CROWD COUNTING WITH FULLY CONVOLUTIONAL NEURAL NETWORK

Ming Liu [1] *Jue Jiang*[2] *Zhenwei Guo*[3] *Zenan Wang*[4] *Yang Liu*[1]

[1] JD Finance, Beijing, China.
[2] Memorial sloan kettering cancer center, medical physics, New York City, NY, United States.
[3]School of Geosciences and Info-physics, Central South University, Changsha, 410083, China.
[4]Department of Gastroenterology, Beijing Chao-Yang Hospital,Beijing,China.

ABSTRACT

Crowd counting estimation is an extremely challenging task due to various crowded scenarios. In this paper, we present a deep learning framework for crowd counting from a single static image with different number of people and arbitrary perspective. In the design of convolutional neural network structure, we employ the VGG16 model but drop the fully connected layers. Meanwhile, high-level features are combined with low-level features through laterally connected feature pyramid network by element-wise addition to ensure higher resolution and more context information. Extensive experiments are conducted on ShanghaiTech and UCF_CC_50 datasets. The results show that our model achieves the lowest mean absolute error (MAE) and comparable mean square error (MSE), and outperforms the current state-of-the-art methods.

Index Terms— Crowd counting, convolutional neural network, feature pyramid network, deep learning

1. INTRODUCTION

Crowd counting, which plays an indispensible role in public management, has drawn a lot of attention since it has the potential application in crowd monitoring and scene understanding [1]. Many solutions of computer vision tasks have been applied into the field of real-time density estimation for crowd counting in different scenarios, such as traffic, shopping malls, airports, and popular tourist attractions. Numerous challenges, for instance, occlusions, non-uniform distribution, high clutter, intra-scene and inter-scene variations in appearance, make the problem very difficult to solve. Traditionally, the solutions can be classified into detection based approach [2], regression based approach [3] and density estimation based approach [4]. However, these traditional methods are usually inefficient for crowd counting task.

In recent years, convolutional neural network (CNN) based approaches such as VGG network [5] have achieved remarkable success in a variety of computer vision areas. Compared to previous hand-crafted feature based methods, the CNN-based approaches [6, 7, 8, 9, 10, 11, 12] have been demonstrated significant improvement in crowd counting related problems. CNNs approaches were primarily introduced by Wang et al. [13] to estimate crowd density through an end-to-end deep CNN regression model. Zhang et al. [12] proposed to address the scene shift problem by finetuning the pre-trained neural network when applyed to a different target scene for crowd counting. Boominathan et al. [7] predicted the density map for a given crowd image by using combination of deep and shallow fully convolutional networks to capture semantic information. Han et al. [8] extracted the features by a pre-trained residual network with a fully connected network for count regression and a Markov Random Field (MRF) to smooth the counting results of the local patches. All of the above works need to divide the testing image into several patches and predict separately with an average of overlapping regions during the inference process. In contrast to the patch-based inference, Shang et al. [14] proposed an end-to-end crowd counting estimation by CNN, which leaded to a reduction of complexity. Zhang et al. [6] proposed a full image-based inference method with a multi-column CNN architecture for crowd density estimation. They also investigated the transfer learning with different datasets. Sindagi et al. [10] proposed a novel end-to-end cascaded network of CNNs to jointly learn crowd count classification and density map estimation through a multi-task network, which incorporated high-level priors and a multi-scale CNN

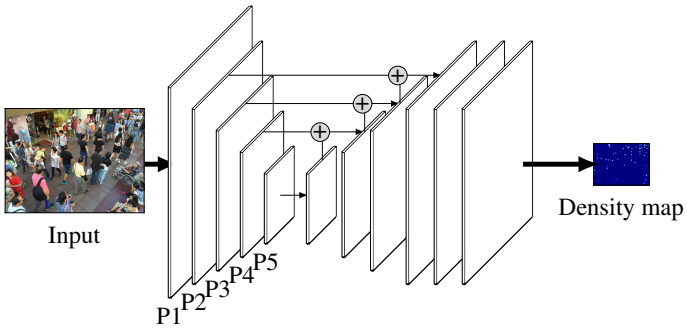


Fig. 1. Structure of proposed convolutional neural network.

network.

