Deep People Counting with Faster R-CNN and Correlation Tracking

Zhiqiang Li[1,2], La Zhang[2], Yikai Fang[3], Jinqiao Wang[2], Huazhong Xu[1], Baocai Yin[3], Hanqing Lu[2]
[1]School of Automation, Wuhan University of Technology, Wuhan, 430070, China
[2]National Laboratory of Pattern Recognition, CASIA, Beijing, 100190, China
[3]Beijing Key Laboratory of Multimedia and Intelligent Software Technology, College of Computer Science and Technology, Beijing University of Technology, Beijing, 100124, China
lizhiqiang@whut.edu.cn, la.zhang@nlpr.ia.ac.cn, fyk@bjut.edu.cn, jqwang@nlpr.ia.ac.cn, wutxhz@163.com, ybc@bjut.edu.cn, luhq@nlpr.ia.ac.cn

ABSTRACT

Crowd counting is a key problem for many computer vision tasks while most existing methods try to count people based on regression with hand-crafted features. Recently, the fast development of deep learning has resulted in many promising detectors of generic object classes. In this paper, to effective leverage the discriminability of convolutional neural networks, we propose a method to people counting based on Faster R-CNN[9] head-shoulder detection and correlation tracking. Firstly, we train a Faster R-CNN head-shoulder detector with Zeiler model to detect people with multiple poses and views. Next, we employ kernelized correlation filter(KCF)[7] to track the people and obtain the trajectory. Considering the results of the detection and tracking, we fuse the two bounding box to obtain a continuous and stable trajectory. Extensive experiments and comparison show the promise of the proposed approach.

Categories and Subject Descriptors

I.2.10 [**Artificial Intelligence**]: Vision and Scene understanding

Keywords

People Counting; Head-shoulder Detector; Kernelized Correlation Filter

1. INTRODUCTION

People counting in videos draws a lot of attention because of its urgent demands in video surveillance, and it is especially critical for metropolis security. In recent years popular people counting methods can be classified into two categories: counting by regression[3], and counting by detection[5]. In counting by regression, after extracting some low-level image features, counting techniques learn a mapping between low-level features and people count. This counting methods can give a whole estimation of the crowd size and is more likely to work in dense scenes. However, low-level features can only provide a descriptive observation of the crowd roughly without exploring specific information of each individual.

In object detection, region-based CNN detection methods are now the main paradigm. It is such a rapidly developing area that has been proposed in the last few years, with increasingly better performance and faster processing speed. However, detection based people counting is still a challenging task for the two reasons: (1) in crowded scenes, severe occlusion and diverse crowd distributions between pedestrians is a common phenomenon, especially for large groups in confined areas. Miss and false detections are two contradictory evaluation criteria and object occlusion further deteriorate the performance; (2) the complexity of the scene causes some people to appear larger or small, fast or slow with multiple views and different illustration conditions. These problems are especially prominent in oblique camera views (where the camera looks down at an angle), which are typical of outdoor surveillance scenes.

To deal with the above problems, in this paper, we design a robust people counting approach based on Faster R-CNN[9] and correlation tracking. This approach includes three main parts: detection, tracking and counting. In the detection stage, the head-shoulder models in vertical and oblique camera views are separately trained. We adopt the Faster R-CNN detection framework with online hard example mining (OHEM)[10]. In the tracking stage, we apply kernelized correlation filter(KCF)[7] tracking algorithm to associate the people with adjacent frames. In the final stage, we fusion the outputs of detection and tracking, and trajectory of head-shoulder is extracted by the result of fusion. A person is counted if his or her head-shoulder has a continuous trajectory.

2. RELATED WORK

Counting by detection Stewart and Andriluka[11] defined a Hungarian loss function and introduced LSTM to train an end-to-end system for crowd counting. Their system addressed the challenge of detecting multiple partially occluded instances by decoding a variable number of outputs from rich intermediate representations of an image. Similar to us, [12] counted people based on detection flow, but they

ICIMCS '16, August 19-21, 2016, Xi an, China

DOI: http://dx.doi.org/10.1145/3007669.3007745

only applied a boosted human detector to collect detector responses which couldn't generate a fine motion curve. Our approach introduces the state-of-the-art KCF[7] tracker to associate the detection result, it can deal with more serious occlusions.

