LETTER

A Statistical Method for Counting Pedestrians in Crowded Environments

Gwang-Gook LEE[†a)], ***Student Member and*** **Whoi-Yul KIM**[†], ***Nonmember***

SUMMARY We propose a statistical method for counting pedestrians. Previous pedestrian counting methods are not applicable to highly crowded areas because they rely on the detection and tracking of individuals. The performance of detection-and-tracking methods are easily degraded for highly crowded scene in terms of both accuracy and computation time. The proposed method employs feature-based regression in the spatiotemporal domain to count pedestrians. The proposed method is accurate and requires less computation time, even for large crowds, because it does not include the detection and tracking of objects. Our test results from four hours of video sequence obtained from a highly crowded shopping mall, reveal that the proposed method is able to measure human traffic with an accuracy of 97.2% and requires only 14 ms per frame.

key words: *pedestrian counting, crowd analysis, video surveillance*

1. Introduction

Counting pedestrians provides useful information for the design, management and monitoring of large public areas. Previous computer vision methods for pedestrian counting are primarily based on the detection and tracking of humans. Humans are first located by either segmenting foreground blobs [1]–[3] or by detecting individuals based on their appearance or shape [4], [5]. Subsequently, the identified humans are tracked in consecutive frames in order to count the number of people going past. The previous detection-and-tracking methods are not adequate for applications in a large public area because reliable detection and tracking of individuals in a large crowd is not an easy task. Moreover, as the population increases, the accuracy decreases and additional computation time is needed.

We propose an alternative method for pedestrian counting. The proposed method is a statistical approach based on feature-based regression. Pedestrians are counted using the relationship between the image features and the number of people who pass by. To count these people, rather than the static humans, a spatiotemporal analysis is performed. Because our method uses a statistical approach that does not involve detection or tracking, it requires minimal computation. Moreover, its performance remains stable for varying traffic levels.

Manuscript received November 24, 2010.
Manuscript revised February 28, 2011.

[†]The authors are with Hanyang University, Seoul, 133–791 Korea.
a) E-mail: gglee@vision.hanyang.ac.kr

DOI: 10.1587/transinf.E94.D.1357

2. Proposed Method

2.1 Determining Pedestrian Numbers from Image Features

For counting pedestrians, we set up a measurement line, or a *virtual gate*, in the video frame (Fig. 1 (a)). The virtual gate was placed orthogonal to the predominant direction of the pedestrians. Observing the image pixels on the virtual gate over time creates a spatiotemporal image, with two coordinates corresponding to the time, t, and the linear coordinate along the virtual gate, s. In the spatiotemporal image, s connects a pixel location (x, y) to the corresponding pixel on the virtual gate (Fig. 1 (b)).

The number of people passing the virtual gate is acquired by counting the number of people in the spatiotemporal image. Conventional detection techniques are not applicable in a spatiotemporal image because human shapes may undergo severe deformation (Fig. 1 (b)). As indicated by arrows, human shapes can be: (A) slanted or bent because of the non-orthogonal directions of movement; (B) elongated or compressed due to slow or fast moving speeds; or (C) occluded by other individuals in heavy traffic.

Rather than trying to detect individuals in the spatiotemporal image, we counted pedestrians statistically using feature-based regression, which has been successfully used

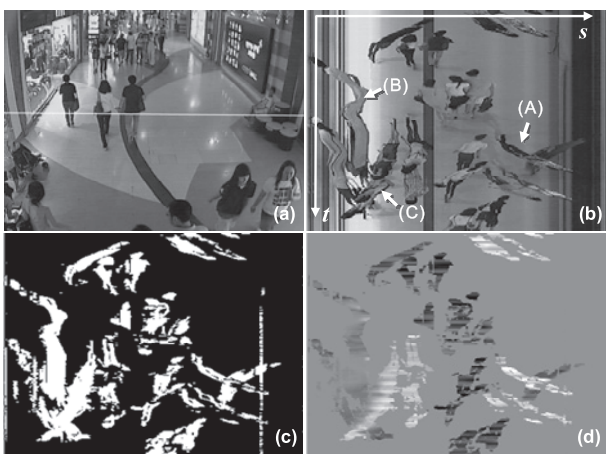


Fig. 1 (a) Input video frame. The white line indicates the virtual gate. (b) Spatiotemporal image; (c) Foreground map; (d) Motion vector map (bright and dark colors correspond to motion directions. Gray corresponds to a zero motion vector).

