OBJECT COUNTING IN VIDEO SURVEILLANCE USING MULTI-SCALE DENSITY MAP REGRESSION

Yi Wang[1]*, Junhui Hou*[2] *and Lap-Pui Chau*[1]

[1]School of Electrical and Electronics Engineering, Nanyang Technological University, Singapore
[2]Department of Computer Science, City University of Hong Kong
E-mail: wang1241@e.ntu.edu.sg, jh.hou@cityu.edu.hk, elpchau@ntu.edu.sg

ABSTRACT

In this paper, we present an effective convolutional neural network (CNN) for object counting in video surveillance, namely multi-scale density map regressor (MSDMR). In contrast to existing CNN-based methods that achieve high accuracy by means of empirically increasing the model capacity with more complex structures/layers, we focus on a compact CNN. Specifically, the MSDMR is mainly designed with the supervision of multi-scale outputs, in which two CNN stacks estimate coarse- and fine-scale density maps, respectively. The integral of the fine density map provides the count of objects. The two stacks are connected in a cascaded manner and jointly trained such that the overall model can learn discriminative and complementary features to produce expressive performance. Experimental results show that the proposed MSDMR can achieve higher accuracy compared with state-of-the-art methods on the surveillance datasets.

Index Terms— Object counting, video surveillance, CNN, density map, multi-scale

1. INTRODUCTION

Object counting that aims to estimate the number of objects recorded by images/videos has gained much attention in the field of intelligent video surveillance [1]. For surveillance cameras, the saved videos suffer from low resolution and low frame rate because of limited network bandwidth and storage. Moreover, the objects (e.g., people or vehicles) in the video present large variations in the size and high occlusion since cameras may be installed at busy road sections for capturing much content. This brings some challenges for counting objects in video surveillance.

Recently, researchers have explored some approaches to tackle the object counting problem, such as motion-based methods [2], detection-based methods [3, 4], and regression-based methods [5, 6]. Specifically, motion-based methods count objects by tracking which fails for the videos with low frame rate. Detection-based methods count objects by detecting its location. However, this type of methods encounter difficulties in counting small objects, especially for surveillance videos with low resolution. Regression-based methods were proposed to learn a mapping from the low-level features to the global count or the density map. Density map projects the density of objects into each pixel of the image, the intensities of which can be summed to obtain the global count.

In this paper, we propose a compact CNN model for object counting in video surveillance, namely multi-scale density map regressor (MSDMR). Our MSDMR (see Fig. 1) can be decomposed into two coherent regression stages: coarse and fine density map estimation. The coarse density map can be considered as the prior of the fine density map. These two stages are connected by employing the multi-task cascades [7]. The proposed architecture is a hourglass structure. The supervision of the coarse density map implemented in the bottleneck of the hourglass structure can partially provide the blocked information for gradient back-propagation. Moreover, this compact architecture only has about 372k parameters to be trained. The experimental results demonstrate the outstanding performance of our light model on UCSD [5] and TRANCOS [8] surveillance dataset.

2. RELATED WORK

This paper focuses on regression-based approaches, which can be separated into two categories: count regression and density map regression.

The count regression methods can map the low-level features to the count. Based on the holistic features of motion segments, Chan *et al.* [5] introduced a Gaussian Process regression to estimate the global count of crowds. Chen *et al.* [9] proposed a joint localized crowd counting by using ridge regression in a multi-output model. Unlike global count regression, this method counts people at different spatial patches. In an extreme case, the count can be model at each pixel, which can fully exploit the spatial information.

For incorporating the spatial information, the density-

This work was supported in part by the Hong Kong RGC Early Career Scheme under Grant 9048123 (CityU 21211518), and in part by the Natural Science Foundation of China under Grant 61873142.

