Research Article

Liping Zhu*, Hong Zhang, Sikandar Ali, Baoli Yang, and Chengyang Li

Crowd counting via Multi-Scale Adversarial Convolutional Neural Networks

https://doi.org/10.1515/jisys-2019-0157
Received Jun 10, 2019; accepted Jan 10, 2020

Abstract: The purpose of crowd counting is to estimate the number of pedestrians in crowd images. Crowd counting or density estimation is an extremely challenging task in computer vision, due to large scale variations and dense scene. Current methods solve these issues by compounding multi-scale Convolutional Neural Network with different receptive fields. In this paper, a novel end-to-end architecture based on Multi-Scale Adversarial Convolutional Neural Network (MSA-CNN) is proposed to generate crowd density and estimate the amount of crowd. Firstly, a multi-scale network is used to extract the globally relevant features in the crowd image, and then fractionally-strided convolutional layers are designed for up-sampling the output to recover the loss of crucial details caused by the earlier max pooling layers. An adversarial loss is directly employed to shrink the estimated value into the realistic subspace to reduce the blurring effect of density estimation. Joint training is performed in an end-to-end fashion using a combination of Adversarial loss and Euclidean loss. The two losses are integrated via a joint training scheme to improve density estimation performance. We conduct some extensive experiments on available datasets to show the significant improvements and supremacy of the proposed approach over the available state-of-the-art approaches.

Keywords: Crowd counting, Multi-Scale, Crowd density estimation, Density map

2010 Mathematics Subject Classification: 68T45

1 Introduction

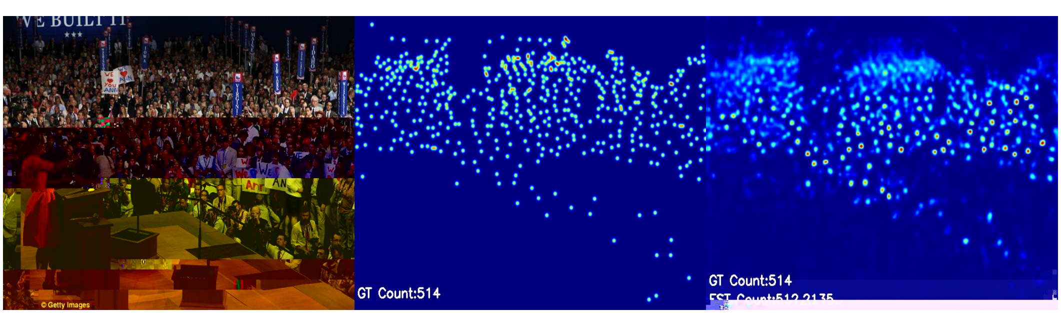
With the rapid growth in the urban population, public safety issues have become the focus of attention in video surveillance. In a real-time analysis of crowds such as public gatherings and sports events, it is necessary to estimate the number and density map of the population. In recent years, crowd analysis has attracted many researchers. Not only it can be applied to urban planning [1], scene understanding [2], and traffic monitoring, but also to the counting tasks of other domains, such as counting cells under the microscope [3–6], vehicle counts [7–11]. However, due to the presence of various complexities, such as complex illumination, pedestrian occlusion in a dense scene, perspective distortion and non-uniform distribution of people, it is a challenging task in computer vision and these issues result in an accuracy of estimation that is far from optimal.

***Corresponding Author: Liping Zhu:** Beijing Key Lab of Petroleum Data Mining, China University of Petroleum, Beijing, 10224, China; Email: 675989420@qq.com
Hong Zhang: Beijing Key Lab of Petroleum Data Mining, China University of Petroleum, Beijing, 10224, China; Email: cup_zh@163.com
Sikandar Ali: Beijing Key Lab of Petroleum Data Mining, China University of Petroleum, Beijing, 10224, China; Email: hqsikandar@qq.com
Baoli Yang: Beijing Key Lab of Petroleum Data Mining, China University of Petroleum, Beijing, 10224, China; Email: yangbaoli111111@qq.com
Chengyang Li: Beijing Key Lab of Petroleum Data Mining, China University of Petroleum, Beijing, 10224, China; Email: 623610394@qq.com