in crowd size estimation [6]–[10]. Feature-based regression assumes that the number of people in an image is highly correlated to the total amount of features extracted from the image. Thus, crowd size is measured by extracting the image features and setting a relationship between the amount of features and the number of people. Foregrounds, edges, or textures are commonly used features. In this study, only foreground pixels were used as image features. We avoided using any edge or texture features because of their sensitivity to lighting changes, image resolution and noise levels. Furthermore, to obtain separate pedestrian counts for opposite directions, we simultaneously examined motion vectors as image features.

We extracted the foreground pixels and motion vectors using conventional methods [11], [12]. To avoid unnecessary computation, foreground segmentation was performed only for the pixels on the virtual gate. We examined motion vectors for every two pixels on the virtual gate, and subsequently interpolated them. To compute the motion vectors, we used 8×8-sized blocks and three levels of hierarchy. As a result of the feature extraction, we created a foreground map, $fg(t, s)$, and a motion vector map, $v(t, s)$, for the spatiotemporal image. In the foreground map, $fg(t, s)$ is equal to one when a pixel, s, on the virtual gate belongs to the foreground at time, t, otherwise it is zero. Similarly, the motion vector map, $v(t, s)$, contains the motion vector for a pixel, s, on the virtual gate at time, t. Figures 1 (c) and (d) give examples of the foreground and motion vector maps, respectively.

The upward direction is that which moves away from the camera, with the downward direction moving toward it. To provide pedestrian counts for both the upward and downward directions separately, we introduced a passing direction, $k \in \{+1, -1\}$. This direction was defined as $+1$ when the inner product of the motion vector and the normal vector of the virtual gate line was equal to or greater than zero and otherwise as -1.

Based on the assumption that the number of people is proportional to the amount of image features, the pedestrian count for a direction, k during the time from t_i to t_j was obtained by accumulating extracted image features using the following formula:

$$F_k(t_1, t_2) = \sum_{t=t_1}^{t_2} \sum_{s=1}^{N} \alpha \cdot \rho(s) \cdot fg(t, s) \cdot \delta(k, d(t, s)). \quad (1)$$

In (1), N is the number of pixels on the virtual gate and $d(t, s)$ is the passing direction for pixel s at time t. A delta function, $\delta(i, j)$, (which equals one if $i = j$, but is otherwise zero) was used to accumulate only the image features in the same direction. Hence the summation of $fg()$ multiplied by $\delta()$ resulted in the number of foreground pixels of the same direction that occurred on the virtual gate during times t_1 and t_2.

The number of foreground pixels was then converted to the number of pedestrians by introducing two scaling factors, $\rho(s)$ and α. To determine $\rho(s)$, we modeled humans as rectangles whose sizes varied linearly with the vertical image coordinates. The rectangle size for each pixel position was easily calculated by annotating the human size manually at several locations and interpolating them. Thus, pixel s, $\rho(s)$, was set as $1/W(s) \cdot H(s)$ where $W(s)$ and $H(s)$ were the width and height of the rectangle. Because the area covered by a human is generally smaller than its bounding box, another scaling factor, α, was employed to fill this gap. This was determined using a short video sequence with a known number of pedestrians.

2.2 Improving Feature Accumulation to Manage Distortions in the Spatiotemporal Domain

As previously mentioned, different moving speeds and directions influence feature observation in the spatiotemporal domain. For example, a slow-moving person produces more foreground pixels by taking a longer time to pass through the virtual gate. To account for the different moving speeds and directions of travel, the feature accumulation in (1) was modified to (2):

$$F_k(t_1, t_2) = \sum_{t=t_1}^{t_2} \sum_{s=1}^{N} \alpha \cdot \rho(s) \cdot \|v(t, s)\| \cdot |\cos\theta_v| \cdot fg(t, s) \cdot \delta(k, d(t, s)). \quad (2)$$

In this equation, the motion magnitude is multiplied so as to include the different moving speeds in the measured pedestrian count. To consider only the motion components that contribute to pass through the virtual gate, the motion vector was projected onto a normal vector in the virtual gate; thus, θ_v is the angle between the motion vector, $v(t, s)$, and the normal vector.