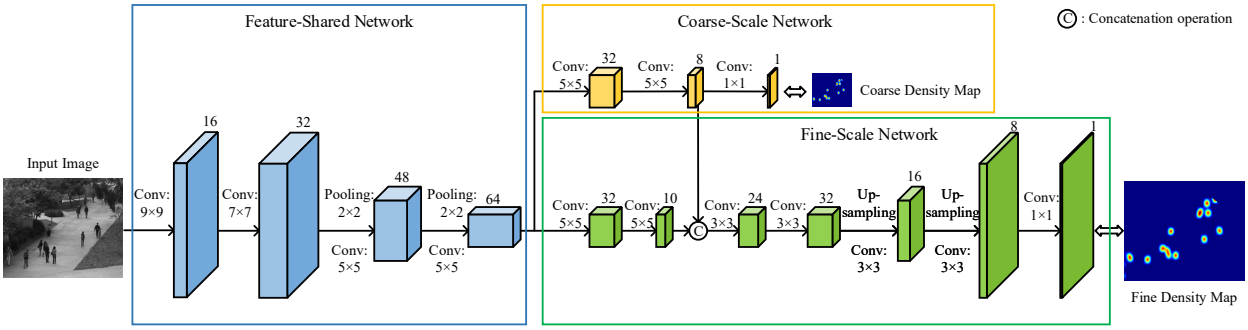

Fig. 1. Architecture of the proposed MSDMR with a cascaded structure.

map-based methods are proposed. Lempitskt and Zisserman [10] proposed a supervised learning framework, in which the density map estimation is modeled as a linear transformation from each pixel in an image to the corresponding densities, and can be defined as a minimization of a regularized risk quadratic cost function. With recent developments in deep learning, researchers take advantage of the CNN to learn data-driven features and generate accurate density maps.

CNN-based density map regression: Reference [11] introduced a CNN model with a switchable training scheme for the density and count estimation. Zhang *et al.* [6] proposed a multi-column CNN (MCNN) model, each column of which can learn the features at certain scales by using different filter sizes. Oñoro-Rubio and López-Sastre [8] elaborated a multi-scale regression model, namely Hydra CCNN, which uses the pyramid of image patches as the input. To understand traffic density, Zhang *et al.* [12] separately exploited optimization method based on rank constrained regression (OPT-RC) and deep-learning method based on fully convolution networks with multi-task learning (FCN-MT). By using pre-trained VGG features and dilated convolutions, Li *et al.* [13] presented a novel network for congested scene recognition (CSRNet), which achieves the state-of-the-art performance on the dataset with high resolution. For the UCSD [5] surveillance dataset, however, it performs not well since the resolution of images is small.

3. THE PROPOSED APPROACH

In this section, we describe the proposed architecture, loss function and ground truth generation.

3.1. The Architecture

Inspired by the cascaded CNNs in [7, 14], we propose a novel CNN-based multi-scale density map regressor as shown in Fig. 1, which is composed of a Feature-Shared Network and two regressors (i.e., Coarse-Scale Network and Fine-Scale Network).

3.1.1. Feature-Shared Network

To extract features, we design a straight-forward CNN architecture, namely Feature-Shared Network. The Feature-Shared Network consists of four convolutional layers each followed by a Parametric Rectified Linear Unit (PReLU) as the activation function, and two max-pooling layers with a filter of size 2×2 and stride 2. As we use two pooling layers, the output feature maps are downsampled by a factor of 4. The output features of this network will be shared by the following two regressors: Coarse- and Fine-Scale Networks.

3.1.2. Coarse-Scale Network

In Coarse-Scale Network, coarse density map is estimated by taking the shared features as input. We use three convolutional layers with filters of sizes 5×5, 5×5 and 1×1, respectively. Each of the first two layers is followed by a PReLU, and the last layer is followed by a ReLU to ensure that the density map gets positive values. The output coarse density map has $1/4$ of the original input size. Furthermore, the features of the second convolutional layer will be fed into the subsequent Fine-Scale Network at the next stage. The features supervised by the coarse density map can provide rough locations and densities of objects of the image. In other words, the Coarse-Scale Network can learn a global understanding of the scene.

3.1.3. Fine-Scale Network

From the output features of the Feature-Shared Network, we set up another forward path to generate a fine-scale density map with the same size as the input image, namely Fine-Scale Network. First, two convolutional layers with the filter of size 5×5 are used to extract features. Then, the features of the second layer at this stage are concatenated with features from the earlier stage. After that, two convolutional layers with the filter of size 3×3 are exploited to fuse the concatenated features. To generate a full-resolution density map, we design a simple yet powerful structure: upsampling network. It consists of a

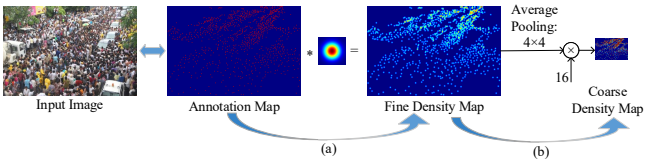

Fig. 2. Generation of fine and coarse density map. The objects are marked as red points in the annotation map.

nearest-neighbor upsampling operation followed by a convolutional layer with a 3×3 kernel. We use two upsampling structures, so the feature maps are upsampled by a factor of 4. At the end of this stage, we employ one more convolutional layer with 1×1 kernel to generate the fine-scale density map. Similar to the first stage, each convolutional layer is followed by a PReLU activation function except for the last layer (with ReLU afterward).