However, none of the above works use the mutil-layer features. We argue that features from all levels are useful for crowd counting. How to exploit mutil-layer features effectively remains an open question and deserves more attention. To this end, we propose a novel network architecture that effectively exploits multi-level features by feature pyramid network (FPN) [15] for accurately estimating the crowd count.

2. PROPOSED METHOD

2.1. Network Architecture

The proposed architecture is composed of a encoder structure, aiming at extracting features, and a decoder, aiming at generating density map. In the task of crowd counting, the problem lies in the high dense people with very small objects. However, in a deep ConvNet, high-level features carries rich semantic information, but relatively fewer information about those small objects. On the other hand, the lower-level features has the information about small objects but fewer information about the context. We assume that the middle-level features also assist the task of crowd counting with compensation for both high-level and low-level features. Thus, inspired by FPN, we exploit the utilization of middle-level features by propagating high-level features from top to bottom through deconvolution to integrate rich context information features and low-level features. The model takes a sinlge-scale image of an arbitrary size as input, and outputs proportionally sized feature map, in a fully convolutional fashion. To be specific, top-down pathway and laterally connected FPN with VGG16 [5] backbone are used to combine the high and low level features. In more detail, VGG16 contains five CNN blocks, each CNN block followed by a max pooling layer to reduce the spatial size to 1/2. We denote this five max pooling layers as P1, P2, P3, P4, P5 individually. The proposed model upsamples each max pooling layer from P5 to P3 and apply element-wise addition with P4 to P2 separately. This leads to a 1/4 times smaller output size of the input image. Fig.1 illustrates the network architechture.

We follow the convolution-activation patten in the VGG16 [5] model but batch normalization is not involved. The 1x1 kernel is used to adjust the number of channels for lateral connection and the final layer to predict density map. Besides, all the other convolutional layers have a 3x3 kernel with stride 1. All the deconvolutions have a 4x4 kernel with stride 2. A convolutional layer with 3x3 kernel is followed after lateral and top-down element-wise addition. All the number of channels in the predicting network is 256, recitified linear unit(Relu) is adopted as the activation function. The pixel-wise mean square error(MSE) loss is used as the objective function to measure the difference between the estimated density map and ground truth, which is defined as follow:

$$Loss = \frac{1}{N}\sum_{i=1}^{N} \|F(X_i, \Theta) - D_i\|_2^2 \tag{1}$$

Where N is the number of images, $F(X_i, \Theta)$ is the esitmated density map, D_i is the ground truth. The estimated number of people C_i can be computed as below:

$$C_i = \int F(X_i, \Theta)dx \tag{2}$$

2.2. Implementation

There are two challenges for training our crowd counting model. One is short of data, all the existing public datasets usually contain a small amount of images, UCF_CC_50 contains 50 images, ShanghaiTech part A and part B contain 1198 images. Especially the number of people in one single image may vary from hundreds to thousands, this raises the difficulty level. The other problem is that different datasets contain different types of images, either RGB or gray scale, they have different channels and need different treatment in one framework. To overcome these issues, we firstly enlarge the training set by random crop 200 patches from every image with fixed size 224x224, and apply the random flip by horizontal direction during the training process. Secondly, we convert all the images to gray scale during training process.

The ground truth supplied by all the public datasets is only the positions of people head in the image. It indicates that the positive samples are only thousands of pixels in a image, the negative samples are millions of pixels. The number of the negative samples is thousands of times of the positive samples, which makes the data extremely imbalanced. This original ground truth would lead the model to a very wrong direction. To overcome this problem, Zhang et al. [6] applied an Gaussian blurring with adaptive kernels to generate ground truth, Sindagi et al. [10] and Boominathan et al. [7] simply blurred each head annotation with one Gaussian kernel normalized to sum to one, which remained the same counts per image and included more pixels for CNN model training. We employed the second way to generate the ground truth with an Gaussian blurring, which was defined as below:

$$D(x_i) = \sum \mathcal{N}(x_i - \hat{x}_i, \sigma) \tag{3}$$

where D is the ground truth density map, $\hat{x}$ corresponding to the annotations, σ is the scale parameter of the 2D Gaussian kernel, i is the patch index.

We trained and evaluated the network on a Nvidia Tesla K40m GPU processor with 11 GB memory. Adam optimization with a learning rate of 0.00001 and momentum of 0.9 were adopted to train the model.

