Available online at www.sciencedirect.com

ScienceDirect

Procedia Computer Science 111 (2017) 154–159

Procedia Computer Science

www.elsevier.com/locate/procedia

8th International Conference on Advances in Information Technology, IAIT2016, 19-22 December 2016, Macau, China

Estimation of crowd density in surveillance scenes based on deep convolutional neural network

Shiliang Pu, Tao Song*, Yuan Zhang, Di Xie

[a]*Hikvision Research Institute, Hangzhou, China*

Abstract

As an effective way for crowd monitoring, control and behavior understanding, crowd density estimation is an important research topic in artificial intelligence applications. In this paper, we propose a new crowd density estimation method by deep convolutional neural network (ConvNet). The contributions are two-folds: first, typical deep networks are imported for crowd density estimation. Second, a new dataset including 31 crowd Subway-carriage scenes with over 160K density annotated images is introduced to better evaluate the accuracy of cross-scene crowd density estimation methods. Experiment results confirm the good performance of our proposed method for real-world application.

Peer-review under responsibility of the organizing committee of the 8th International Conference on Advances in Information Technology

Keywords: Crowd density; convolutional neutral network; deep learning.

1. Introduction

Crowd density estimation is mainly used in public places such as pavements, malls, stations and the squares. The estimation methods could lead to a better understanding of crowd behavior, which could help to improve design of the built environment and increase pedestrian safety. Crowds may be greatly various in their distributions and shape patterns, which is a main obstacle for pattern recognition. What's more, people often overlap when it is crowded and sometimes only their heads protrude, so the method of estimating crowd density by analyzing the individuals to

* Corresponding author. Tel.: +86-0571-88075998-62161.
E-mail address: songtao6@hikvision.com

1877-0509
Peer-review under responsibility of the organizing committee of the 8th International Conference on Advances in Information Technology
10.1016/j.procs.2017.06.022

count their numbers is out of the ability. To estimate the crowd density, crowd features need to be designed first and then a classifier needs to be trained to discriminate the crowd density. In the past few decades, many approaches have been developed to find out both a better way for crowd feature description and a more efficient classifier for classification[1-5].

Davies proposed to estimate stationary crowds using image processing technics such as background removal and edge-detection, and optical flow computation for the estimation of crowd motion[1]. Marana introduced in texture-based features which was based on gray level co-occurrence matrix (GLCM) and classified the crowd density using a self-organizing neural network[2]. Li used support vector machine (SVM) to enhance the estimation performance based on wavelet representation[3]. As neural network discriminates features more precisely like human neuron network, Kim showed the effectiveness of combing it with texture information and optical flow in public areas[4]. However, since the handcraft features used in these methods are designed for the certain environments, these methods could not obtain satisfactory results in practical applications.

Recently, deep convolutional neural network (ConvNet) has enjoyed a great success in many research topics such as large-scale image recognition[5], object detection[6] and semantic segmentation[7]. Fu was the first that introduced multi-stage ConvNet for crowd density estimation and showed impressive experiment results on uniform dataset[8]. However, their network was not "deep" and the effectiveness of their method is limited for real-world frames. As far as we know, there has been no literature about the research of deep ConvNet on crowd density estimation. So we appeal that more work could be done for this classification task in practical environments.

In this paper, two classic deep ConvNets, Googlenet[9] and VGGnet[10], are promoted for crowd density estimation. Besides, a new practical dataset including 31 crowd Subway-carriage scenes with over 160K annotated pictures is introduced to better evaluate the estimation accuracy by the proposed method. Experiment results confirm the effectiveness of this method for practical application. The rest of this paper is organized as follows: in Section 2, we formally define the problem and introduce the detail implements of the deep ConvNet. Section 3 introduces the newly built dataset and shows experimental results of the proposed method. Conclusions are given in Section 4.

2. Implementation

2.1. Deep ConvNets for crowd estimation

The estimation of crowd densities by deep ConvNets directly extract image features and mapping features to crowd density levels such as very low, low, medium, high and very high. The number of persons for each range and the number of ranges itself may depend on the specific application and particular characteristics of the area[2].

Benefiting from the convolution operation, images can be used directly as the input and feature maps are acquired automatically. Typical deep ConvNet usually consist of convolutional layers, pooling layers, neuron layers, and fully-connected layers[11], as shown in Fig.1. 1) Convolutional layers convolve the input image or feature maps with a linear filter, the output feature maps represent the responses of each filter. 2) Pooling layers are non-linear down-sampling layers which yield maximum or average values in each sub-region of input image or feature maps, which increase the robustness of translation and reduce the number of network parameters. 3) Activation layers apply nonlinear activations on input neurons. Common activations are sigmoid function, hyperbolic tangent function, rectified linear unit, etc. 4) Fully-connected layers compute outputs by connected to all the feature map elements of the prior layer. Forward and backward passes are two essential computations for a ConvNet: The forward pass takes the inputs and produces the outputs (solid arrows). With the computed loss, the backward pass produces the gradient with respect to the parameters and to the inputs, which are in turn back-propagated to earlier layers (dotted arrows).