Some earlier methods of crowd counting considered it as a computer vision problem, counting the number of pedestrians by detecting and tracking, and then training a detector to detect the number of pedestrians appearing in the crowd image. However, if the crowd is very dense, the occlusion between pedestrians is more serious, which may result in poor detection. Moreover these methods are based on the traditional hand-featured regression, achieving better performance than detection through regressing the number of pedestrians on the image. Additionally, this method uses manual features like HOG [12], and therefore difficult to achieve the best result due to insufficient expression of local features, angle, and large-scale variation of the crowd image. Inspired by a recent successful solution of multiple computer vision tasks with convolutional neural network (CNN), many CNN-based methods [13–15] were developed to solve these issues and obtained remarkable success. For instance [15–17] used a multi-channel CNN structure to emphasize the scale variation and achieved good results in the crowd density estimation, using different sizes of convolutional kernels to deal with different sizes of a head in input images and try to solve the head scale variation. In the crowd density map, each marking point represents the location of a pedestrian, and the number of crowd is obtained by pixel integration in the density map. Current CNN-based methods [8, 18, 19] use multi-path convolutional neural network, and Euclidean loss is used as an objective function to optimize model, each sub-network uses different convolutional kernel sizes to extract multi-scale features. Local optimization is achieved by minimizing Euclidean loss, and finally fine-tuned all sub-network by joint training.

To solve these issues based on the multi-column CNN [19] which has a success of working in the crowd counting, a new crowd counting framework called Multi-Scale Adversarial Convolutional Neural Network (MSA-CNN) is proposed. The multi-column is used to extract high-dimensional features of the crowd image, and then a series of fractionally-strided convolutional layers process to restore the detail of features caused by max-pooling layers, so that to obtain a high-resolution density map. In addition, inspired by Generative Adversarial Network (GAN) in successful image interpretation [20], we propose the adversarial training method to reduce the blurring effect and improve the quality of the density map. Figure 1 shows the result of our method on one sample. In this paper, our main contributions are summarized as follows:

1. We proposed a novel parameter-optimized MSA-CNN to solve crowd counting and density estimation issues.
2. After extracting the high-level image features of the crowd, several fractionally-strided convolutional layers are used to restore some details of the image caused by the previous max-pooling, therefore improving the quality of the estimated density map, and ultimately improving the accuracy.
3. We conduct extensive experiments on the two representative datasets [9, 12] and compared the outcomes with existing methods. Our method was proved superior to the current state-of-the-art performance.



(a) (b) (c)

Figure 1: The proposed method results, (a) input image (the part_A from ShanghaiTech dataset), (b) ground density map, (c) estimated density map via our proposed method.

2 Related works

Current crowd density estimation methods are broadly divided into: 1) detection-based methods, 2) regression methods based on hand-crafted features, and 3) CNN-based methods. These are briefly explained as follows:

Detection-Based methods:
The initial adoption of a single-person-based framework considers the population as a single entity group to estimate the number of pedestrians [7, 9, 12, 21, 22], and none of these methods are applied for a single still image. Since early related research simply focused on video surveillance scenarios to fully explore the information of motion and appearance. For instance, [12] trained dynamic detectors pass two consecutive segments of a video sequence frames to capture this information, and then the recurrent neural network framework has been used for head detection in the crowd scene. [23] use GoogLeNet's deep functionality in the Long Short-Term Memory (LSTM) framework to return the bounding box of the head. [4, 5] proposed a trajectory clustering method based on tracking visual features to finish crowd counting in video surveillance, but this method also cannot estimate the number of people in a single static image. Moreover, the detection and tracking method seriously affects the performance of the estimated population when the crowd is very dense and the image prone to occlusion.

Regression-Based methods:
the most widely used methods for crowd counting is feature-based regression [12–14, 24], which regressed the scalar values (number of people) or density maps [3, 24]. The main steps of the method are divided into: (1) extracting the foreground; (2) extracting various features of the foreground, such as the area of the crowd [3, 12, 13, 16], the edge information [3, 12, 14, 25], or texture information [3, 6], and (3) estimating the number of persons with a regression function. The linear [1] or piece-wise linear [15] function is a relatively simple model and exhibits good performance. Other more effective methods are Ridge Regression (RR) [3], Gaussian Process Regression (GPR) [13] and Neural Network (NN) [26], these methods are suitable for crowd counting algorithms of monitoring videos, due to foreground segmentation. It is very difficult task and the performance of the algorithm is largely affected by it. There are also some works for crowd counting of still images, [8] suggested making use of multi-source information to estimate the number of people in a single image. [27] estimated counts by combining information from multiple sources, such as point of interest (SIFT) [28], fourier analysis, wavelet decomposition, Gray-Level Co-occurrence Matrix (GLCM) features, and low confidence head detections. [17] trained a support vector machine (SVM) with features extracted from a pre-trained model, and then estimated the number of people in a single still image. The regression-based methods are better than the detection methods, this method can only extract low-level features, so it is also not the best way to map features to the number of pedestrians.

