Crowd Counting via Residule Attention Network

Yijun Yuan

Abstract

In this paper, we build a CNN based method to map the crowd scene image to its density distribution map. First, we develop a Residule Attention Network that can take arbitrary size image as input and predict the crowd distribution. Then, a novel geometry-adaptive kernel is designed to automaticly alleviate the fusing between people, which can provide a very high quility true density map. We evaluate our work on ShanghaiTech and UCSD dataset. Outperform existing state-of-the-art method on ShanghaiTech partB and UCSD dataset.

1. Introduction

Recently there is a great interest in understanding environment. This series of technique is meaningful on security, resource scheduling, advertising. The basic computer vision technique on environment analysis are detection, tracking and behaviour analyzing. However, in crowd scene, all of those tasks can not be resolved properly most because the object in scene, pedestrain for example, are too small, it may only take few pixel each.

Crowd scene understanding is becoming more and more important thses years since it can help predict abnormal conduct, such as making disturbance and criminal behavior, evaluate and reschedule the public space, for example, monitoring pedestrian flow rate and disigning of the air port terminal.

Thus it is a very requirement that some new tasks that can monitor on global level to understanding the crowd environment. Here crowd counting are one of those tasks on high level. The most primier crowd counting task can provide the count of object in certain genre. While recently, instead of directly obtain the count number, people generate predict the object distribution first, and then count the number from previous density map. Also, some people directly locate people from the predicted density map with geometric algorithms.

Although crowd counting is a very basic problem which aim to detect the count of people in certain crowd scene, there are many challenges in this problem. For the scene, it may contain occlusion and non-uniform illumination. For image quality, because most of the image resource are from monotor beside the walkway, the resolution may be very low, and the distorsion could happen. In some very crowd scene, it is almost impossible to indicate a person with human eyes.

Traditional computer vision approach on this problem are on top of Detection[12], regression[13] and density estimation[15]. While generally, CNN-based methods[2][3][4][5][6] outperform the traditional computer vision approach[12][13][15], thus this paper will only compare with CNN-based method.

CNN-based method mainly differ on network and training process. On network, there exists scale-aware models, context-aware models and multi-task models. And on training process, it can be patch-based and End-to-End training.

In this paper, we use an end-to-end Attention Residule Network to predict a crowd distribution density map. This Net can be divided into two parts, U-net part and Attention net part. Inspired by successful multi-scale architecture[2][7] and attention residule model [1], Attention net take two U-net to generate feature and attention, then follow [1] to utilize this attention.

The contribution of our work can be divide into two folder: 1. A very light network on top of residule attention that achieve outperformance comparing with state-of-the-art method on dataset we use. 2. A novel geometric adaptive kernel that unquestionable adquate to alleviate the fusing of different people on the groudtruth density map.

2. Related work

Recent years, CNN-based method can be well applied to different tasked. And various CNN-based approach also achieve very reasonable progress on this field.

[16] direct achieve crowd scene people count, while more works predict a distribution density map then integrate the map to obtain the count number instead. The [7] shows with the help of density map prediction, counting process will be more accurate.

[2] take advantage of multi-column, achieved a state-of-the-art performance with a very light network by training three branches of CNN with different size of convolution to take different scale into consideration. [3] take concentration on the crowd distribution, put the crowd patch and

sparse patch into different CNN. [9] instead of only predict a density map, one more branch is appended to classify the crowd level. [16] directly predict human count with multiple branches while a gate is used to weight those prediction.

From the previous work, we find our approach should take great effort on three types of property, in order to considerablly resolve this problem. They are contexture, localization and sparsity.

For the contexture and localization, [17] provide a famous structure called U-net, the left half network downsample the features to achieve better contexture feature, while the skip line from left half to the unsampled right half network can put the localization together with the contexture. U-net can properly handle the first two property.

Generally, there are several method people tend to use to handle sparsity. Patchlize the whole image and indicate each of those pathes to be dense or sparse. Or evaluate the sparsity of whole image by setting several density level before predicting density map. There are also many other methods to predict the sparsity.