	Part A		Part B	
Method	**MAE**	**MSE**	**MAE**	**MSE**
C. Zhang et al. [12]	181.8	277.7	32.0	49.8
MCNN[6]	110.2	173.2	26.4	41.3
LBP+RR	303.2	371.0	59.1	81.7
Sindagi et al.[10]	101.3	152.4	20.0	31.1
Han et al. [8]	79.1	130.1	17.8	26.0
Proposed method	**67.6**	**110.6**	**10.1**	**18.8**

Table 1. The comparison of results: Estimation errors on the ShanghaiTech dataset.

3. EXPERIMENT AND RESULTS

For the purpose of evaluation, we follow the standard metrics used by many existing methods for crowd counting. These metrics are defined as follows:

$$MAE = \frac{1}{N}\sum_{i=1}^{N}|y_i - y_i^{'}| \quad (4)$$

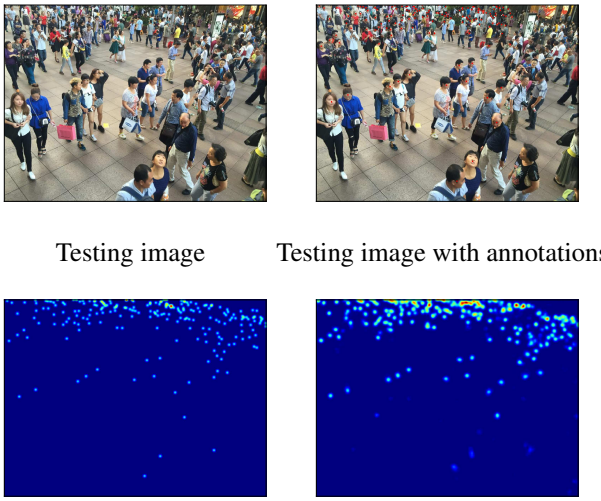
$$MSE = \sqrt{\frac{1}{N}\sum_{i=1}^{N}|y_i - y_i|^2} \quad (5)$$

Where MAE stands for mean absolute error, MSE stands for mean squared error. N is the number of testing images, y_i is ground truth count and $y_i^{'}$ is estimated count.

3.1. ShanghaiTech dataset

The ShanghaiTech dataset was introduced by Zhang et al. [6]. It contains 1198 annotated images with a total number of 330,165 people. This dataset consists of two parts: Part A with 482 images that are randomly crawled from the Internet, and Part B with 716 images which are taken from busy streets of metropolitan areas in Shanghai. The crowd density varies significantly, very dense crowd density in Part A and relatively sparse crowd density in Part B. Both parts were further divided into training and testing sets. Training set in Part A contains 300 images and that in Part B contains 400 images. Testing set in part A contains 182 images and that in part B contains 316 images.

We compared our proposed method with five recent approaches, C. Zhang et al. [12], MCNN by Y. Zhang et al. [6], Sindagi et al. [10], Han et al. [8] and Local Binary Pattern (LBP) [6]. Except for LBP, the others are all CNN based methods. The comparison of the results was listed in Table 1, which demonstrated our proposed method achieved the lowest MAE and MSE values and outperformed all the other methods by a large margin. Fig.2 showed examples of the ground truth density map and estimated density map of image in Part B.



Testing image | Testing image with annotations

Ground truth, count:216 | Estimation, count:216

Fig. 2. The ground truth and estimated density map of proposed model on ShanghaiTech Part B.

3.2. UCF_CC_50 dataset

The UCF_CC_50 dataset [9] is an extreamly challenging dataset for not only the limited number of images but also the dramatically changed number of people from image to image. This dataset contains 50 highly densed images with different spatial resolutions in different scenarios. The number of people in the image varies from 94 to 4543 with an average of 1280 individuals per image, and a total 63974 annotations were provided.

A standard 5-fold cross-validation [9] was performed to evaluate the proposed method. We compared our model with six previous methods, the comparison of different methods was listed in Table 2. The work of C. Zhang et al. [12], Y. Zhang et al. [6], Han et al. [8], Xiong et al. [11] are all CNN based models. Lempitsky et al. [16] used the Fourier analysis and texture features (SIFT) extracted from a image to learn a density map. Idress et al. [9] estimated the crowd count by multi-source features. From Table 2, we can conclude that our proposed model archieved the lowest MAE and comparable MSE among all the methods. Some testing examples with the associated ground truth and esimated density map were shown in Fig.3.

