We show the qualitative comparison between baseline and our method in Fig. 12. The two congested scenes have different perspectives, and our method achieves more accurate estimation results. More importantly, our density maps are brighter. That is because we diminish the scale variations of the input images, and the density maps of the proposed method have fewer picture contrasts.

**Without Adversarial Network** While we only employ the reverse perspective network, the MAE decreases by 7.0%, yet it is still not as good as that of ours. This is because the reverse perspective network has limitations, for instance, in the first example of Fig. 12, the most congested areas are not enough stretched, and the scale variation is still fierce. The adversarial network force the regressor to pay more attention to these areas.

**End-to-End Training** We train the reverse perspective network with the regression network in an end-to-end fashion, and the performance decreases by 5.7%. This is because such a strategy ignores the internal structure of the perspective maps and results in inferior results.

**Warp Features Instead** We pre-train the reverse perspective network, and utilize it to warp the features directly.

This strategy is less time consuming, yet it only receives 2.9% performance increase. This is because warping features wrecks the semantic information of the deep features and makes the objects unrecognizable.

**Traditional Learning Method** We pre-train the reverse perspective network with the SGD method. As a result, the network is overfitted and delivers similar perspective factors with limited variance. Thus the method obtains a 3.5% performance decrease. This is because simple averaging the loss function wears down the implicit information of perspectives and scenes.

**Extensibility** To evaluate the extensibility of the proposed framework, we employ a new backbone with various receptive fields as the backbone of CSRNet, which is the first ten convolution layers of ResNet-101 [11]. The new baseline is trained with original and warped images, respectively. As shown in Table. 6, our proposed method decreases MAE by 14.0% compared with new baseline, yet not as good as the VGG based one. Moreover, the VGG based one with warped images obtains better performance than ResNet one with original images. This experiment affirms that the proposed network is more effective than the various receptive fields.

## 6. Conclusion

In this paper, we propose a reverse perspective network to diminish the scale variations while estimating the crowd densities. The reverse perspective network estimates the perspective distortions in an unsupervised way and warps the original images to obtain similar scales. To further solve the scale issue in seriously congested areas, we force the regressor to learn from the augmented ground-truth via an adversarial network. Moreover, to verify the proposed framework, we collected a UAV-based vehicle counting dataset, which has dramatic intra-scene and inter-scene scale variations. Extensive experimental results demonstrate the state-of-the-art performance of our method. In the future work, we will research the perspective evaluation of videos, which has continuous perspectives and scenes.

**Acknowledgements** This work was supported in part by the Italy-China collaboration project TAL-ENT:2018YFE0118400, in part by National Natural Science Foundation of China: 61620106009, 61772494, 61931008, U1636214, 61836002 and 61976069, in part by Key Research Program of Frontier Sciences, CAS: QYZDJ-SSW-SYS013, in part by Youth Innovation Promotion Association CAS, in part by the University of Chinese Academy of Sciences.



