ISSN 2278-3091
Volume 8, No.6, November – December 2019
International Journal of Advanced Trends in Computer Science and Engineering
Available Online at http://www.warse.org/IJATCSE/static/pdf/file/ijatcse13862019.pdf
https://doi.org/10.30534/ijatcse/2019/13862019

Crowd Counting Using CSR-Net Architecture

Pragnyaban Mishra[1], Raju[2], Sai Rupa[3], Pavithra[4]
[1] Associate Professor, Dept of CSE (KLEF) Green Fields, Vaddeswaram,India, pragnyaban@kluniversity.in
[2] CSE Student of 4th Year at KLEF, Green Fields, Vaddeswaram, India, rajukancharla21@gmail.com
[3] CSE Student of 4th Year at KLEF, Green Fields, Vaddeswaram, India, rupakonakanchi@gmail.com
[4] CSE Student of 4th Year at KLEF, Green Fields, Vaddeswaram, India, pavithrapolakam24@gmail.com

ABSTRACT

Crowd counting or crowd estimating is a technique used to count the number of individuals in a specific scene or specific image. Manual counting of persons will become difficult when there is huge crowd density. So with upgrading technology, we need automation in this area. This problem occurs in many domains. For instance, counting the number of people in a musical concert, counting the number of devotees in a temple, counting the number of audience in a cricket stadium. Currently, CNN architecture is widely using in many computer vision-based applications. Crowd counting is one such application. We present a modified counting convolution neural network with a density map to achieve our task. The challenges, like partial occultation, overlapping of peoples in an image may be viewed as a hindrance for counting the crowd. These challenges don't be an obstacle to our proposed model. Our CSR-Net Model takes the image as input and from the input image it creates a density map based on the ground truth values generated from the image and from the generated density map we predict the total count of people presented in the image. We trained our model using ShanghaiTech Dataset

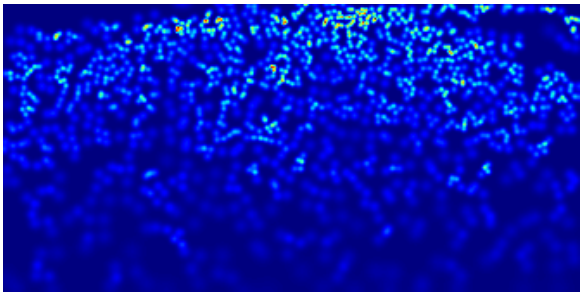
Key Words: CSR-Net , CNN , Architecture , Density Map , Ground Truth , Predict

1. INTRODUCTION

There are many techniques are presented for predicting the people in an image. We developed a better solution for crowd counting to a congested scene image. Present approaches for congested scene analysis are developed based on simple crowd counting techniques (It is just like a moving detector it just returns the count of people who is easily identified). But for the congested scene images like the images are shown here, this simple approach cannot analyze all people in the given image, so if we create the density map for the targeted image then we get a more accurate result for the given image. The distribution map helps us to get more accurate and comprehensive information, which could be critical for making correct decisions in high-risk environments. We have seen several techniques to create a distribution map and it's a challenging task to generate an accurate distributed pattern.

Crowd-Image

Density Map



Crowd-Image containing 1148 people in the dataset ShanghaiTech Part A, Image in second row shows the **distributed map** generated for the Crowd-Image

One major issue can be seen in predicting style, where the generated result values follow pixel by pixel prediction, output density maps must include spatial coherence so that they can get the smooth transition between the nearest pixels. Also, the diversified scenes, for example, irregular crowd clusters and completely different camera views, would create the task difficult, particularly for exploitation new ways without using DNNs. The recently developed approaches on crowd counting depend on DNN-Based ways because of its high accuracy they achieved in semantic segmentation tasks [1, 2, 3, 4, 5] and also the significant progress they need to be created in visual prominence [6]. The bonus of exploitation DNNs comes from the greater hardware community whenever DNNs are quickly investigated and enforced on GPUs [7], FPGAs [8, 9, 10], and ASICs [11]. Among them, the low-power, small-size scheme is particularly appropriate for deploying congested scene analysis are largely supported by multi-scale architecture [12 ,13,14,15,16] they achieved high performance during these fields however the architecture they used additionally introduce to significant disadvantages once