CNN-Based method:
Recent CNN-based methods are also a kind of regression methods. It is introduced separately because it is different from the traditional regression methods which are based on traditional hand-crafted features. It is possible to extract high-dimensional features of the crowd images by the convolutional operation. [15] proposed a CNN-based method for crowd counting in different scenes, and then fine-tuned the pre-trained network based on foreground information when passing a test data, this method achieves good performance on the most of existing datasets, but their train and test datasets require foreground maps, while in crowd counting applications, there are no foreground maps available. In [14, 19], a multi-column network structure is used to deal with the scale change problem. Using traditional CNN, each column is separately trained; the obtained three models are merged and then fine-tuned them. The fully connected layer uses a 1 × 1 convolution kernel to fuse the feature maps from a particular scale of training and regress a density map. Inspired by

fore it will tend to blur map on edges and outliers of the image [26]. However, when using adversarial loss, it will judge whether a pixel is “real” or “fake”, by optimizing loss function to encourage the “fake” have the same as “real” pixel distribution. In principle, it is possible to prompt a clear image and avoid blur as well, so it is impossible to generate blurred images [31]. But if we simply use the adversarial loss as objective function may cause exceptions in the spatial structure and even it exists outliers in the input label space. So we refer to the previous work [20, 32, 33] and further add a conventional loss to improve the solution. The following sub-sections discuss the details of the objective function formula.

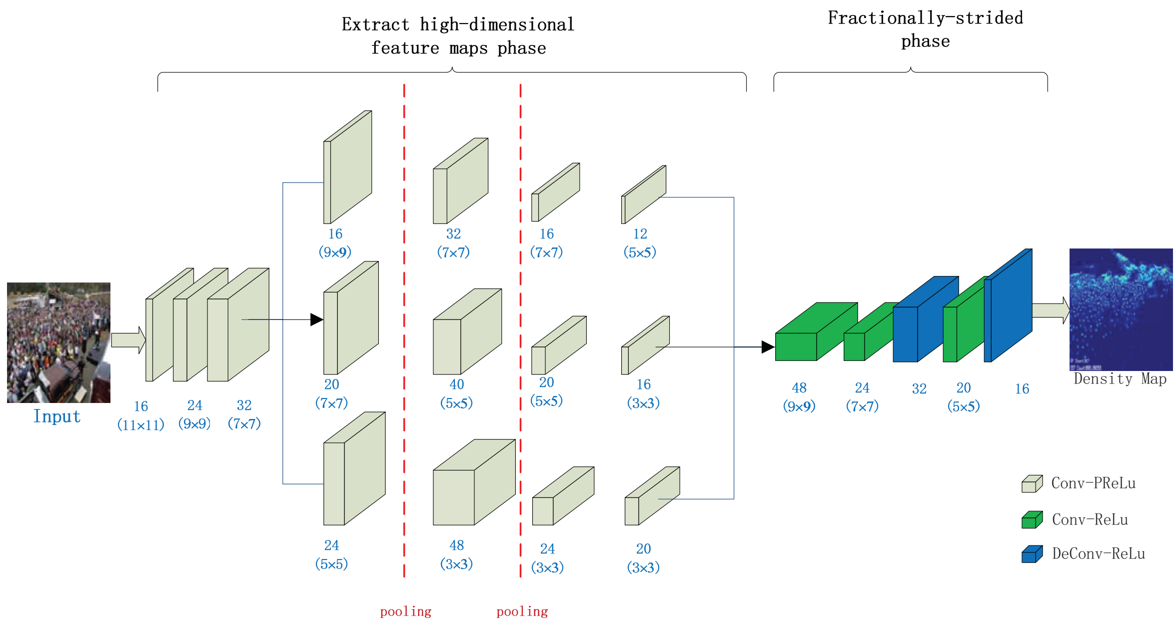


Figure 2: Generator stage: the first part is used to extract high-dimensional feature map, which is basically composed of convolutional layer-PReLU (Conv-PReLU), pooling represents max-pooling layer, factor is 2; then the second is fractionally-strided convolutional phase, its basic composition is deconvolution-ReLU (DeConv-ReLU).