Object detection Object detection is one of the oldest and most fundamental problems in computer vision. Early work Barinova[1] employed Hough voting with a codebook, but still required multi-stage pipelines and complex tuning. The ACF pedestrian detection method[4] used adaptive contour feature for human detection and segmentation.

Follow after Fast R-CNN[6], Faster R-CNN[9] was proposed to reduce the computational burden of proposal generation. It consisted of two modules. The first was a Region Proposal Network (RPN) which took an image (of any size) as input and outputted a set of rectangular object proposals, each with an objectness score. It was a fully convolutional network for generating object proposals that would be fed into the second module. The second module was the Fast R-CNN object detection network whose purpose was to refine the proposals. The ingenious means was to share the same convolutional layers for the RPN and Fast R-CNN object detection network. Now the image only passed through the CNN once to obtain object detection result.

The recent work[8] selected hard examples for training deep networks. Similar to our applied approach[10], all these methods based their selection on the current loss for each datapoint. Akin to OHEM[10] approach, Loshchilov[8] investigated online selection of hard examples for mini-batch SGD methods.

Correlation tracking After the success of [2], correlation filter-based tracking framework had shown to be significantly efficient for robust tracking. By taking advantage of kernel trick, correlation filter was supposed to be more powerful. Correlation filter had proved to be competitive with far more complicated approaches, but used only a fraction of the computational power, at hundreds of frames-per-second.

3. OVERVIEW

There are two representative camera views(Oblique or vertical, as shown in Fig.2) in video surveillance. Due to the difference of the perspective between oblique and vertical camera views, the people's appearance shows a big difference at different situations. So, it is hard to judge a person of different perspective through a unified detection model. In this paper, we apply Faster R-CNN[9] to train two head-shoulder detection models with the oblique or vertical perspective datasets respectively. Then, we add online hard example mining (OHEM)[10] in the Faster R-CNN training process to decrease false detections. To the best of our knowledge, the person in one frame maybe is detected miss or false, but in other frames is correct. After getting bounding box of crowd by Faster R-CNN detection with OHEM, kernelized correlation filter(KCF)[7] tracker is used to obtain the track of crowd. Finally, we fusion the detection and tracking results to obtain a continuous and stable trajectory for people counting. Fig.1 illustrates the overall framework of the proposed method.

3.1 Faster R-CNN Based Head-shoulder Detection

Stewart[11] and Gao[4] detected head and whole body for human detection. Due to the information of head is not enough to detect, The performance of head detector is limited and only can use in the fixed scene where it trained in. The detector that Gao proposed may fail when it comes to the pedestrian crowd, where cameras are generally not in a bird-view. An example of pedestrians in an ordinary surveillance camera is shown in Fig.1. It has two visible characteristics: (1) pedestrian images in the surveillance videos have different scales due to perspective distortion; (2) due to severe occlusions, heads and shoulders are the main cues to judge whether there exists a pedestrian at each position. The body parts of pedestrians are not reliable for human annotation. Taking these characteristics into account, we detect people by detecting their head-shoulder, which is reliable and sufficient. Faster R-CNN is a powerful detector, one of it's speciality is that it can detect objects which have different scales. This can perfect solve the problem of scale changing. So, we summarize the key point of the Faster R-CNN framework next. Readers can go to the original paper [9] for more technical details.

In the RPN, the last shared convolutional layers of a pre-trained network(Zeiler and Fergus model(ZF)) are followed by a 3 x 3 convolutional layer. This corresponds to mapping a receptive field or large spatial window(e.g., 171 x 171 for ZF) in the input image to a low-dimensional feature vector(256-d for ZF with ReLU following). Two sibling 1 x 1 convolutional layers are then added for classification and regression branches of all spatial windows. To deal with detected head-shoulder in different scales and aspect ratios, anchors are introduced in the RPN. Each anchor is associated with a scale and an aspect ratio. According to the size and shape of head-shoulder in the dataset, we use 3 scales(64^2, 128^2, and 256^2 pixels) and 1 aspect ratios(1:1) in vertical camera views. In oblique camera views, we use 5 scales(16^2, 32^2, 64^2, 128^2, and 256^2 pixels) and 1 aspect ratios(1:1), leading to k = 5 anchors at each sliding-window location. The RPN can be trained end-to-end by stochastic gradient descent (SGD) for both classification and regression branches. For the whole framework, we concern with both the RPN and Fast R-CNN modules since the shared convolutional layers. Ren et al.[9] proposed an approximate joint training strategy which was trained the RPN and Fast R-CNN end-to-end as they were independent. Note that the output of the RPN is actually has a great influence on the performance of the Fast R-CNN. In this paper, we adopt this method training the RPN and Fast R-CNN simultaneously what reduces the training time by about 25-50% comparing with alternating training. For the joint training, other hyper-parameters are not carefully chosen for our particular dataset.