Here we will propose a clear but robust way on this property.

Inspired by the very idea in Attention Residule Network On Classification[1], attention branch can help eliminate the effect of background and enhance good features in foreground, we use attention residule on the whole network level, to help intensify the value in region of interest. Thus the sparsity can be settled in a global way.

For those End-to-End method, the quility and the conceal structure the groundtruth will determine the prediction because the ground-truth can be considered as a upper bound for prediction quality.

In state-of-the-art methods, people use distance between person to help simulate the interaction in between and provide structure contexture for the scene[2].

The geometric adaptive kernel in [2] attempts to alleviate the fusing of gaussian distribution between people. And in this paper, we propose a more effective way to handle this fusing problem.

3. Methodology

We here employed a light CNN network for crowd distribution prediction. Our method contains two part, CNN model and a geometric adaptive kernel.

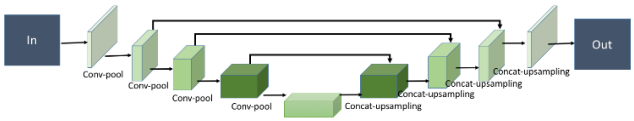

Figure 1: The basic U-net unit in our model.

3.1. Residule Attention Network

Our Residule Attention Network can be find in fig.2 It can be consider as a attention model metioned in [1]. This attention model can be divided into two branches: mask branch and trunk branch. The trunk branch provide basic feature while the mask branch help to select the basic feature. Considering both branch are required to take fully use of contexture and localization, we put two seperate U-net on each branch instead. The Unet structure can be find in fig.1. From MCNN[2] we find very light network can also provide a great performance on this task, with features only have very limited channel. Here our U-net feature channel from left to right are 8, 16, 32, 64, 128, 64, 32, 16, 8. Since downsampling in left-half of U-net tend to reduce feature, we double its channel before each max-pooling. For Crowd counting problem, high localization is not too important comparing with contexture, so during the concatenating, we put upsamled feature twice of skipped feature from left-half network. After achieve the mask feature $T(x)$ and trunk

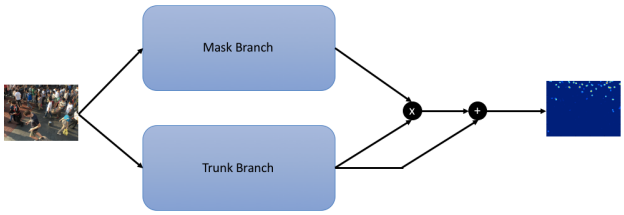

Figure 2: Structure of Residule Attention Network.

feature $M(x)$ with input x, we weighted $T(x)$ with $M(x)$ for which the mask is used as control gates to select the trunk feature. The output feature F should be:

$$F_{i,c}(x) = M_{i,c}(x) * T_{i,c}(x) \tag{1}$$

where $i \in \{1, 2, 3...N\}$, N is the number of spatial pixel of image, and c range over all channels.

Also, the attention mask can serve as a gradient update filter during back propagation. The gradient of mask for input feature in soft mask branch are as follow:

$$\frac{\partial M(x,\theta)T(x,\phi)}{\partial \phi} = M(x,\theta)\frac{\partial T(x,\phi)}{\partial \phi} \tag{2}$$

where θ, ϕ indicate the parameters of mask and trunk branch respectively. From Residule Attention Network paper[1],

> this property make attetnion modules robust to noisy labels. Mask branches can prevent wrong gradients (from noisy labels) to update trunk parameters.

While naively dot producting with mask feature ranging from zero to one will naturely degrade the value of trunk feature. Thus there should be a ensurement that the attention model can be more worse than a single trunk branch. Hence one way to reformula the output feature is:

$$F_{i,c}(x) = (1 + M_{i,c}(x)) * T_{i,c}(x) \tag{3}$$

where $M(x)$ is mask feature range from 0 to 1, $T(x)$ is basic feaure. When it is not easy to properly learn a attention mask, $(1 + M(x))$ will be approximate to 1, which will result in the original trunk feature.