3.3. Model Variation

The original VGG16 network contains five different CNN blocks. After five times max-pooling the image size reduces 32 times. We investigated two different approaches to assemble the model. We denoted the proposed model as 'P5-4x', the other approach denoted by 'P4-4x' upsampled from P4 to P2 and were added with P3 to P1. The output was also 4

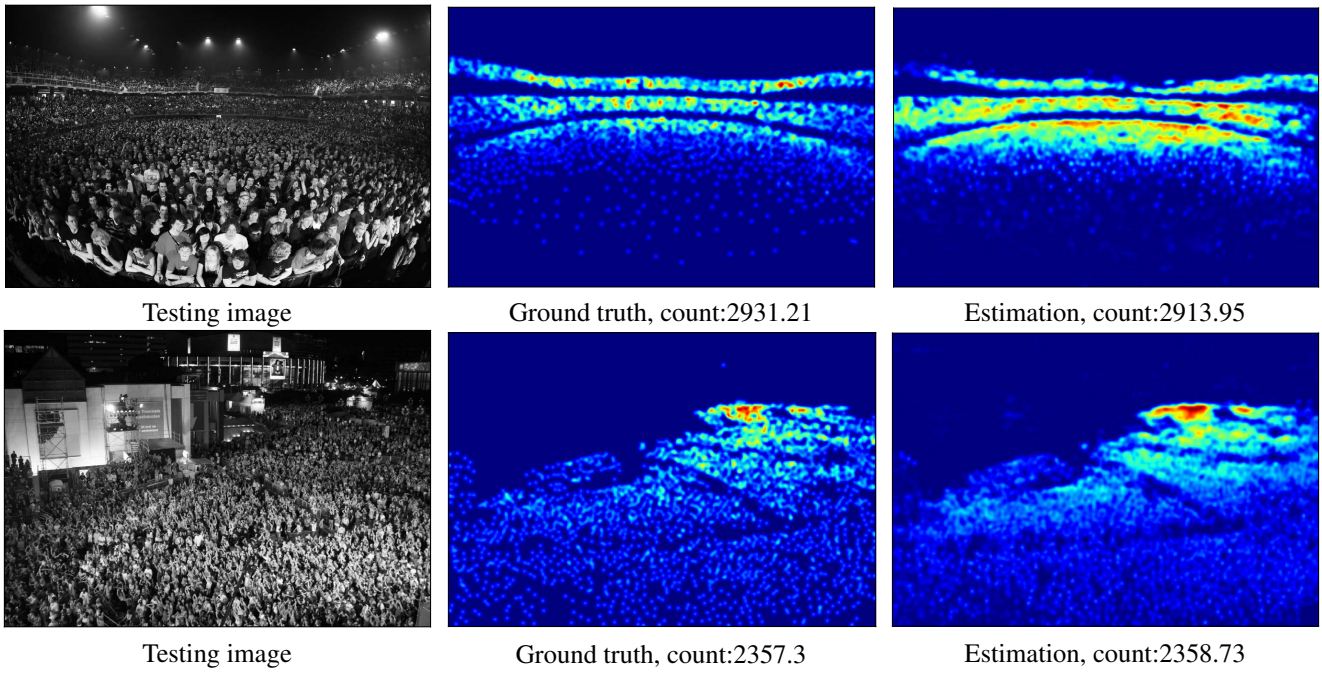


Fig. 3. The ground truth and estimated density map of proposed convolutional neural network model on UCF_CC_50 dataset.

Method	MAE	MSE
Lempitsky et al. [16]	493.4	487.1
Idrees et al. [9]	419.5	541.6
C. Zhang et al. [12]	467.0	498.5
MCNN [6]	377.6	509.1
Han et al. [8]	254.1	352.5
ConvLSTM-nt [11]	284.5	**297.1**
Proposed method	**253.1**	356.4

Table 2. The comparsion of results: Estimation errors on the UCCF_CC_50 dataset.

times smaller than the input image. We compared these two models on both ShanghaiTech and UCF_CC_50 datasets, results were listed in Table 3. P4-4x had much better results on ShanghaiTech Part B and P5-4x performed much better on the others. Crowd density in ShanghaiTech Part B is relatively sparse and that in Part A and UCF_CC_50 is extremely dense. The results supported the observtion that deeper network performed better in more crowded scenes while shallow network could gain good results in less crowded scenarios.