3.2 Online Hard Example Mining

Shrivastava et al.[10] implemented the online hard example mining algorithm (OHEM) in the Fast R-CNN[6] detector. The Experiment result shows that OHEM improves the mAP of Fast R-CNN from 65.7% to 69.8% on VOC12. In like manner, we apply Faster R-CNN framework[9] combined with OHEM to detect crowd. Based on Fast R-CNN, Faster R-CNN must be trained RPN module and Fast R-CNN module integrally. So, we only can handle the RoIs from RPN output during stochastic gradient descent (SGD) iteration. More specifically, the OHEM proceeds as follows. For an input image at SGD iteration t, we first compute a conv feature map using the conv network. Then the RoI

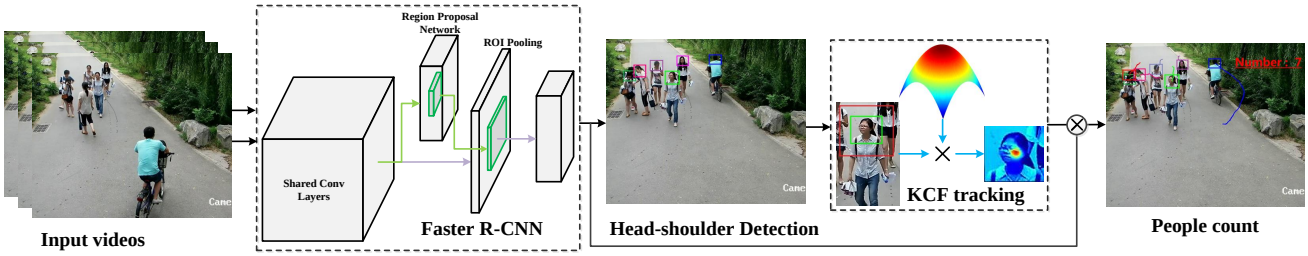

Figure 1: Overview of the proposed deep people counting method.

network uses this feature map and the all the input RoIs, instead of a sampled mini-batch, to do a forward pass. Recall that this step only involves RoI pooling, a few fc layers, and loss computation for each RoI. The loss represents how well the current network performs on each RoI. Hard examples are selected by sorting the input RoIs by loss and taking examples for which the current network performs worst. Most of the forward computation is shared between RoIs via the conv feature map, so the extra computation needed to forward all RoIs is relatively small. Moreover, because only a small number of RoIs are selected for updating the model, the backward pass is no more expensive than before.

3.3 KCF Based Head-shoulder Tracking

We track each head-shoulder based on kernelized correlation filters(KCF)[7] tracker, which achieves very impressive results on Visual Tracker Benchmark. In KCF[7], Henriques et al. assumed that the cyclic shifts version of base sample was able to approximate the dense samples over the base sample. Suppose that we have a one-dimensional data $\mathbf{x} = [x_1, x_2, ..., x_n]$. It has an intriguing property that all the circulant matrices can be expressed as below:

$$\mathbf{X} = \mathbf{F}^H \mathbf{diag}(\mathbf{F}\mathbf{x})\mathbf{F} \tag{1}$$

where $\mathbf{F}$ is known as the DFT matrix, which transforms the data into Fourier domain, and $\mathbf{F}^H$ is the Hermitian transpose of $\mathbf{F}$. The goal of KCF tracker training is to find a function $f(\mathbf{z}) = \mathbf{w}^T\mathbf{z}$ that minimizes the squared error over samples x_i and their regression targets y_i,

$$\min_{\mathbf{w}} \sum_i (f(\mathbf{x}_i) - y_i)^2 + \lambda ||\mathbf{w}||^2$$

where λ is a regularization parameter that controls overfitting, as in the SVM. As mentioned earlier, the minimizer has a closed-form,

$$\mathbf{w} = (X^T X + \lambda I)^{-1} X^T \mathbf{y} \tag{2}$$

where the data matrix X has one sample per row $\mathbf{x}_i$, and each element of $\mathbf{y}$ is a regression target y_i. I is an identity matrix. Substituted by Eqn.1, we have the solution:

$$\hat{\mathbf{w}}^* = \frac{\hat{\mathbf{x}}^* \odot \hat{\mathbf{y}}}{\hat{\mathbf{x}}^* \odot \hat{\mathbf{x}} + \lambda} \tag{3}$$

where $\hat{\mathbf{x}} = \mathbf{F}\mathbf{x}$ denotes the DFT of $\mathbf{x}$, and $\hat{\mathbf{x}}^*$ denotes the complex-conjugate of $\hat{\mathbf{x}}$.