3.2. Geometric Adaptive Kernel

Inspired by [18], which attempt to achieve a high quility density map that can almost indicate each person from density distribution, we tend to belive do progress on groundtruth distribution generation will work because end-to-end learning is always trying to generate prediction having the groundtruth structure.

The geometric adaptive kernel is attempt to alleviate the fusing between people's distribution. We write GDK as a simplified form. Hence an intuitive direction is by using directed vector.

The generation algorithms can be divide into several setps: (Give kernel size k and sigma σ)

1. General gaussian distribution density map for x vector and y vector.

First we generate 1-D gaussian kernel on x direction and y direction to also fulfill the kernel box. To make it a directed vector, we make sure certain elements on the left of center on x kernel will be negative, otherwise positive. Also elements above the center on y kernel will be negative, otherwise positive.

Hence we have

$$P(x^{'}) = \sum_{i=1}^{N} \delta(x^{'} - x_i^{'}) \tag{4}$$

$$GT_x(x^{'}) = P(x^{'}) * G_\sigma \tag{5}$$

$$GT_y(x^{'}) = P(x^{'}) * G_\sigma \tag{6}$$

Where P is the pulse function, $x^{'}$ is the pixel range from 1 to N on certain row in y map, certain column on x map. G_σ is the directed gaussian function. GT_x, GT_y are x density map and y density map respectively.

And because each kernel have both negative part and positive part, when the directed kernel of two people will not fuse like general kernel, it will ellimite instead.

2. Achieve enhancement ratio

From the difinition of 2D normal distribution, it can be divide into two 1D normal distribution if they are independently distributed:

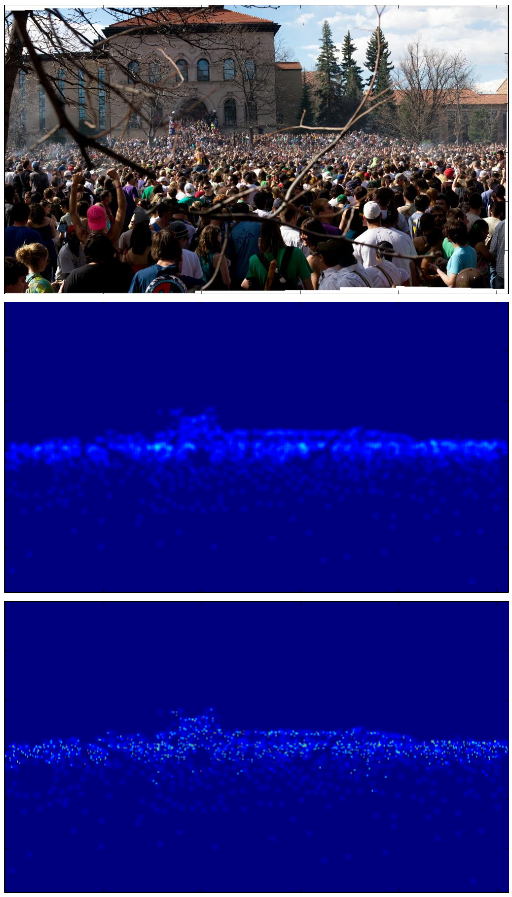

Figure 3: The qualitive comparison between true maps generated by our Geometric adaptive kernel and general gaussian kernel(both with kernel size 15, σ 5). From top to bottom are original RGB image, map generated by gaussian kernel, map generated by Our geometric adaptive kernel

$$G_{\sigma xy} = G_{\sigma x} \cdot G_{\sigma y} \tag{7}$$

From previous stage, we achieved the directed density map. Then by product those two absolute x,y map, the new producted map can tell the infomation of field, which is very similar to electronic field. It is clear to find that the "energy" field of each people will reject each other.

And here it is easy to find, if one person is not close to others, the gaussian distribution of it won't change.