Table 1: The detail of parameters setting.

layer	parameter	layer	parameter
conv1	11×11×16 conv,padding 5	conv2-3	5×5×20 conv,padding 2
conv2	9×9×24 conv,padding 4	conv2-4	3×3×16 conv,padding 1
conv3	7×7×32 conv,padding 3	conv3-1	5×5×10 conv,padding 2
conv1-1	9×9×16 conv,padding 5	pool3-1	3×3×24 conv,padding 1
pool1-1	2×2 max-pooling,stride 2	conv3-2	3×3×48 conv,padding 1
conv1-2	7×7×32 conv,padding 3	pool3-2	2×2 max-pooling,stride 2
pool1-2	2×2 max-pooling,stride 2	conv3-3	3×3×24 conv,padding 1
conv1-3	7×7×16 conv,padding 3	conv3-4	3×3×20 conv,padding 1
conv1-4	5×5×12 conv,padding 2	conv4	9×9×48 conv,padding 4
conv2	7×7×20 conv,padding 3	conv5	7×7×24 conv,padding 3
pool2-1	2×2 max-pooling,stride 2	decv6	2×2×32 deconv,stride 2
conv2-2	5×5×40 conv,padding 2	conv7	5×5×20 conv,padding 2
pool2-2	2×2 max-pooling,stride 2	decv8	2×2×16 deconv,stride 2

Algorithm 1 The training process of estimating density map for our method

Input: N training image patches $\{X_i\}_{i=1}^{N}$ with ground truth density maps $\{P_i^{GT}\}_{i=1}^{N}$, and the size of each ground truth density map is $\frac{1}{4}$ of original image
Output: Trained Generator network parameters Θ_G which includes Θ_{G1} and Θ_{G2}
Initialize Θ_{G1} with random Gaussian weights
Pre-training the first stage of generator network for T_d epochs
for $t = 1$ to T_d do **do**
  for $i = 1$ to N do **do**
    $\grave{}_i^{G1}=argminL_E$
    update Θ_{G1} by stochastic gradient descent
  end for
end for
/*Fine-tuning the first generator network parameters and Training for T_c epochs*/
Initialize parameters of discriminator network as Θ_D and Fractionally-strided phase as Θ_{G2} with random Gaussian weights
for $i = 1$ to T_c do **do**
  for $i = 1$ to N do **do**
    $\grave{}_i^{I}=argminL_I$
    update Θ_D, Θ_{G2} and fine-tuning Θ_{G1}
  end for
end for

3.2 Objective function

It has been widely acknowledged that Euclidean loss has certain disadvantages [34] such as sensitivity to outliers and image blur. Motivated by GAN in image reconstruction and these observations, a combined scheme of Euclidean loss and weighted adversarial loss as the final loss function for solving the issue of L2-minimization was incorporated [20]. The objective function is as follow:

- **Euclidean loss**

$$L_E = \frac{1}{N}\sum_{i=1}^{N} ||G_{\theta G}(X_i, \Theta) - P_i^{GT}||_2 \tag{1}$$

Where N is the number of training samples, X_i is the i^{th} training sample, Θ representing the network parameters, $G_{\theta G}(X_i, \Theta)$ indicates the density maps and are estimated by the network, P_i^{GT} representing the i^{th} ground true density map.

- **Adversarial loss**

$$L_A = -\log(D_{\theta D}(G_{\theta G}(I))) \tag{2}$$

Where $G_{\theta G}$ and $D_{\theta D}$ are the outputs of the Generator and Discriminator network structures respectively, L_A representing the adversarial loss function. I indicates the input crowd image.

- **Final objective**

$$L_I = L_E + \lambda L_A \tag{3}$$

In this formula, λ indicates the weight multiple that connects the two functions. We set the value is 10^{-3}, L_A is the Adversarial loss function, while L_E is Euclidean loss.

3.3 Training and Implementation Details

In training and testing phase, the ground truth density map data is necessary. The original data provides the crowd image and the corresponding annotated head position, so we only need to convert the available point