networks go deeper: a great amount of training time and non-effective branch structure[17], In [20] they used logistic model for crowd counting and [21] using segmentation map guidance they counted the crowd.
We designed our model using CSR-Net architecture for crowd counting and for generating the accurate density maps. Not similar to the current works like [12, 13] because they use the Deep-CNN as additional, we mainly focused on designing a density map generator based on CNN. Our model's backbone is pure convolution layers that support the input image for flexible resolution. To reduce the network complexity, we used the small size of convolution filters in all layers. We expanded the first 10 layers from VGG-16[18] which acts as front-end to our model and we used back-end as dilated convolution layers which enlarges the interested fields and extract the required features without losing the resolutions(Because there are no pooling layers are presented)
The remaining paper structured as follows. Section 2 -Related Work Section 3 - Proposed Solution and Section 4 – Result Section 5 – Conclusion

2. RELATED WORK

They are plenty of techniques for predicting the total number of people presented in an image, here are the few approaches that we studied.

2.1 Detection Based Approach

Here, we tend to use a moving window-like detector to spot individuals in a picture and return the total of identified spots. The strategies used for detection need well knowledge detectors which can extract low-level features. Though these ways work well for the detection of faces, they do not perform well on huddled pictures as most of the target objects aren't visible

2.2 Regression Based Approach

In the above approach, we can't get the low-quality features in an image. By using the regression-based technique we can overcome this issue. In the regression-based approach, Firstly the image is cropped in toPatches and for each patch features are extracted

2.3 Density-Estimation Based Approach

Even after using the regression-based approach we aren't unable to identify some faces in the image, so here in the density estimation based approach, First it creates a distribution map for the given image and from that density map using the algorithm it maps between the extracted features and the density map.

2.4 CNN Based Approach

On seeing the previous approaches better approach for predicting the count of people presented in an image. Unlike the regression-based approach where we create patches for the image and get the low-quality features, here we will build an end-to-end regression method using a convolution neural network. It processes the whole image and generates the count of people presented in the image. This approach will give better and more accurate results when compared to the previous approaches and this approach generates a quality density map

2.5 Limitations of Advanced Approach

Most recently, [12] proposed the Switch-CNN using a density level classifier to choose different regressors for particular input patches. Sindagi [13] present a Contextual Pyramid CNN, which uses CNN networks to estimate context at various levels for achieving lower count error and better quality density maps. These two approaches achieved better performance, and they use density level classifier and Multi column-based architecture. Anyhow, we noticed many disadvantages in these approaches one main disadvantage is the training of the model, It is very hard to train the model using this method.[17]. Such a distended network structure needs a lot of time to train the model (2) whereas multi-column architecture introduces unwanted things from its structure Section-1. Different columns seem to perform similarly without obvious differences. (3) Before sending a picture to multi column-based architecture both of these approaches need the density level classifiers. However, it is very hard to generate a density map which shows the accurate result of on congested scene, In a congested scene, the object pixel value changes highly. And also, It is more complicated to design and it causes unwanted values because of using the fine-grained classifiers. (4) For labeling the parameters to the input region and allocation of final parameters to the generation of distribution map, we need to spend more time on parameters. On viewing the structure of MCNN due to the lack of parameters it generates a very low quality of density map.
By knowing all the disadvantages of these approaches we proposed a unique approach that extracts the maximum features from the given image and generates the density map.

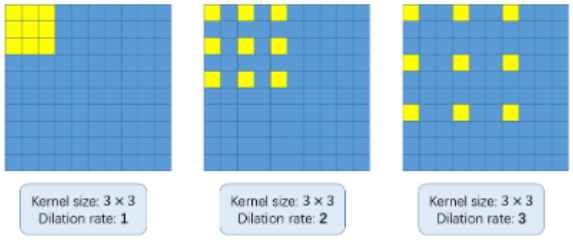
3. PROPOSED SOLUTION

CSR-Net is a technique we implemented in our model, deploys a deeper CNN for capturing high-level features and generating high-quality density-maps without increasing the complexity of network

3.1 CSRNet architecture

This uses the VGG-16 as the front end because VGG-16 has high abilities in transfer learning. The output from the VGG-16 is 1/8th of the given input size. And the CSR-Net uses a convolution layer as a back end called Dilated Convolution Layer.