Then normalize each kernel region from the predicted map, the value in center of each kernel will be enhanced if it was effected in previous step.

So it is plausible to define the enhance ratio:

$$r_{i,\sigma} = \frac{V_i^{'}}{V_i} \tag{8}$$

where r is what we called the enhance ratio, i is the $i-th$ person in the scene, $V^{'}$ is the element value in the map from step 2 on human $i's$ localtion, V is the center value in the general 2D gaussian kernel.

3. Achieve adaptive σ for each person

So the last step is to generate our Geometric Adaptive kernel.

Since each person have its own enhance ratio r, the updated sigma is:

$$\sigma_i^{'} = \frac{\sigma_i}{r_i} \tag{9}$$

where $\sigma_i^{'}$ are for person i on adaptive kernel, σ_i are for original kernel.

4. Experiments

We evaluate our model on two different datasets - UCSD dataset and ShanghaiTech partB. No augmentation is taken in this experiment. The implementation and training are on Tensorflow framework on one GeForce GTX TITAN X.

4.1. Evaluation metrics

Following most of the crowd counting works, we implement MAE and MSE to evaluate the performance:

$$MAE = \frac{\sum_{i=1}^{N} |y_i - \widehat{y}_i|}{N} \tag{10}$$

$$MSE = \sqrt{\frac{\sum_{i=1}^{N} (y_i - \widehat{y}_i)^2}{N}} \tag{11}$$

where i is the image index range from 1 to total test number N, y_i are prediction and $\widehat{y}_i$ are groundtruth count for the $i-th$ image.

4.2. Dataset

The dataset we use are UCSD dataset[11] and Shanghaitech partb[2].

4.2.1 UCSD dataset

The UCSD dataset was collected from a camera at walkway. It is among the first datasets for people counting. It contains 2000 frames with size 158 $\times$ 238. This dadaset are collected from a walkway in the UCSD compus. It also provide ROI for the compus image to filter out the irrelevant object besides the walkway. This dataset is split into training set and testing set. While the training set contrains frames from range 601-th to 1400-th. The rest frame are for testing.

UCSD dataset also provide some extra info, which will not be utilized in our experiment because it won't help much from our experience.

This dataset has relatively low density with an average of 15 people in a frame[] from a same view point on a single location.

4.2.2 ShanghaiTech dataset

ShanghaiTech dataset are introduced by [2]. It contains 1198 annotated images, with a total of 330,165 people with head annotated. This dataset is among the largest on ther number of annotated people. This dataset contains two partitions: Part A and Part B. 482 images from Part A are randomly crowled from Internet. And 716 images of Part B are taken from the busy streets of metropolitan areas in Shanghai.

Both ShanghaiTech Part A and Part B are split into training set and testing set. 300 of Part A for training and the rest for testing. 400 images of Part B for training and the rest for testing.

ShanghaiTech Part A has much larger density images comparing with Part B. The crowd density between two dataset varies significantly, large varying scale range, perspective distortion and complex daily scene make it more challenge to count than ever dataset.

4.3. Evaluation

Table 1: Comparison between different crowd counting methods on UCSD.

	MAE	MSE
Zhang et al.[4]	1.60	3.31
Zhang et al.[2]	1.07	1.35
Walach and Wolf[5]	1.10	
Kang et al.[7]	1.12	2.06
Sheng et al.[8]	2.86	13.0
Sam et al.[10]	1.62	2.10
Ours w/o GDK	**1.0652**	1.3514

Table 2: Comparison between different crowd counting methods on ShanghaiTech Part b.

	MAE	MSE
Zhang et al.[4]	32.0	49.8
Zhang et al.[2]	26.4	41.3
Marsden et al.[6]	23.76	33.12
Sindagi et al.[9]	20.0	**31.1**
Sam et al.[10]	21.6	33.4
Ours w GDK	**17.4890**	31.4609

From above tables, we find our work achieve better MAE on UCSD and ShanghaiTech part b than all of those state-of-the-art CNN based method.