4. CONCLUSION

In this paper, we presented a deep learning framework for estimating crowd counting from a single static image with various number of people and arbitrary perspective. With the combination of high-level and low-level features in one network architecture, the proposed method achieved lower estimation errors compared to the other methods on both ShanghaiTech and UCF_CC_50 datasets. Additionally, the results demonstrated that the proposed method work very well in both relatively sparse and extremely dense scenes. Compared to the other CNN-based methods, our method was very efficient and provided effective results with less errors through one single network without any post-processing.

	Part A		Part B		UCF	
Method	**MAE**	**MSE**	**MAE**	**MSE**	**MAE**	**MSE**
P4-4x	72.8	119	**9.1**	**15.2**	298	400.5
P5-4x	**67.6**	**110.6**	10.17	18.8	**253.1**	**356.4**

Table 3. The Comparsion of results on both ShanghaiTech and UCF_CC_50 datasets for Proposed P5-4x and P4-4x models.

5. 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*, 2017.

[2] Navneet Dalal and Bill Triggs, "Histograms of oriented gradients for human detection," in *Computer Vision and Pattern Recognition, 2005. CVPR 2005. IEEE Computer Society Conference on*. IEEE, 2005, vol. 1, pp. 886–893.

[3] Antoni B Chan and Nuno Vasconcelos, "Bayesian poisson regression for crowd counting," in *Computer Vision, 2009 IEEE 12th International Conference on*. IEEE, 2009, pp. 545–551.

[4] Bolei Xu and Guoping Qiu, "Crowd density estimation based on rich features and random projection forest," in *Applications of Computer Vision (WACV), 2016 IEEE Winter Conference on*. IEEE, 2016, pp. 1–8.

[5] Karen Simonyan and Andrew Zisserman, "Very deep convolutional networks for large-scale image recognition," *arXiv preprint arXiv:1409.1556*, 2014.

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

[7] Lokesh Boominathan, Srinivas SS Kruthiventi, and R Venkatesh Babu, "Crowdnet: A deep convolutional network for dense crowd counting," in *Proceedings of the 2016 ACM on Multimedia Conference*. ACM, 2016, pp. 640–644.

[8] Kang Han, Wanggen Wan, Haiyan Yao, and Li Hou, "Image crowd counting using convolutional neural network and markov random field," *arXiv preprint arXiv:1706.03686*, 2017.

[9] Haroon Idrees, Imran Saleemi, Cody Seibert, and Mubarak Shah, "Multi-source multi-scale counting in extremely dense crowd images," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2013, pp. 2547–2554.

[10] 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), 2017 14th IEEE International Conference on*. IEEE, 2017, pp. 1–6.

[11] Feng Xiong, Xingjian Shi, and Dit-Yan Yeung, "Spatiotemporal modeling for crowd counting in videos," *arXiv preprint arXiv:1707.07890*, 2017.

[12] Cong Zhang, Hongsheng Li, Xiaogang Wang, and Xiaokang Yang, "Cross-scene crowd counting via deep convolutional neural networks," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2015, pp. 833–841.

[13] Chuan Wang, Hua Zhang, Liang Yang, Si Liu, and Xiaochun Cao, "Deep people counting in extremely dense crowds," in *Proceedings of the 23rd ACM international conference on Multimedia*. ACM, 2015, pp. 1299–1302.

[14] Chong Shang, Haizhou Ai, and Bo Bai, "End-to-end crowd counting via joint learning local and global count," in *Image Processing (ICIP), 2016 IEEE International Conference on*. IEEE, 2016, pp. 1215–1219.

[15] Tsung-Yi Lin, Piotr Dollár, Ross Girshick, Kaiming He, Bharath Hariharan, and Serge Belongie, "Feature pyramid networks for object detection," *arXiv preprint arXiv:1612.03144*, 2016.

[16] Victor Lempitsky and Andrew Zisserman, "Learning to count objects in images," in *Advances in Neural Information Processing Systems*, 2010, pp. 1324–1332.