Consider the below image:



The concept behind this convolution layer is it increases the kernel size without increasing the parameters so that we can extract the low-quality features very easily. On incrementing the dilation rate you can see how it increases the kernel like in the above picture, so if you increase the dilation rate then we can extract the low-level features very easily. This can be used as a pooling layer

Let's consider a input-image x(m,n), a image-filter w(i,j), and assume the dilation rate as r. The result y(m,n) will be:

$$y(m,n) = \sum_{i=1}^{M}\sum_{j=1}^{N} x(m + r * i\,, n + r * j)w(i,j)$$

We can conclude the given equation using a (k*k) kernel with a dilation rate r. The kernel increases to:

$$([k + (k\text{-}1)*(r\text{-}1)] * [k + (k\text{-}1)*(r\text{-}1)])$$

The ground truth values are created for the image. Using the Gaussian kernel each person's head is blurred in the given input image and further the image is divided into a total of 9 patches. 1/4th of the original image is contained in each patch

4 quarters are cropped from the first 4 patches and random cropping will take place for the remaining 5 patches. Finally, the training set contains the double of the image from each patch

Table 1: Configuration on CSR-Net

Configuration of CSR-Net			
A	B	C	D
input(unfixed-resolution color image)			
front-end (fine-tuned from VGG-16)			
conv3-64-1 conv3-64-1			
max-pooling			
conv3-128-1 conv3-128-1			
max-pooling			
conv3-256-1 conv3-256-1 conv3-256-1			
max-pooling			
conv3-512-1 conv3-512-1 conv3-512-1			
back-end (four different configurations)			
conv3-512-1 conv3-512-1 conv3-512-1 conv3-256-1 conv3-128-1 conv3-64-1	conv3-512-2 conv3-512-2 conv3-512-2 conv3-256-2 conv3-128-2 conv3-64-2	conv3-512-2 conv3-512-2 conv3-512-2 conv3-256-4 conv3-128-4 conv3-64-4	conv3-512-4 conv3-512-4 conv3-512-4 conv3-256-4 conv3-128-4 conv3-64-4
conv1-1-1			

3.2 Training Details

For training of the CSR-Net architecture completely, we use the stochastic Gradient Descent. The learning rate is fixed to 1e-6 during training and to differentiate the ground-truth value and generated density map we use the loss function as the Euclidean distance this is expressed as follow:

$$\mathbf{L}(\ominus) = \frac{\mathbf{1}}{\mathbf{2N}}\sum_{\mathbf{i=1}}^{\mathbf{N}} ||\mathbf{Z}(\mathbf{X_i}\,;\ominus) - \mathbf{Z_i^{GT}}||_{\mathbf{2}}^{\mathbf{2}}$$

Here N is the size of the trained patch and for the evaluation of model CSR-Net we use the MAE and MSE, they are expressed as follows:

$$MAE = \frac{1}{N}\sum_{i=1}^{N} |C_i - C_i^{GT}|$$

$$MSE = \sqrt{\frac{1}{N}\sum_{i=1}^{N} |C_i - C_i^{GT}|^2}$$

Ci represent the predicted value

$$C_i = \sum_{l=1}^{L} \sum_{w=1}^{W} Z_{l,w}$$

Width of the density map is represented by L and W.

For the given input image density map is generated, 0 represents no person presented in the pixel. And the pixel value will set a pre-defined value if there is a person presented in the pixel and the total person's presented in the image estimated based on the pixel values presented in the image

4. Result

In this section, the evaluation metrics are applied to our model, and then an excision study of ShanghaiTech Part-A dataset is used to test our model to analyze the configuration