Remark. In contrast to the CNN model in [14] supervised by two tasks that belong to different domains, i.e., classification and density map regression, our model is composed of two tasks in the same domain, i.e., regression of coarse and fine density maps, which can facilitate feature extraction for better performance. Besides, with the multi-scale supervisions, our proposed architecture can avoid the bottleneck of information in this hourglass architecture (See Section 4.2.3).

3.2. Loss function

The loss function of the entire architecture is given by:

$$L = L_f + \lambda L_c \tag{1}$$

where L_c and L_f are the loss of the coarse- and fine-scale density map regressor, respectively, and λ is the wight term to control the balance between two regressors.

We use the smooth L1 loss [4] for each regressor since it is more robust to outliers than the mean squared error (MSE) loss. The loss functions of the two regressors are defined as:

$$L_c = \frac{1}{N}\sum_{i=1}^{N} \text{Smooth}_{L1}\left(F_c(X_i;\Theta_c) - D_i^c\right), \tag{2}$$

$$L_f = \frac{1}{N}\sum_{i=1}^{N} \text{Smooth}_{L1}\left(F_f(X_i;\Theta_f,\theta_c) - D_i^f\right), \tag{3}$$

where N is the number of training images; X_i is the ith image; D^c and D^f are the ground truth coarse and fine density map, respectively; F_c and F_f are the output of Coarse- and Fine-Scale Network, respectively; Θ_c and Θ_f are the parameters of Coarse- and Fine-Scale Network, respectively; θ_c is the features from the second convolutional layer of Coarse-Scale Network. $F_f(X;\Theta_f,\theta_c)$ indicates that the fine density map estimation relies on the features of both Coarse- and Fine-Scale Network.

3.3. Ground truth generation

Object counting datasets provide dot annotations to represent the location of objects in a certain image. We firstly produce an annotation map by setting one at the object locations and zero at the other place. Then, the ground truth density map is generated by blurring the annotation map with a 2D Gaussian kernel G_σ with variance σ, where the kernel is normalized to one. The sum of each pixel value of a density map is equal to the count of objects. The ground truth fine density map for the ith image can be calculated by

$$D_i^f(x) = \sum_{j=1}^{C_i} \delta(x - x_j) * G_\sigma(x), \tag{4}$$

where C_i is the ground truth count corresponding to the ith image; x_j is the jst object location of the ith image; the symbol '$*$' is a convolution operation. The computational process can be seen in Fig. 2 (a).

To obtain the coarse density map, we need to resize the fine density map with the $1/4$ size. It will be generated automatically at the training stage. As shown in Fig. 2 (b) we downsample the fine density map by a factor of 4 using one average pooling layers with a filter of size 4×4 and stride 4, which is followed by a pixel-wise multiplier with a factor of 16. Due to the average pooling, the energy of the next coarse density map is $1/16$ of that of the current fine one. Therefore, we compensate the loss of the energy by multiplying 16 to guarantee the equal energy.

4. EXPERIMENTS

4.1. Implementation Details

Training configuration. We adopt an end-to-end training strategy and train the architecture from scratch. We utilize Adam optimization with the learning rate of 0.00001 to train our model [14]. λ is set to 0.1 because we consider that the fine density map is more important for object counting than the coarse density map. To generate the ground truth density map, we fix σ to 4 for the Gaussian filter.

Data augmentation. For each dataset, we randomly crop 21 patches from each image with $1/4$ of the original input size as training data. With the same augmentation techniques as [13, 14], horizontal flipping is randomly exploited for the training set of each dataset.

4.2. Evaluation and Analysis

4.2.1. UCSD Dataset

The UCSD dataset [5] is taken from a surveillance camera. It contains 2000 frames, sampled by video clips with the size of 158×238 and the framerate of 10. The region of interest (ROI) of images is provided. For a fair comparison, we

Table 1. Comparison on UCSD dataset.