In Fig.1, we draw the tracking trajectory based counting method in which tracking trajectory extracted from detection responses are used to get the crowd number. Counting based on tracking trajectory is more robust to the interference from occlusion, false and miss detections.

3.4 People Counting

A stable fusion scheme is proposed to match every tracked bounding box with each detected bounding box in its following frames. We use bounding box of crowd getting by Faster R-CNN[9] detection model with OHEM[10] to initialize kernelized correlation filter(KCF)[7] tracker for a "new" pedestrian had not been matched. We extract motion curve according to the results of the KCF tracker. In our situation, we make sure that a tracked box is proper matched whether it has an Intersection-over-Union (IoU) overlap higher than 0.7 with any detected box. When one tracked box is continuously not matched in next 10 frames, we believe that the initialization box is a false detection. Notice that we don't remove the box and curve once it has no match in the following frame, but instead we keep it active for 10 frames. This makes our motion curve building robust for the case when the pedestrian in a minority of frames is not detected. It is important that the OHEM makes Faster R-CNN detection results almost no false detection. Besides, a threshold is introduce. If the length(of time) of a curve is beyond the threshold(1 second i.e. 25 frames), the pedestrian corresponding to the curve will be counted.

4. EXPERIMENTS

We evaluate the performance of people counting on two camera view dataset: Vertical and Oblique. The vertical camera view dataset include 3 videos in different scenes. We choose 2500 frames in 2 videos for Faster or ACF training and the rest video for testing. The oblique camera view dataset is splitted into two parts. 5 eight-minute long video sequences out of 5 scenes are treated as training sets. The test set has 3 ten-minute long video sequences from 3 different scenes. The details of both dataset are shown in Tab.1. and some instances are shown in Fig.2. From left to right: dataset name, number of frame, resolution, frame per second, minimum and maximum number of people, total number of people instances.

Table 1: People counting dataset.

Dataset	Frame	Resolution	FPS	Count	Total
`Vertical`	5527	352 x 288	25	0-7	4965
`Oblique`	39594	1280 x 720	25	0-40	363498

We use the same metrics as conventional works for evaluating counting performance: mean absolute error $mae = E(|k_j - k'_j|)$ and mean squared error $mse = E((k_j - k'_j)^2)$, where k_j and k'_j are the true number and the estimated number of objects in frame j, respectively. k'_j is computed