4.1 ShanghaiTech Dataset

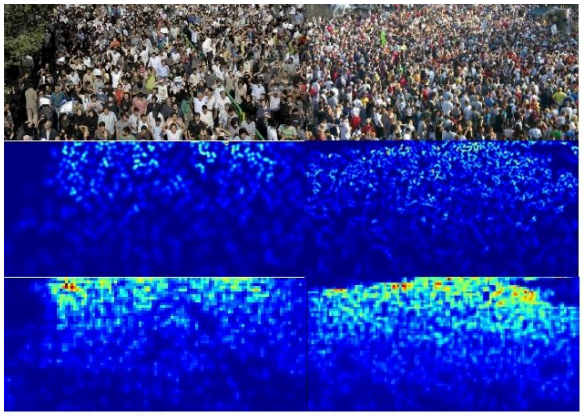
We have seen different datasets widely available on the internet but those datasets are not suitable for training and testing our model. ShanghaiTech Dataset suits for predicting count of people in an image. It has a total of 1198 images and a total of 330,165 people are presented in it. As far we studied only this dataset contains the number of people when compared to other datasets. It has two folders in it one is PART_A and another one is PART_B. In PART_A we will have a total of 482 randomly selected congested scene images from the internet and PART_B it contains a total of 716 images where these images are captured from the high metropolitan areas in Shanghai city. And the bonus of this dataset is it has ground truth values for each image. In this dataset crowd presented in each image are completely different from one image to others. We used 300 images in PART_A for training our model and remaining 182 images for testing our model and from the PART_B we have chosen 400 images for training and the remaining 316 images to test our model

4.2 Excisions on ShanghaiTech Part-A

In this part, we perform a cutting study to understand the different configurations of the CSR-Net Architecture on the ShanghaiTech Dataset PART_A [17]. It's a challenging task to estimate the count of people presented in the 482 congested scene images with various positions and different resolutions. Which contains 241,667 individual persons. So we made four configurations that are shown as below table. Each configuration will have different dilation rates. CSR-Net A is one of the networks having the dilation rate of 1 and the CSR-Net B is the second network having the dilation rate of 2 and the CSR-Net D is the fourth network having the dilation rate of 4 and the CSR-Net C is the third network having the dilation rate combination of both 2 and 4. We just want to try how well the dilation rate plays a role in the CSR-Net Architecture so we kept the remaining parameters are same. We trained the four network model using ShanghaiTech Dataset PART_A as we mentioned in section 3.2 and after training our model we tested our model using the evaluation metric process like we mention in section 3.2. Even after adding the dropout [19] there is no high improvement in our model so we didn't add the dropout in our model.
The detailed representation of our configuration model results are mentioned below and we noticed that in the CSR-Net B configuration we addressed the lowest error which means the highest accuracy is noticed in the CSR Net-B configuration, So we used CSR-Net B configuration to propose our model.

Table 2: Comparison between models

Architecture	MAE	MSE
CSR Net A	69.4	115.4
CSR Net B	**67.8**	**114.6**
CSR Net C	71.2	120.54
CSR Net D	75.6	120.78



Original Count - 384
Predicted Count - 384

Original Count - 1148
Predicted Count - 1179

In the above image, the first part is the original image and the second part is the generated ground truth result and the third part is the generated density map

Table 3: Comparison between PSNR and SSIM , Quality of density maps generated by CSR-Net in ShanghaiTech Dataset.

Dataset	PSNR	SSIM
ShanghaiTech Part-A	23.79	0.76
ShanghaiTech Part-B	27.02	0.89

5. CONCLUSION

We proposed a better method for estimating count for the given congested scene using the CSR-Net Architecture. Using this Architecture a high quality density map is generated and on comparing different dilation rates we selected the best configuration and trained our model using ShanghaiTech

Dataset PART_A [300 images], PART_B [400 images] and tested our model with ShanghaiTech Dataset PART_A [182 images], PART_B [316 images] and we noticed the highest accuracy

6. ACKNOWLEDGMENT

This work was supported by the KL University IBM Technology Research center for Computer Science Students – an industry Research collaborated lab