## References

- [1] Paulo Almeida, Luiz S Oliveira, Alceu De Souza Britto, Eunelson J Silva, and Alessandro L Koerich. Pklot-a robust dataset for parking lot classification. *Expert Systems With Applications*, 42(11):4937–4949, 2015. 6
- [2] Zhang Anran, Yue Lei, Shen Jiayi, Zhu Fan, Zhen Xiantong, Cao Xianbin, and Shao Ling. Attentional neural fields for crowd counting. *IEEE International Conference on Computer Vision*, 2019. 6
- [3] David Bau, Bolei Zhou, Aditya Khosla, Aude Oliva, and Antonio Torralba. Network dissection: Quantifying interpretability of deep visual representations. *Computer Vision and Pattern Recognition*, 2017. 4
- [4] Xinkun Cao, Zhipeng Wang, Yanyun Zhao, and Fei Su. Scale aggregation network for accurate and efficient crowd counting. *Computer Vision and Pattern Recognition*, pages 757–773, 2018. 1, 2, 3, 6, 7
- [5] Antoni B Chan, Zhangsheng John Liang, and Nuno Vasconcelos. Privacy preserving crowd monitoring: Counting people without people models or tracking. *Computer Vision and Pattern Recognition*, pages 1–7, 2008. 6, 7
- [6] Edo Collins, Radhakrishna Achanta, and Sabine Susstrunk. Deep feature factorization for concept discovery. *European Conference on Computer Vision*, 14:352–368, 2018. 4
- [7] Diptodip Deb and Jonathan Ventura. An aggregated multicolumn dilated convolution network for perspective-free counting. *Computer Vision and Pattern Recognition*, pages 195–204, 2018. 2
- [8] Junyu Gao, Qi Wang, and Xuelong Li. Pcc net: Perspective crowd counting via spatial convolutional network. *IEEE Transactions on Circuits and Systems for Video Technology*, abs/1905.10085, 2019. 1
- [9] Abel Gonzalezgarcia, Davide Modolo, and Vittorio Ferrari. Do semantic parts emerge in convolutional neural networks. *International Journal of Computer Vision*, 126(5):476–494, 2018. 4
- [10] Ricardo Guerrerogomezolmedo, Beatriz Torrejimenez, Roberto Javier Lopezasastre, Saturnino Maldonadobascon, and Daniel Onororubio. Extremely overlapping vehicle counting. *Computer Vision and Pattern Recognition*, pages 423–431, 2015. 6
- [11] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. *Computer Vision and Pattern Recognition*, pages 770–778, 2016. 8
- [12] Meng-Ru Hsieh, Yen-Liang Lin, and Winston H. Hsu. Drone-based object counting by spatially regularized regional proposal networks. *IEEE International Conference on Computer Vision*, 2017. 6
- [13] Siyu Huang, Xi Li, Zhongfei Zhang, Fei Wu, Shenghua Gao, Rongrong Ji, and Junwei Han. Body structure aware deep crowd counting. *IEEE Transactions on Image Processing*, 27(3):1049–1059, 2018. 1
- [14] Jyhjing Hwang, Tsungwei Ke, Jianbo Shi, and Stella X Yu. Adversarial structure matching for structured prediction tasks. *Computer Vision and Pattern Recognition*, pages 4056–4065, 2019. 6
- [15] Haroon Idrees, Imran Saleemi, Cody Seibert, and Mubarak Shah. Multi-source multi-scale counting in extremely dense crowd images. *Computer Vision and Pattern Recognition*, pages 2547–2554, 2013. 6
- [16] Max Jaderberg, Karen Simonyan, Andrew Zisserman, and Koray Kavukcuoglu. Spatial transformer networks. *Neural Information Processing Systems*, pages 2017–2025, 2015. 5
- [17] Wan Jia and Chan Antoni. Adaptive density map generation for crowd counting. *IEEE International Conference on Computer Vision*, 2019. 6, 7
- [18] Xiaolong Jiang, Zehao Xiao, Baochang Zhang, Xiantong Zhen, Xianbin Cao, David Doermann, and Ling Shao. Crowd counting and density estimation by trellis encoder-decoder network. *Computer Vision and Pattern Recognition*, 2019. 1, 2, 3, 6, 7
- [19] Yuhong Li, Xiaofan Zhang, and Deming Chen. Csrnet: Dilated convolutional neural networks for understanding the highly congested scenes. *Computer Vision and Pattern Recognition*, pages 1091–1100, 2018. 1, 2, 6, 7
- [20] Jiang Liu, Chenqiang Gao, Deyu Meng, and Alexander G Hauptmann. Decidenet: Counting varying density crowds through attention guided detection and density estimation. *Computer Vision and Pattern Recognition*, pages 5197–5206, 2018. 1, 3
- [21] Ning Liu, Yongchao Long, Changqing Zou, Qun Niu, Li Pan, and Hefeng Wu. Adcrowdnet: An attention-injective deformable convolutional network for crowd understanding. *Computer Vision and Pattern Recognition*, pages 3220–3229, 2018. 1, 2, 6, 7
- [22] Weizhe Liu, Mathieu Salzmann, and Pascal Fua. Context-aware crowd counting. *Computer Vision and Pattern Recognition*, pages 5094–5103, 2019. 3
- [23] Xialei Liu, Joost Van De Weijer, and Andrew D Bagdanov. Leveraging unlabeled data for crowd counting by learning to rank. *Computer Vision and Pattern Recognition*, pages 7661–7669, 2018. 1
- [24] Yuting Liu, Miaojing Shi, Qijun Zhao, and Xiaofang Wang. Point in, box out: Beyond counting persons in crowds. *Computer Vision and Pattern Recognition*, 2019. 1
- [25] Alex Nichol and John Schulman. Reptile: a scalable meta-learning algorithm. *Learning*, 2018. 5
- [26] Viresh Ranjan, Hieu Le, and Minh Hoai. Iterative crowd counting. *European Conference on Computer Vision*, pages 278–293, 2018. 6, 7
- [27] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation. *Medical Image Computing and Computer Assisted Intervention*, pages 234–241, 2015. 2
- [28] Deepak Babu Sam and R Venkatesh Babu. Top-down feedback for crowd counting convolutional neural network. *National Conference on Artificial Intelligence*, pages 7323–7330, 2018. 2
- [29] Deepak Babu Sam, Neeraj Sajjan, R Venkatesh Babu, and Mukundhan Srinivasan. Divide and grow: Capturing huge diversity in crowd images with incrementally growing cnn. *Computer Vision and Pattern Recognition*, pages 3618–3626, 2018. 1, 3