Method	MAE	RMSE
Traditional approaches		
Gaussian process regression [5]	2.24	7.97
Ridge regression [9]	2.25	7.82
Cumulative attribute regression [15]	2.07	7.90
Lempitskt's [10]	1.70	-
Deep learning-based approaches		
Zhang *et al.* [11]	1.60	3.31
CCNN [8]	1.51	-
MCNN [6]	1.07	1.35
FCN-MT [12]	1.67	3.41
CSRNet [13]	1.16	1.47
MSDMR (ours)	**1.04**	**1.33**

Table 2. Comparison on TRANCOS dataset. '-' indicates the method does not provide the results.

Method	MAE	GAME (1)	GAME (2)	GAME (3)
CCNN [8]	10.99	13.75	16.69	19.32
FCN-MT [12]	5.31	-	-	-
CSRNet [13]	3.56	5.49	8.57	15.04
MSDMR (ours)	**2.97**	**4.39**	**6.43**	**9.92**

employ the same configuration with [5]. We train our model by 601st to 1400th frames and test it by the remaining 1200 frames. According to [8, 13], we use ROI to mask the images and density maps.

Following previous works [6, 11], we use Mean Absolute Error (MAE) and Root Mean Square Error (RMSE) as the evaluation metrics. The MAE and RMSE of the proposed method and compared methods are shown in Table 1. The traditional global count regression methods [5, 9, 15, 10] are inferior to deep learning-based density map estimation methods [11, 8, 6, 12, 13]. Our MSDMR obtains the best results compared with state-of-the-art approaches.

4.2.2. *TRANCOS Dataset*

The TRANCOS dataset [8] is captured from traffic surveillance cameras in different road sections. It comprises 1244 images, in which 823 images are used for training and 421 images are used for testing. Each vehicle is manually labeled with a dot. The ROI is also provided for evaluation. The Grid Average Mean Absolute Error (GAME) [8] is used as the evaluation metric for this dataset. The GAME (L) divides an image into 4^L non-overlapping regions and calculate the MAE of each region, and then the overall error is the sum of all MAEs. Note that the GAME (0) is equivalent to the MAE.

Results are shown in Table 2. Our method outperforms state-of-the-art methods for all GAMEs. Compared with [13], the proposed MSDMR decreases the MAE, GAME (1), GAME (2), and GAME (3) by 17%, 20%, 25%, and 34%, respectively. Note that the low GAMEs means the proposed method predicts the count precisely not only for the whole image but also for the sub-regions of the image. Some visual results predicted by the proposed method are shown in Fig. 3.

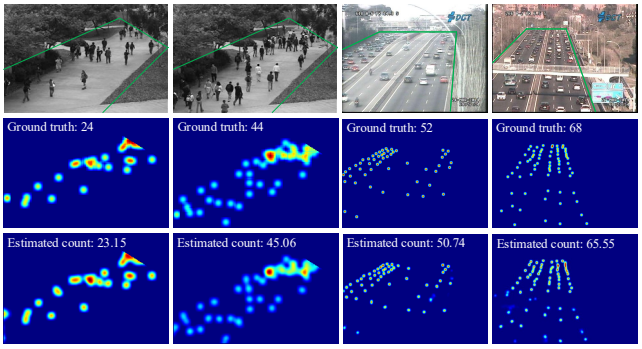

Fig. 3. Qualitative results from the UCSD (first two columns) and TRANCOS (last two columns) dataset. The ROI is inside the green lines. The first row shows the input images. The second row shows the corresponding ground truth density map and count. The third row shows the estimated density map and count of our proposed method.

Table 3. Ablation study. '-' indicates the model diverges.

Dataset	UCSD		TRANCOS	
CDM	×	√	×	√
MAE	1.17	**1.04**	-	**2.97**

Remark. Our model has a compact architecture with 0.372 million parameters compared with the CSRNet with 16.26 million parameters. Our light model can perform well on video surveillance datasets, which is crucial for the embedded application.