4.1 Experiment on ShanghaiTech dataset

The ShanghaiTech dataset was created by [19]. The dataset includes 1,198 annotated indoor and streetscape images with a total of 33015 pedestrians, as well as crowd images at different angles, and consists of two parts: 482 images in Part_A and 716 images in Part_B. The two parts of the dataset are further divided into a train set and a test set. The train set of Part_A and Part_B are 300 and 400 images respectively, the rest of the images are used as test dataset. The proposed method is compared with the recent five best methods: [15], MCNN [19], Switching-CNN [18], Cascaded-MTL [35], and CP-CNN [30] on ShanghaiTech datasets. Comparative results are shown in Table 2. [15] proposed two learning objectives for crowd counting and density estimation. Further, they learned the network by alternately training two objective functions. [19] used a multi-column CNN to solve the multi-scale difference issue on crowd images and proposed a density map generation method. [18] proposed a switched CNN classifier, it can select the suitable network branch to solve the problem of large-scale and perspective variation, and at the same time improve the accuracy of crowd estimation. [35] proposed a multi-task cascade CNN that utilizes a high-level prior to learn crowd count classification and density map estimation tasks. In [30] the author extracted global and local context information of the image to generate a high-quality density map and lower estimation error. It can be seen from Table 2 that result of MSA-CNN com-

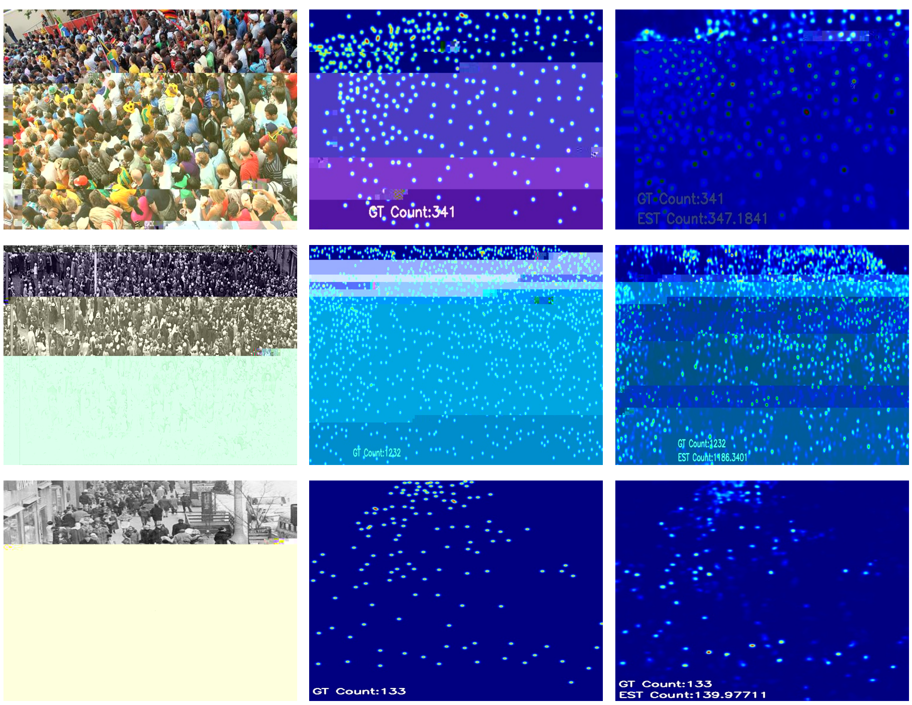


Figure 4: The density map estimated by MSA-CNN on the Shanghai Tech Part_B dataset, the first column is test images, the second column is ground truth density map, and the third column is the estimated density map by our approach(MSA-CNN).

Table 2: Comparison results on ShanghaiTech dataset.

	Part_A		Part_B	
Method	MAE	MSE	MAE	MSE
[15]	181.8	277.7	32.0	49.8
MCNN [19]	110.2	173.2	26.4	41.3
Cascaded-MTL [35]	101.3	152.4	20.0	31.1
Switching-CNN [18]	90.4	135.0	21.6	33.4
CP-CNN [30]	73.6	106.4	20.1	30.1
MSA-CNN (ours)	72.4	104.7	22.7	35.4

pared with other methods on this dataset. Figs. 4 and 5 illustrate some samples of the SahanghaiTech dataset. These samples are predicted by MSA-CNN along with the ground truth, our proposed method achieves lower count error.