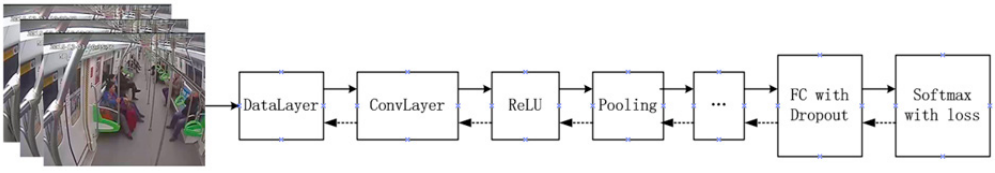

Fig. 1. The structure of standard Deep ConvNet.

2.2. Architectural details

In this paper, we utilize two classic deep ConvNets, Googlenet[9] and VGGnet[10], which have shown excellent performances in ILSVRC-2012[5], for the crowd density estimation task on real scene images.

Googlenet is a carefully crafted design ConvNet which balances the increase of depth and width of the network and the computational budget. It introduces "Inception" architecture to approximate an optimal local sparse structure in a convolutional vision network by readily available dense components. As shown in Fig. 2(a), convolution filters with different kernel sizes are paralleled within an "Inception" to get multiscale receptive fields simultaneously. The outputs of this group of filter banks are concatenated into a single output vector forming the input of the next stage. In general, Googlenet is a network consisting of modules of the "Inception" architecture stacked upon each other, with occasional max-pooling layers with stride 2 to halve the resolution of the grid. The layer parameters are denoted as "receptive field size + stride size(S)". More expatiation for this net could be found in reference[9].

VGGnet is a deep ConvNet with mostly small (3×3) convolution filters, as shown in Fig. 2(b), throughout the whole net and follow two simple design rules: (i) for the same output feature map size, the layers have the same number of filters; and (ii) if the feature map size is halved, the number of filters is doubled so as to preserve the time complexity per layer. Since the dep-conv a stack of two 3×3 convolution layer has an effective respective field of 5×5, and a stack of three equals 7×7 and so forth, a series of small non-linear rectification layers instead of a single one make the decision function more discriminative, and decrease the number of parameters at the same time. It shows that a significant improvement on the prior-art configurations can be achieved by pushing the depth to 16–19 weight layers. More expatiation for this net could be found in reference[10].

Our implementation is derived from the publicly available C++ Caffe toolbox[11]. The neural network is trained using frame samples in the train subset, which are labeled as Very-low, Low, Medium, High and Very-high based on the number of persons in the image. By quantizing the estimated crowd density, the output of the neural network is classified into 5 crowd density levels. The performance is evaluated by the classification accuracy of the test subset.

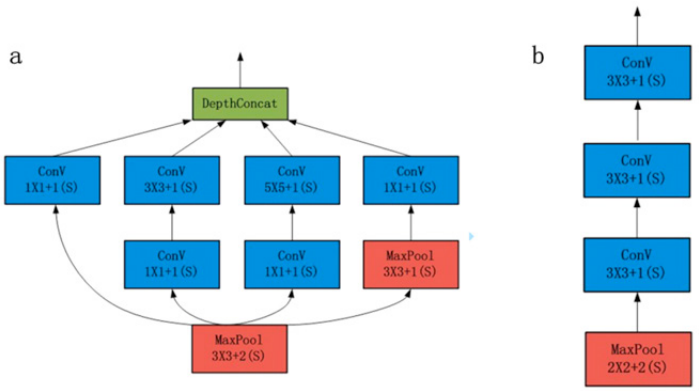

Fig. 2. (a) Inception module architecture in Googlenet (b) Deep-Conv architecture of the VGGnet.

3. Classification experiments

3.1. Dataset

The real scene dataset, Subway-carriage dataset, is used to assess the proposed method. The dataset is extracted from 31 different crowd Subway-carriage video clips, totally including 161840 RGB frames, with a unified resolution of 352×288. According to the population size, the set is quantized into five density levels: Very-low (30071 frames), Low (53961 frames), Medium (38108 frames), High (10896 frames) and Very-high (28804 frames). The whole dataset is divided into three subsets: train, validation and test, with sample sizes of 71625, 11456 and 78759 respectively. In order to properly test the generalization ability of the algorithm, we ensure that the samples of

3 subsets are from different carriage clips, so that there is no overlap between the Train, Validation and Test subset samples. Some image samples of Train and Test subset are shown in Fig. 3. It is obvious that the images of this dataset possess more complicated scenes than uniform datasets such as PETS_2009[12] because of the high perspectives, dense railings and ample advertisements.