Although different pedestrian moving speeds and directions can be accounted for using motion vectors, Eq. (2) cannot accurately measure high crowd levels. When a scene is crowded, occlusions occur between individuals that make foreground pixels less observable. Hence, the pedestrian count calculated using (2) tended to underestimate the actual count as the scene became more crowded. To compensate for these inaccuracies, a nonlinear regression was applied to the count estimate:

$$F'_k(t_1, t_2) = a \cdot F_k(t_1, t_2)^b, \quad (3)$$

where a and b are the regression parameters determined during the initial training. Because feature observation loss is incremented with increasing crowd levels we chose a function of the power form for the regression. The measurement duration $t_2 - t_1$ was fixed at 60 seconds for all of our experiments because we used the feature accumulation results of (2) as input in the nonlinear regression. We employed the gradient descent method as the optimization algorithm for parameter learning.

2.3 Advantages of Statistical Analysis in the Spatiotemporal Domain

The pedestrian count explained by (1) expresses the basic concept of feature-based regression. The number of people who pass by a measurement line is measured by counting the number of foreground pixels. The proposed method extracts the image features from the virtual gate line and accumulates them for sequential frames. This incremental accumulation makes the counting process as the same with an image analysis in the spatiotemporal domain.

This statistical analysis in the spatiotemporal domain brings some advantages to the proposed method. First, it greatly reduces the computational burden, as the number of pedestrians is obtained by extracting the image features and accumulating them, rather than through the use of detection or tracking. Furthermore, instead of analyzing a whole video frame, only the pixels on the virtual gate line are processed. Second, the performance of the proposed method remains stable for highly crowded scenes. The accuracy of previous detection and tracking methods decreased as the number of people in a scene increased. The statistic basis of the proposed method enables the accuracy and processing times to remain more stable, regardless of the crowd size.

3. Experiments and Discussion

For the evaluation, we used an experimental dataset comprised of four hours of video sequences. The video sequences were acquired at two different locations in the most crowded shopping mall in Korea; the video was captured at 15 fps with a frame size of 352×240 (Fig. 2). We recorded the video sequences at two different times (10:00–11:00 AM and 7:00–8:00 PM) because the characteristics of pedestrian traffic in the complex building differ depending on the time of day (early versus late).

As the ground truth for evaluation, the number of people passing the virtual gate was counted manually for every minute. The initial 20 minutes of each sequence was employed as a training set in order to determine the parameters (i.e., α, a and b), and the remaining 40 minutes of the video sequences were used for evaluation. The same coefficients were maintained across all experiments for the video sequences obtained from the same camera.

The relative accuracy of the proposed method ranged from 95% to 100%, averaging 97.20% (Table 1). The algorithm required only 13–16 ms to process one frame on an Intel Pentium IV 2.67 GHz PC. Figures 3 and 4 show the graphical evaluation results for Videos 1 and 2. The accuracy remained stable, in spite of the significant differences in traffic levels between the video sequences at the different times (a minimum of 200, with a maximum of 1,200 for 40 minutes).

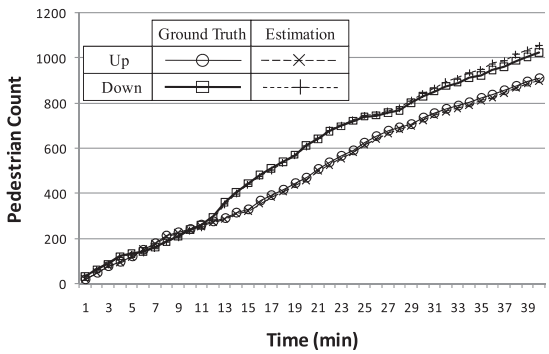
The method proposed in this paper is similar to crowd estimation methods [6]–[10]. Crowd size estimation methods compute the number of people in a crowd or measure crowd density using image features. Cho et al. proposed a method of estimating crowd level using a neural network and counts of foreground/edge pixels [7]. Their method

Table 1 Evaluation results.