- [30] Zan Shen, Yi Xu, Bingbing Ni, Minsi Wang, Jianguo Hu, and Xiaokang Yang. Crowd counting via adversarial cross-scale consistency pursuit. In *Computer Vision and Pattern Recognition*, pages 5245–5254, June 2018. 1, 3
- [31] Miaoqing Shi, Zhaohui Yang, Chao Xu, and Qijun Chen. Revisiting perspective information for efficient crowd counting. *Computer Vision and Pattern Recognition*, pages 7271–7280, 2018. 1, 2, 3
- [32] Zenglin Shi, Le Zhang, Yun Liu, Xiaofeng Cao, Yangdong Ye, Mingming Cheng, and Guoyan Zheng. Crowd counting with deep negative correlation learning. *Computer Vision and Pattern Recognition*, pages 5382–5390, 2018. 1, 3
- [33] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. *International Conference on Learning Representations*, abs/1409.1556, 2014. 4, 7
- [34] Sindagi Vishwanath, Yasarla Rajeev, and Patel Vishal. Pushing the frontiers of unconstrained crowd counting: New dataset and benchmark method. *IEEE International Conference on Computer Vision*, 2019. 6
- [35] Wikipedia. Reverse perspective. [https://en.wikipedia.org/wiki/Reverse\\_perspective](https://en.wikipedia.org/wiki/Reverse_perspective). 2
- [36] Zhaoyi Yan, Yuchen Yuan, Wangmeng Zuo, Xiao Tan, Yezhen Wang, Shilei Wen, and Errui Ding. Perspective-guided convolution networks for crowd counting. *Computer Vision and Pattern Recognition*, 2019. 1, 2, 3
- [37] Cong Zhang, Hongsheng Li, Xiaogang Wang, and Xiaokang Yang. Cross-scene crowd counting via deep convolutional neural networks. *Computer Vision and Pattern Recognition*, pages 833–841, 2015. 1, 3, 7
- [38] Shanghang Zhang, Guanhang Wu, Joao Paulo Costeira, and Jose M F Moura. Fcn-rlstm: Deep spatio-temporal neural networks for vehicle counting in city cameras. *International Conference on Computer Vision*, pages 3687–3696, 2017. 7
- [39] Yingying Zhang, Desen Zhou, Siqin Chen, Shenghua Gao, and Yi Ma. Single-image crowd counting via multi-column convolutional neural network. *Computer Vision and Pattern Recognition*, pages 589–597, 2016. 1, 2, 6, 7