Fig. 3. Image samples of Subway-carriage dataset. (a) Train subset (b) Test subset. (From left to right: Very-low, Low, Medium, High and Very-high).

3.2. Experiment results and analysis

In the experiment, both the Googlenet and VGGnet are fine-turned with pre-trained models from Imagenet[5]. Input images are resized to 224×224, and the number of output layer channels is set to 5 corresponding to the indexes of crowd density levels. The neural networks are trained using train subset images, which have been labeled with one of the density levels as ground truth.

The assessment criteria of experiment results contain two main types: one is for evaluation of the five grade classification accuracy directly, including the overall 5- class accuracy together with classification accuracy of each density level. The other is for the evaluation of three levels of density classification performance, namely 3-class overall accuracy. This assessment method unites the low subsets named “Very-low” and “Low” as an entirety, as well as the high subsets named “Very-high” and “High”, which makes another objective evaluation for the classification performance. Classification accuracy (CA) is defined as the ratio between the number of correctly classified samples and the number of sample number with corresponding ground truth labels.

Table 1- 4 provides the classification accuracy results for the two networks respectively. The estimated levels mostly distribute in main diagonals, which demonstrates the validity of two ConvNets methods. It can be seen that most of the misclassified samples are between the neighboring levels. The reason is that the number of people in the scenes varies little during the video, so dividing the samples both in train and test subset to different levels becomes harder as the number of the adjacent crowd level is very close. Specifically, for the 5-class estimation of Googlenet in Table 1, most of the misclassified samples distribute between “Very-low” and “Low”, as well as “High” and “Very-high” level, which could be seen as intra-class correct classifications in the 3-class estimation. Fig. 4 displays some of the test samples with population density classification results, which are marked by the predicted density level information (Green) as well as the ground-truth value (Red).

Table 1. Crowd density estimation results of Googlenet (5-Class)

Model	Density Level	Very-low	Low	Medium	High	Very-high	Overall
	Very-low	43.25%	55.17%	0.45%	1.13%	0.02%	
	Low	2.13%	89.67%	5.27%	2.83%	0.10%	
Googlenet	Medium	0	1.66%	81.12%	17.02%	0.19%	72.77%
	High	0	5.83%	3.64%	3.37%	87.16%	
	Very-high	0	0	1.98%	17.64%	80.38%	

Table 2. Crowd density estimation results of Googlenet (3-Class).

Model	Density Level	Low	Medium	High	Overall
Googlenet	Low	94.38%	3.39%	2.23%	91.73%
	Medium	1.66%	81.12%	17.22%	
	High	1.05%	2.28%	96.68%	

Table 3. Crowd density estimation results of VGGnet-16 (5-Class).

Model	Density Level	Very-low	Low	Medium	High	Very-high	Overall
VGGnet-16	Very-low	92.77%	5.42%	1.80%	0.00%	0.01%	82.66%
	Low	3.78%	79.06%	17.03%	0.12%	0.00%	
	Medium	0.34%	3.23%	92.99%	3.38%	0.05%	
	High	0	0.00%	16.11%	66.78%	17.11%	
	Very-high	0	0	29.92%	4.32%	65.76%	

Table 4. Crowd density estimation results of VGGnet-16 (3-Class).

Model	Density Level	Low	Medium	High	Overall
VGGnet-16	Low	88.84%	11.08%	0.08%	86.49%
	Medium	3.57%	92.99%	3.44%	
	High	0	27.44%	72.56%	

Table 5. Comparison on running time between the two networks (ms)

Model	Layers	Size	CPU		GPU	
			Forward pass	Backward pass	Forward pass	Backward pass
Googlenet	22	39.3MB	6118.45	6235.36	145.964	257.004
VGG-16	16	512MB	25197.2	40956.3	431.358	892.396

Comparison on running time between the two networks is also performed under the same hardware platform, which indicates the complexity of models. The testing images have unified resolution size of 224×224, with the platform of CPU (i7-4.00GHz), 24 GB RAM and GPU (NVIDIA Titan X) 12GB video RAM. As can be seen from the experimental results (Table 5), VGG-16, although its total number of network layers is less than Googlenet, occupies more model size of parameters and running time because of its internal large channel layers.

Fig. 4. Test samples with experimental results. (Predicted density level is in Green while the ground-truth value is in Red).