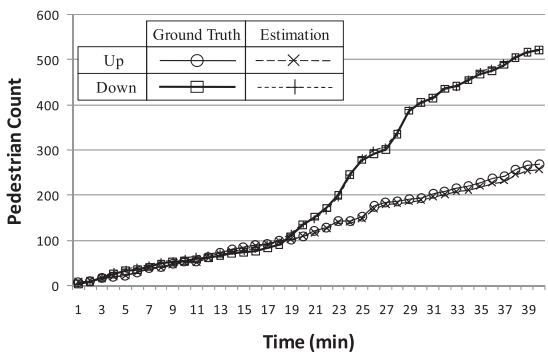
		Upward			Downward		
		Ground Truth	Estimation	Accuracy	Ground Truth	Estimation	Accuracy
Video 1	10 AM	268	257	95.98	522	522	100
		Processing Time per Frame: 12.67 ms					
	7 PM	910	901	98.98	1025	1054	97.12
		Processing Time per Frame: 13.73 ms					
Video 2	10 AM	813	785	96.58	211	201	95.22
		Processing Time per Frame: 15.30 ms					
	7 PM	1194	1238	96.32	1215	1284	97.44
		Processing Time per Frame: 15.68 ms					

(a) (b)

Fig. 2 Examples of test sequences: (a) Video 1 and (b) video 2.

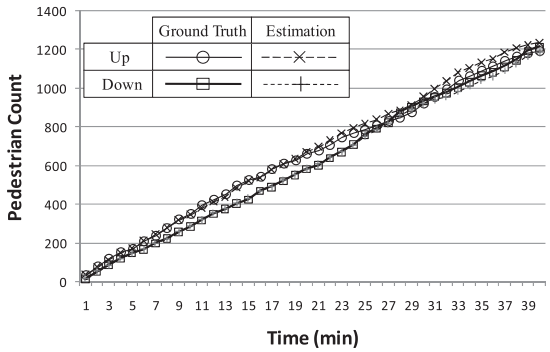


(a)

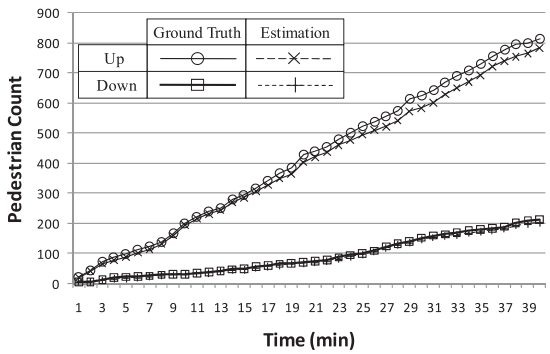


(b)

Fig. 3 Evaluation results for video 1.



(a)



(b)

Fig. 4 Evaluation results for video 2.

resulted in 93.89% accuracy for sequences of CCTV images. Celik et al. also used foreground pixels [9]. Their results only differed from the ground truth by less than or equal to one person for 86–92% of test images. In [10], crowd counting was performed using various image features and Gaussian process regression. For an outdoor video sequence of 50 minutes that contained, at most, 46 people in a frame, their estimate was within 2–3 people of the ground truth for 91–98% of the test images.

The major difference between the crowd size estimation and the proposed method is that the proposed method accumulates image features over sequential frames. Crowd size estimation methods determine the number of people that exist in a scene. All people produce the same count in crowd counting, regardless of movement, and the total pedestrian count cannot be determined using crowd size estimation. For example, in [10], the dataset contains 49,885 people; however, this count does not coincide with the actual number of pedestrians passing through the area, as it is the total number of people counted from the different images. Because the proposed method uses an analysis over time, it can count the number of people who pass by a certain point over a given period of time.

Compared to previous methods based on human detection and tracking, the proposed method is much faster and provides similar or higher accuracy. A blob tracking method that uses a top-view camera [1] showed a precision of 100% and a recall of 95% with a processing speed of 12 *fps*. However, the test sequence used a total of only 21 pedestrians with, at most, three appearing on the scene at the same time. The blob tracking method in [2] showed an absolute error of less than one person for groups of people from 2 to 11. Even though this method achieved a real time performance processing speed of up to 25 frames per second, the processing speed reduced to 11 *fps* as the number of groups in the scene increased. Zhao et al. employed elliptical human models to detect pedestrians from a foreground area and to track the people located [3]. In their experiments, 5.3% of the trajectories could not be tracked properly. Their method was able to process only two frames per second on a 2.8 GHz PC. In [4], a human detection method using hierarchical template matching attained a detection rate of 90 to 95% with a small number of false alarms; however, only two to five frames were processed per second on a Pentium-M 2 GHz machine. Another detection-based method using shape and motion cues [5] achieved detection rates of approximately 90%, with a small number of false alarms, for videos obtained at a railway station. With the aid of graphics hardware for fast computation, a real-time performance (20 fps) was obtained. Conversely, the proposed method required only approximately 14 ms per frame. The low complexity is beneficial for complex environments with a large number of cameras because it helps the algorithm either run on an embedded system or process multiple inputs on a single machine.