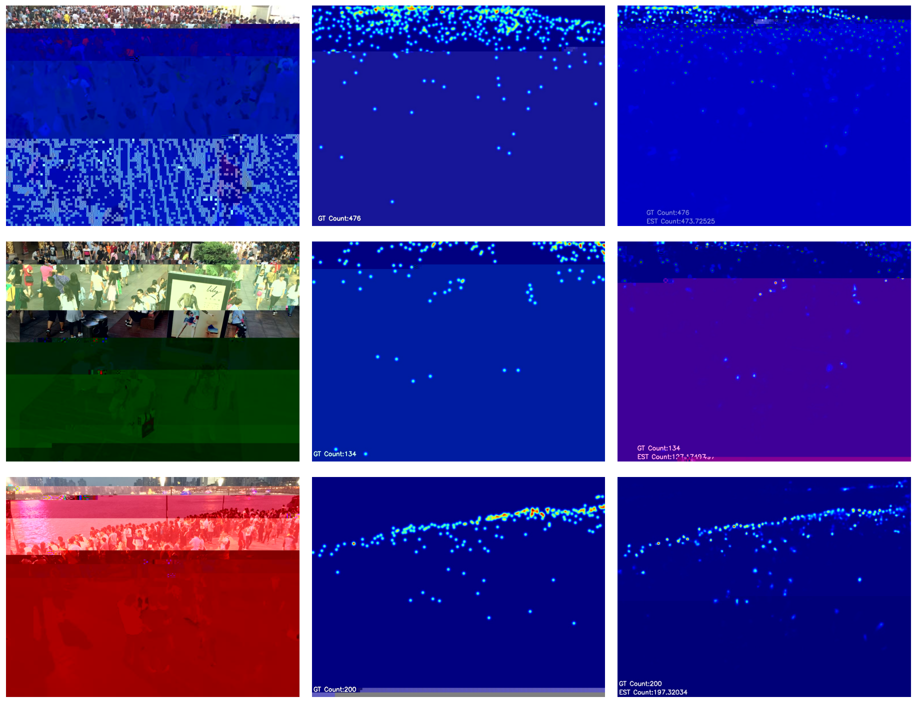


Figure 5: The density map estimated by MSA-CNN on the Shanghai Tech Part_B dataset, the first column is test images, the second column is ground truth density map, and the third column is estimated density map by our approach (MSA-CNN).

4.2 Experiment on UCF_CC_50 dataset

UCF_CC_50 was first introduced by [12]. It is a challenging dataset which consists of 50 images of the crowd, with a total of 63,974 persons. The crowd counts range from 96 to 4543. There is a large variation of crowd density in the image. Following [12], we also use five-fold cross-validation to report the average test performance. The author in [15] proposed to combine multiple source information such as Fourier analysis, head detection and texture features to generate density map and crowd counting. A comparative result with the existing six methods is shown in Table 3. Our method achieves lower error than other methods. Figure 6 shows some examples of visualization obtained by our method on the UCF_CC_50 dataset.

4.3 Comparisons with State-of-the-art

The proposed approach is compared with several state-of-the-art methods on two benchmarks, and the results are shown in Table 2, 3. Table 2 indicates comparison on ShanghaiTech datasets; the proposed MSA-CNN obtains significant improvement over prior methods, and acquires the best MAE and MSE on the Part_A dataset. This dataset is closer to the realistic monitoring screens than the others, which states that our algorithm has a good performance on the actual scenes and achieves better stability. It also shows a good result on the Part_B dataset, it shows the robustness of the proposed method which can be applied to scenes with sparse crowds. In Table 3, we compare the performance of MSA-CNN with other methods using MAE and MSE as metrics on the UCCF_CC_50 dataset. MSA-CNN outperforms all others methods in MAE and gets a com-

Table 3: Comparisons on UCCF_CC_50 dataset.

Method	MAE	MSE
[12]	419.5	541.6
[15]	467.0	498.5
MCNN [19]	377.6	173.2
Cascaded-MTL [35]	322.8	341.4
Switching-CNN [18]	318.1	439.2
CP-CNN [30]	295.8	320.9
MSA-CNN (ours)	293.9	361.6