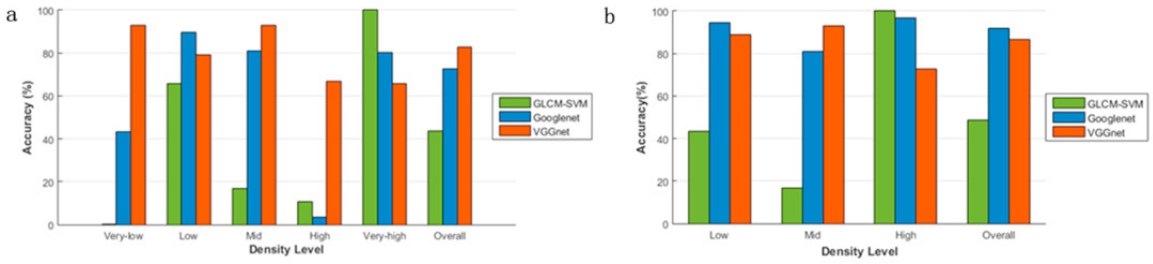

Fig. 5. Crowd density estimation result comparison of different methods. (a) 5- class; (b) 3- class.

Furthermore, we also make comparison with Yang's[13] work which applied GLCM as texture description and SVM for classification. Energy, correlation, contrast and homogeneity features are generated by GLCM in 0°, 45°, 90° and 135° directions and constituted a 16-dimensional eigenvector, which is then input into multi-class support vector machines consisting of 10 binary SVM classifiers. Multi-class SVM model is obtained by the samples in Train subset. Fig. 5 displays the comparable classification results from GLCM-SVM method and our deep-learning method on the Subway-carriage dataset. The performance reasons are also analyzed: In Yang's approach, a few characteristic parameters from normalized GLCM are regarded as features. There are often dense railings and ample advertisements which result in plenty of texture features in the test scenes (see Fig. 3), so it tends to estimate as high density under such cases. However, feature maps of ConvNets, which are acquired automatically, are independent of the textural background, which shows better generalization than handcraft features.

4. Conclusions

In this paper, we propose a new crowd density estimation method by deep ConvNets. We also build a new crowd dataset of subway carriage scenes with over 160K annotated pictures to evaluate the accuracy of cross-scene crowd density estimation methods. Experiment results provide the best accuracy of 91.73% on average and confirm that our proposed method could perform well for practical applications. Further work includes making the system more accurate in performance by adding rough people counting function.

References

1. Davies AC, Yin JH , Crowd monitoring using image processing, *Electronics & Communication Engineering Journal*, 1995. p. 37-47.
2. Marana AN, Velastin SA, Costa LF, et al. Automatic estimation of crowd density using texture, *Safety Science*, 1998, 28(3). p.165-175.
3. Li X, Shen L, Li H. Estimation of Crowd Density Based on Wavelet and Support Vector Machine. *Transactions of the Institute of Measurement & Control*, 2006, 28(3). p. 299-308.
4. Kim G, An T, Kim M. Estimation of crowd density in public areas based on neural network. *Ksii Transactions on Internet & Information Systems*. 2012, 6(9). p. 2170-2190.
5. Krizhevsky A, Sutskever I, Hinton GE. ImageNet Classification with Deep Convolutional Neural Networks. *Advances in Neural Information Processing Systems*, 2012, 25(2).
6. Girshick R, Donahue J, Darrell T, et al. Rich Feature Hierarchies for Accurate Object Detection and Semantic Segmentation. *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2014. p. 580-587.
7. Long, J, Shelhamer, E, Darrell, T. Fully convolutional networks for semantic segmentation *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2015. p. 1337-1342.
8. Fu M, Xu P, Li X, et al. Fast crowd density estimation with convolutional neural networks. *Engineering Applications of Artificial Intelligence*, 2015, 43. p. 81-88.
9. Szegedy C, Liu W, Jia Y, et al. Going deeper with convolutions. *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2015. p. 1-9.
10. Simonyan K, Zisserman A. Very Deep Convolutional Networks for Large-Scale Image Recognition. *Computer Science*, 2014.
11. Jia Y, Shelhamer E, Donahue J, et al. Caffe: Convolutional Architecture for Fast Feature Embedding. Eprint Arxiv, 2014. p. 675-678.
12. Ferryman J, Ellis A. PETS2010: Dataset and Challenge. *7th IEEE International Conference on Advanced Video and Signal Based Surveillance. IEEE Computer Society,* 2010. p. 143-150.
13. Yang J, Li J, He Y. Crowd Density and Counting Estimation Based on Image Textural Feature. *Journal of Multimedia,* 2014.