One limitation of the proposed method is that the accuracy could decrease if the camera angle is too far away from the frontal-view or too close to the side-views. The side-view produces more occlusions because pedestrians walk along the main direction of a corridor in most cases. This form of occlusion easily occurs, even when the scene is not very crowded, and cannot be analyzed using the non-linear regression given by Eq. (3). The proposed method also could produce over-estimates when objects larger than humans (e.g., cars or carriers) pass by because it does not distinguish between objects, relying only on low level image features.

4. Conclusions

We proposed a novel method for counting pedestrians. Unlike previous methods, which count individuals using detection and tracking, our approach applies a feature-based regression in the spatiotemporal domain. Our method requires less computation, while providing similar or higher accuracy. Moreover, the performance of the proposed method remains stable for video sequences with large crowds. Hence, the proposed method is highly applicable to systems in complex environments.

References

[1] B. Antić, D. Letić, D. Ćulibrk, and V. Crnojević, "K-means based segmentation for real-time zenithal people counting," Proc. 16th IEEE Int. Conf. Image Processing, pp.2565–2568, Cairo, Nov. 2009.
[2] P. Kilambi, E. Ribnick, A.J. Joshi, O. Masoud, and N.

Papanikolopoulos, "Estimating pedestrian counts in groups," Computer Vision and Image Understanding, vol.10, pp.43–59, April 2008.

[3] T. Zhao, R. Nevatia, and B. Wu, "Segmentation and tracking of multiple humans in crowded environments," IEEE Trans. Pattern Anal. Mach. Intell., vol.30, no.7, pp.1198–1211, July 2008.

[4] Z. Lin and L.S. Davis, "Shape-based human detection and segmentation via hierarchical part-template matching," IEEE Trans. Pattern Anal. Mach. Intell., vol.32, no.4, pp.604–618, April 2010.

[5] C. Beleznai and H. Bischof, "Fast human detection in crowded scenes by contour integration and local shape estimation," Proc. IEEE Conf. Computer Vision and Pattern Recognition, Vienna, June 2009.

[6] S.A. Velastin, J.H. Yin, A.C. Davies, M.A. Vicencio-Silva, R.E. Allsop, and A. Penn, "Automated measurement of crowd density and motion using image processing," Proc. Int. Conf. on Road Traffic Monitoring and Control, pp.127–132, 1994.

[7] S.-Y. Cho, T.W.S. Chow, and C.-T. Leung, "A neural-based crowd estimation by hybrid global learning algorithm," IEEE Trans. Syst. Man Cybern., vol.29, no.4, pp.535–541, Aug. 1999.

[8] D. Kong, D. Gray, and T. Hai, "Counting pedestrians in crowds using viewpoint invariant training," British Machine Vision Conference, 2005.

[9] H. Celik, A. Hanjalic, and E.A. Hendriks, "Towards a robust solution to people counting," Int. Conf. Image Processing, pp.2401–2404, Atlanta, Oct. 2006.

[10] A.B. Chan, Z.-S.J. Liang, and N. Vasconcelos, "Privacy preserving crowd monitoring: Counting people without people models or tracking," Proc. Computer Vision and Pattern Recognition (CVPR), pp.1–7, Anchorage, June 2008.

[11] C. Stauffer and W.E.L. Grimson, "Adaptive background mixture models for real-time tracking," Proc. Int. Conf. on Computer Vision (ICCV), vol.2, pp.246–252, 1999.

[12] B. Lukas and T. Kanade, "An iterative image registration technique with an application to stereo vision," Proc. Int. Joint Conf. on Artificial Intelligence (IJCAI), pp.674–679, 1981.