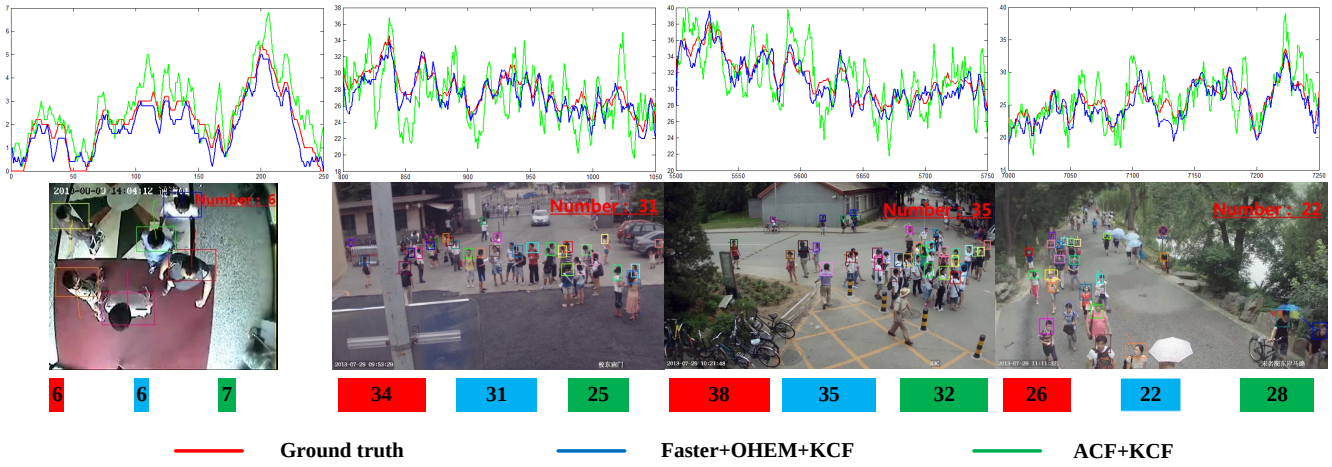

Figure 2: Our counting results on the Vertical View and Oblique View crowd counting dataset. (Top) result curve for each test scene, where X-axis represents the frame index and Y-axis represents the counting number. (Down) one sample selected from the corresponding test scene. Best viewed in color.

as the sum of estimated densities over the whole image.

In order to show the performance of the proposed approach, we compare Faster R-CNN[9] detector training with KCF[7] tracker to the baseline ACF[4] detector with KCF tracker approach in this section. We also run experiments to show whether Faster R-CNN detector with online hard example mining (OHEM)[10] is effective. We perform comparing the crowd counting only detection and detection combine with tracking. The quantitative results of people counting on our vertical and oblique camera view dataset are shown in Tab.2 and Fig.2.

Table 2: Comparison results with traditional approaches.

	Vertical View		Oblique View	
Method	mae	mse	mae	mse
ACF	1.51	3.41	6.60	61.43
Faster	1.05	1.71	3.93	21.15
Faster+OHEM	1.04	1.76	3.29	15.22
ACF+KCF	1.01	1.62	4.75	31.08
Faster+KCF	0.58	0.58	2.61	9.33
Faster+OHEM+KCF	**0.49**	**0.49**	**1.94**	**5.21**

5. CONCLUSIONS

In this paper, an effective people counting method based on tracker combine with detection is presented. Unlike most previous methods which estimate the whole crowd size from one frame, we achieve robust people counting by taking advantage of Faster R-CNN based head-shoulder detection and correlation tracking results. An online hard example mining scheme is applied to remove false alarms, and we fuse the detection and tracking results for accurate people counting. A prototype system based on this method demonstrate its robustness to noise and complex changes. Our experimental demonstrated the superiority of the proposed approach on two datasets of crowded scenes.

6. ACKNOWLEDGMENTS

This work was supported by 863 Program 2014AA015104, and National Natural Science Foundation of China 61273034, and 61332016.

7. REFERENCES

[1] O. Barinova, V. Lempitsky, and P. Kohli. On detection of multiple object instances using hough transform. *In CVPR*, 2010.

[2] D. S. Bolme, J. R. Beveridge, B. A. Draper, and Y. M. Lui. Visual object tracking using adaptive correlation filters. *In CVPR*, pages 2544–2550, 2010.

[3] K. Chen, S. Gong, T. Xiang, and C. C. Loy. Cumulative attribute space for age and crowd density estimation. *In CVPR*, pages 2467–2474, 2013.

[4] W. Gao, H. Ai, and S. Lao. Adaptive contour features in oriented granular space for human detection and segmentation. *In CVPR*, 2009.

[5] W. Ge and R. Collins. Marked point processes for crowd counting. *In CVPR*, pages 2913–2920, 2009.

[6] R. B. Girshick. Fast r-cnn. *In ICCV*, 2015.

[7] J. F. Henriques, R. Caseiro, P. Martins, and J. Batista. Highspeed tracking with kernelized correlation filters. *In PAMI*, 2015.

[8] I. Loshchilov and F. Hutter. Online batch selection for faster training of neural networks. *arXiv preprint arXiv:1511.06343*, 2015.

[9] S. Ren, K. He, R. B. Girshick, and J. Sun. Faster r-cnn: towards real-time object detection with region proposal networks. *In NIPS*, pages 91–99, 2015.

[10] A. Shrivastava, A. Gupta, and R. Girshick. Training region-based object detectors with online hard example mining. *arXiv preprint arXiv:1604.03540*, 2016.

[11] R. Stewart and M. Andriluka. End-to-end people detection in crowded scenes. *In NIPS*, 2015.

[12] J. Xing, H. Ai, L. Liu, and S. Lao. Robust crowd counting using detection flow. *In ICIP*, 2011.