4.2.3. *Ablation Study*

Here, we investigate the effectiveness of the coarse density map regressor on the performance. It is composed of two settings: the MSDMR with or without the coarse density map (CDM). The ablation study is conducted on UCSD and TRANCOS dataset. Table 3 shows the results of two settings. These results indicate that the coarse density map regressor can provide extra supervision to improve the performance and even avoid the divergence of the hourglass structure.

5. CONCLUSION

In this work, we propose a novel CNN-based multi-scale density map regressor for object counting in video surveillance. To implement it, we design a compact architecture with the cascaded structure. The proposed architecture first regresses a coarse density map with low resolution, which provides complementary features to further generate a fine density map with high resolution. The fine density map can be integrated to predict an accurate count. Experimental results demonstrate the state-of-the-art performance of our method.

6. REFERENCES

[1] Vishwanath A Sindagi and Vishal M Patel, "A survey of recent advances in cnn-based single image crowd counting and density estimation," *Pattern Recognition Letters*, vol. 107, pp. 3–16, 2018.

[2] Yen-Lin Chen, Bing-Fei Wu, Hao-Yu Huang, and Chung-Jui Fan, "A real-time vision system for nighttime vehicle detection and traffic surveillance," *IEEE Transactions on Industrial Electronics*, vol. 58, no. 5, pp. 2030–2044, 2011.

[3] Mikel Rodriguez, Ivan Laptev, Josef Sivic, and Jean-Yves Audibert, "Density-aware person detection and tracking in crowds," in *International Conference on Computer Vision (ICCV)*. IEEE, 2011, pp. 2423–2430.

[4] Ross Girshick, "Fast r-cnn," in *International Conference on Computer Vision (ICCV)*. IEEE, Dec 2015, pp. 1440–1448.

[5] Antoni B Chan, Zhang-Sheng John Liang, and Nuno Vasconcelos, "Privacy preserving crowd monitoring: Counting people without people models or tracking," in *Computer Vision and Pattern Recognition (CVPR)*. IEEE, 2008, pp. 1–7.

[6] Yingying Zhang, Desen Zhou, Siqin Chen, Shenghua Gao, and Yi Ma, "Single-image crowd counting via multi-column convolutional neural network," in *Computer Vision and Pattern Recognition (CVPR)*. IEEE, 2016, pp. 589–597.

[7] Jifeng Dai, Kaiming He, and Jian Sun, "Instance-aware semantic segmentation via multi-task network cascades," in *Computer Vision and Pattern Recognition (CVPR)*. IEEE, 2016, pp. 3150–3158.

[8] Daniel Oñoro-Rubio and Roberto J López-Sastre, "Towards perspective-free object counting with deep learning," in *European Conference on Computer Vision (ECCV)*. Springer, 2016, pp. 615–629.

[9] Ke Chen, Chen Change Loy, Shaogang Gong, and Tony Xiang, "Feature mining for localised crowd counting," in *British Machine Vision Conference (BMVC)*, 2012, vol. 1, p. 3.

[10] Victor Lempitsky and Andrew Zisserman, "Learning to count objects in images," in *Neural Information Processing Systems Conference (NIPS)*, 2010, pp. 1324–1332.

[11] Cong Zhang, Hongsheng Li, Xiaogang Wang, and Xiaokang Yang, "Cross-scene crowd counting via deep convolutional neural networks," in *Computer Vision and Pattern Recognition (CVPR)*. IEEE, 2015, pp. 833–841.

[12] Shanghang Zhang, Guanhang Wu, Joao P Costeira, and José MF Moura, "Understanding traffic density from large-scale web camera data," in *Computer Vision and Pattern Recognition (CVPR)*. IEEE, 2017, pp. 4264–4273.

[13] Yuhong Li, Xiaofan Zhang, and Deming Chen, "Csrnet: Dilated convolutional neural networks for understanding the highly congested scenes," in *Computer Vision and Pattern Recognition (CVPR)*. IEEE, 2018, pp. 1091–1100.

[14] Vishwanath A Sindagi and Vishal M Patel, "Cnn-based cascaded multi-task learning of high-level prior and density estimation for crowd counting," in *Advanced Video and Signal Based Surveillance (AVSS)*. IEEE, 2017, pp. 1–6.

[15] Ke Chen, Shaogang Gong, Tao Xiang, and Chen Change Loy, "Cumulative attribute space for age and crowd density estimation," in *Computer Vision and Pattern Recognition (CVPR)*. IEEE, June 2013, pp. 2467–2474.