REFERENCES

1. Jonathan Long, Evan Shelhamer, and Trevor Darrell. Fully convolutional networks for semantic segmentation. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pages 3431–3440, 2015.
2. Yunchao Wei, Xiaodan Liang, Yunpeng Chen, Xiaohui Shen, Ming-Ming Cheng, Jiashi Feng, Yao Zhao, and Shuicheng Yan. Stc: A simple to complex framework for weaklysupervised semantic segmentation. IEEE transactions on pattern analysis and machine intelligence, 39(11):2314– 2320, 2017. https://doi.org/10.1109/TPAMI.2016.2636150
3. Yunchao Wei, Jiashi Feng, Xiaodan Liang, MingMing Cheng, Yao Zhao, and Shuicheng Yan. Object region mining with adversarial erasing: A simple classification to semantic segmentation approach. In IEEE CVPR, 2017.
4. Fisher Yu and Vladlen Koltun. Multi-scale context aggregation by dilated convolutions. In ICLR, 2016.
5. L. C. Chen, G. Papandreou, I. Kokkinos, K. Murphy, and A. L. Yuille. Deeplab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected crfs. IEEE Transactions on Pattern Analysis and Machine Intelligence, PP(99):1– 1, 2017.
6. Junting Pan, Elisa Sayrol, Xavier Giro-i Nieto, Kevin McGuinness, and Noel E O'Connor. Shallow and deep convolutional networks for saliency prediction. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pages 598–606, 2016.
7. Yangqing Jia, Evan Shelhamer, Jeff Donahue, Sergey Karayev, Jonathan Long, Ross Girshick, Sergio Guadarrama, and Trevor Darrell. Caffe: Convolutional architecture for fast feature embedding. In Proceedings of the 22nd ACM international conference on Multimedia, pages 675–678. ACM, 2014.
8. Jiantao Qiu, Jie Wang, Song Yao, Kaiyuan Guo, Boxun Li, Erjin Zhou, Jincheng Yu, Tianqi Tang, Ningyi Xu, Sen Song, Yu Wang, and Huazhong Yang. Going deeper with embedded FPGA platform for convolutional neural network. In Proceedings of the 2016 ACM/SIGDA International Symposium on Field-Programmable Gate Arrays, FPGA '16, pages 26–35, New York, NY, USA, 2016. ACM.
9. Xiaofan Zhang, Xinheng Liu, Anand Ramachandran, Chuanhao Zhuge, Shibin Tang, Peng Ouyang, Zuofu Cheng, Kyle Rupnow, and Deming Chen. Highperformance video content recognition with long-term recurrent convolutional network for FPGA. In Field Programmable Logic and Applications (FPL), 2017 27th International Conference on, pages 1–4. IEEE, 2017. https://doi.org/10.23919/FPL.2017.8056833
10. Xiaofan Zhang, Anand Ramachandran, Chuanhao Zhuge, Di He, Wei Zuo, Zuofu Cheng, Kyle Rupnow, and Deming Chen. Machine learning on FPGAs to face the IoT revolution. In Computer-Aided Design (ICCAD), 2017 IEEE/ACM International Conference on, pages 819–826. IEEE, 2017.
11. Renzo Andri, Lukas Cavigelli, Davide Rossi, and Luca Benini. Yodann: An ultra-low power convolutional neural network accelerator based on binary weights. In VLSI (ISVLSI), 2016 IEEE Computer Society Annual Symposium on, pages 236–241. IEEE, 2016. https://doi.org/10.1109/ISVLSI.2016.111
12. Deepak Babu Sam, Shiv Surya, and R Venkatesh Babu. Switching convolutional neural network for crowd counting. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, volume 1, page 6, 2017.
13. Vishwanath A Sindagi and Vishal M Patel. Generating highquality crowd density maps using contextual pyramid CNNs. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pages 1861–1870, 2017.
14. Yunchao Wei, Xiaodan Liang, Yunpeng Chen, Xiaohui Shen, Ming-Ming Cheng, Jiashi Feng, Yao Zhao, and Shuicheng Yan. Stc: A simple to complex framework for weaklysupervised semantic segmentation. IEEE transactions on pattern analysis and machine intelligence, 39(11):2314– 2320, 2017. https://doi.org/10.1109/TPAMI.2016.2636150
15. 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, pages 640–644. ACM, 2016. https://doi.org/10.1145/2964284.2967300
16. Daniel Onoro-Rubio and Roberto J L´opez-Sastre. Towards perspective-free object counting with deep learning. In European Conference on Computer Vision, pages 615–629. Springer, 2016. https://doi.org/10.1007/978-3-319-46478-7_38
17. 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, pages 589–597, 2016.

18. Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. arXiv preprint arXiv:1409.1556, 2014.
19. 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, pages 640–644. ACM, 2016. https://doi.org/10.1145/2964284.2967300
20. Byung Mook Weon , Logistic model for crowd counting, pages 4. Publication 336869124 IJATCSE Oct 2019
21. Hao Xu, Chengyao Zheng, Yuncong Nie, Siyu Xia, Crowd counting with segmentation map guidance, In Proceedings of the 2019 Chinese Control Conference, ,pages 1-6, IJATCSE,July 2019 DOI: 10.23919/ ChiCC.2019. 8865761