5. Conclusion

In this paper, we have developed a crowd counting method, based on a novel deep learning architecture. We employ residule attention network on UCSD and ShanghaiTech dataset, achieve outperformance comparing with state-of-the-art works on UCSD and ShanghaiTech Part b.

References

[1] Fei Wang, Mengqing Jiang, Chen Qian, Shuo Yang, Cheng Li, Honggang Zhang, Xiaogang Wang, Xiaoou Tang. Residual Attention Network for Image Classification, in: CVPR, arXiv preprint arXiv: 1704.06904 (2017).

[2] Y. Zhang, D. Zhou, S. Chen, S. Gao, Y. 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. 589597.

[3] D.B. Sam, S. Surya, R.V. Babu, Switching convolutional neural network for crowd counting, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2017.

[4] C. Zhang, H. Li, X. Wang, X. Yang, Cross-scene crowd counting via deep convo- lutional neural networks, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2015, pp. 833841.

[5] E. Walach, L. Wolf, Learning to count with CNN boosting, in: European Con- ference on Computer Vision, Springer, 2016, pp. 660676.

[6] M. Marsden, K. McGuiness, S. Little, N.E. OConnor, Fully convolutional crowd counting on highly congested scenes, arXiv preprint arXiv:1612.00220 (2016).

[7] D. Kang, Z. Ma, A.B. Chan, Beyond counting: comparisons of density maps for crowd analysis tasks-counting, detection, and tracking, arXiv preprint arXiv: 1705.10118 (2017).

[8] B. Sheng, C. Shen, G. Lin, J. Li, W. Yang, C. Sun, Crowd counting via weighted VLAD on dense attribute feature maps, IEEE Trans. Circuits Syst. Video Technol. (2016).

[9] V. Sindagi, V. Patel, Cnn-based cascaded multi-task learning of high-level prior and density estimation for crowd counting, in: Advanced Video and Sig- nal Based Surveillance (AVSS), 2017 IEEE International Conference on, IEEE, 2017.

[10] D.B. Sam, S. Surya, R.V. Babu, Switching convolutional neural network for crowd counting, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2017.

[11] A.B. Chan, Z.-S.J. Liang, N. Vasconcelos, Privacy preserving crowd monitor- ing: Counting people without people models or tracking, in: Computer Vision and Pattern Recognition, 2008. CVPR 2008. IEEE Conference on, IEEE, 2008, pp. 17.

[12] P. Dollar, C. Wojek, B. Schiele, P. Perona, Pedestrian detection: an evaluation of the state of the art, IEEE Trans. Pattern Anal. Mach. Intell. 34 (4) (2012) 743761.

[13] C.C. Loy, K. Chen, S. Gong, T. Xiang, Crowd counting and profiling: methodol- ogy and evaluation, in: Modeling, Simulation and Visual Analysis of Crowds, Springer, 2013, pp. 347382.

[14] Vishwanath A. Sindagi, Vishal M. Patel. A survey of recent advances in CNN-based single image crowd counting and density estimation, in: Pattern Recognition Letters, arXiv preprint arXiv:1707.01202 (2017).

[15] V.-Q. Pham, T. Kozakaya, O. Yamaguchi, R. Okada, Count forest: co-voting un- certain number of targets using random forest for crowd density estimation, in: Proceedings of the IEEE International Conference on Computer Vision, 2015, pp. 32533261.

[16] S. Kumagai, K. Hotta, T. Kurita, Mixture of counting CNNs: adaptive inte- gration of cnns specialized to specific appearance for crowd counting, arXiv preprint arXiv:1703.09393 (2017).

[17] O. Ronneberger and P.Fischer and T. Brox. U-Net: Convolutional Networks for Biomedical Image Segmentation, in Medical Image Computing and Computer-Assisted Intervention (MICCAI), preprint arXiv:1505.04597.

[18] Generating High-Quality Crowd Density Maps using Contextual Pyramid CNNs. Vishwanath A. Sindagi, Vishal M. Patel, in CVPR, preprint arXiv:1708.00953.