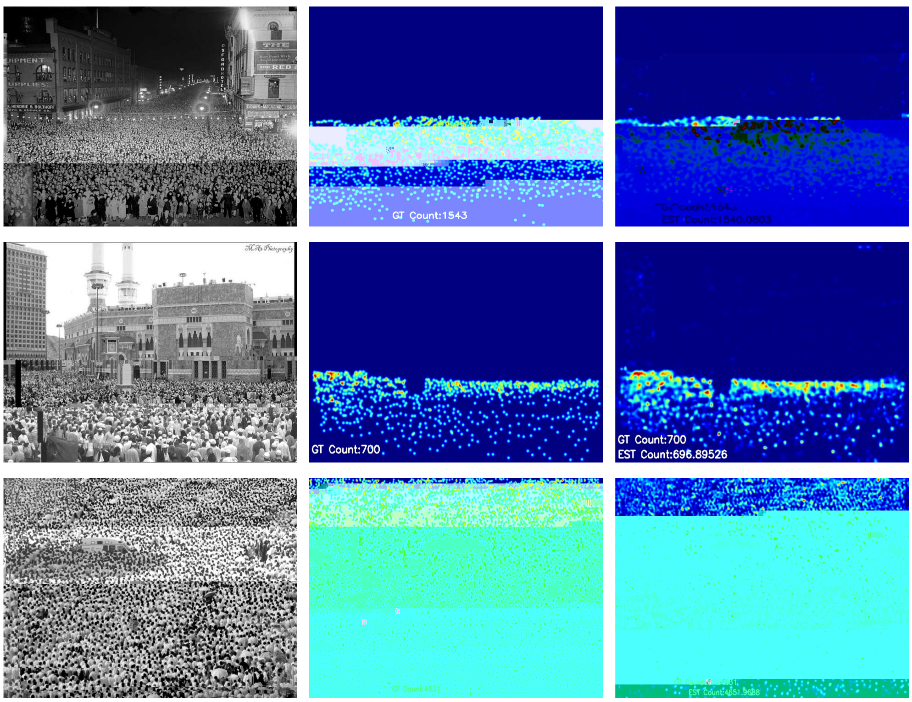


Figure 6: The density map estimated by MSA-CNN on the UCF_CC_50 dataset, the first column is test images, the second is ground truth density map, and the third is estimated density map by our approach (MSA-CNN).

petitive MSE score, which indicates the robustness of predicted count. Considering practical applications of crowd counting algorithm, we perform a simple and practical study. As shown in Table 4, MCNN has the least parameters, and CP-CNN is 500 times more than MCNN. In contrast, our algorithm has a relatively small amount of parameters.

Table 4: Number of parameters(in millions).

Method	Number of parameters
[12]	22.5
MCNN [19]	0.13
Switching-CNN [18]	15.1
CP-CNN [30]	68.4
MSA-CNN (ours)	0.55

5 Conclusion

In this paper, a multi-scale adversarial convolutional neural network is designed for estimating crowd density map and the number of pedestrians in crowd images. The improved multi-column convolutional neural network is used to extract high-dimensional feature maps. These fractionally-strided convolutional layers try to recover the loss of detail caused by previous max-pooling layers. Since, we adopted the advantage of the superior performance of GAN in image reconstruction, thereby improving the resolution of the estimated density map and reducing the crowd estimation error. The model is trained in an end-to-end manner by optimizing a weighted combination of Euclidean loss and adversarial loss and the number of parameters is low. A lot of experiments on challenging datasets are conducted, in contrast to the existing methods, our method demonstrated significant improvements.

References

[1] B.B. Zhan, D.N. Monekosso, P. Remagnino, S.A. Velastin, and L.Q. Xu. Crowd analysis: a survey. *Machine Vision and Applications*, 19(5):345–357, 2008.

[2] J. Shao, K. Kang, C.C. Loy, and X.G. Wang. Deeply learned attributes for crowded scene understanding. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 4657–4666, 2015.

[3] K. Chen, C.L. Chen, S.G. Gong, and T. Xiang. Feature mining for localised crowd counting. In *British Machine Vision Conference*, 2012.

[4] V. Lempitsky and A. Zisserman. Learning to count objects in images. In *Neural Information Processing Systems*, pages 1324–1332, 2010.

[5] E. Walach and L. Wolf. Learning to count with cnn boosting. In *European Conference on Computer Vision*, Computer Vision ÎC ECCV 2016, pages 660–676. Springer International Publishing, 2016.

[6] Y. Wang and Y. Zou. Fast visual object counting via example-based density estimation. In *IEEE International Conference on Image Processing (ICIP)*, pages 3653–3657, 2016.

[7] M.R. Hsieh, Y.L. Lin, and W.H. Hsu. Drone-based object counting by spatially regularized regional proposal network. In *IEEE International Conference on Computer Vision*, pages 4165–4173, 2017.

[8] R.D. Oñoro and S.R.J. López. Towards perspective-free object counting with deep learning. In Bastian Leibe, Jiri Matas, Nicu Sebe, and Max Welling, editors, *Computer Vision ÎC ECCV 2016*, pages 615–629. Springer International Publishing, 2016.

[9] E. Toropov, L.Y. Gui, S.H. Zhang, and S. Kottur. Traffic flow from a low frame rate city camera. In *IEEE International Conference on Image Processing*, pages 3802–3806, 2015.

[10] S.H. Zhang, G.H. Wu, J.P. Costeira, and J.M.F. Moura. Fcn-rlstm: Deep spatio-temporal neural networks for vehicle counting in city cameras. In *IEEE International Conference on Computer Vision*, pages 3687–3696, 2017.

[11] S.H. Zhang, G.H. Wu, J.P. Costeira, and J.M.F. Moura. Understanding traffic density from large-scale web camera data. In *IEEE Computer Vision and Pattern Recognition*, 2017.

[12] H. Idrees, I. Saleemi, C. Seibert, and M. Shah. Multi-source multi-scale counting in extremely dense crowd images. In *Computer Vision and Pattern Recognition*, pages 2547–2554, 2013.

[13] A. Bansal and K.S. Venkatesh. People counting in high density crowds from still images. *Computer Science*, 2015.

[14] L. Boominathan, S.S.S. Kruthiventi, and V.R. Babu. Crowdnet: A deep convolutional network for dense crowd counting. In *ACM on Multimedia Conference*, pages 640–644, 2016.

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

[16] N. Paragios and V. Ramesh. A mrf-based approach for real-time subway monitoring. In *Proceedings of the 2001 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, volume 1, pages I–I, 2001.

[17] K. Tota and H. Idrees. Counting in dense crowds using deep features. *Center for Research in Computer Vision*, 2015.

[18] D.B. Sam, S. Surya, and R.V. Babu. Switching convolutional neural network for crowd counting. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2017.

[19] Y.Y. Zhang, D. Zhou, S. Chen, S.H. Gao, and Y. Ma. Single-image crowd counting via multi-column convolutional neural network. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2016.

[20] P. Isola, J.Y. Zhu, T.H. Zhou, and A.A. Efros. Image-to-image translation with conditional adversarial networks. In *Computer Vision and Pattern Recognition*, pages 5967–5976, 2016.

[21] D. Cires, U. Meier, and J. Schmidhuber. Multi-column deep neural networks for image classification. In *Computer Vision and Pattern Recognition*, volume 157, pages 3642–3649, 2012.

[22] M. Li, Z.X. Zhang, K.Q. Huang, and T.N. Tan. Estimating the number of people in crowded scenes by mid based foreground segmentation and head-shoulder detection. In *International Conference on Pattern Recognition*, pages 1–4, 2009.

[23] C. Szegedy, W. Liu, Y.Q. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, and A. Rabinovich. Going deeper with convolutions. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1–9, 2015.

[24] K. Dan, G. Douglas, and T. Hai. Counting pedestrians in crowds using viewpoint invariant training. In *British Machine Vision Conference*, 2005.

[25] C.S. Regazzoni and A. Tesei. Distributed data fusion for real-time crowding estimation. *Signal Processing*, 53(1):47–63, 1996.

[26] C. Wang, H. Zhang, L. Yang, S. Liu, and X.C. Cao. Deep people counting in extremely dense crowds. In *ACM International Conference on Multimedia*, pages 1299–1302, 2015.

[27] H. Zhang, V. Sindagi, and V.M. Patel. Image de-raining using a conditional generative adversarial network. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2017.

[28] D.G. Lowe. Distinctive image features from scale-invariant keypoints. *International Journal of Computer Vision*, 60(2):91–110, 2004.

[29] L. Zhang and M.J. Shi. Crowd counting via scale-adaptive convolutional neural network. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2017.

[30] V.A. Sindagi and V.M. Patel. Generating high-quality crowd density maps using contextual pyramid cnns. In *International Conference on Computer Vision*, 2017.

[31] A. Boesen-Lindbo-Larsen, S. Kaae-S∅nderby, H. Larochelle, and O. Winther. Autoencoding beyond pixels using a learned similarity metric. In *International Conference on Machine Learning*, 2015.

[32] I.J. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio. Generative adversarial networks. In *Neural Information Processing Systems*, 2014.

[33] D. Pathak, P. Krahenbuhl, J. Donahue, T. Darrell, and A.A. Efros. Context encoders: Feature learning by inpainting. In *IEEE Computer Vision and Pattern Recognition*, 2016.

[34] J. Johnson, A. Alahi, and F.F. Li. Perceptual losses for real-time style transfer and super-resolution. In Bastian Leibe, Jiri Matas, Nicu Sebe, and Max Welling, editors, *European Conference on Computer Vision*, pages 694–711. Springer International Publishing, 2016.

[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 *Advanced Video and Signal Based Surveillance*, 2017.