# 14

# Measurement of Pedestrian Traffic Using Feature-based Regression in the Spatiotemporal Domain

Gwang-Gook Lee and Whoi-Yul Kim
*Hanyang University,*
*Republic of Korea*

## 1. Introduction

Measurement of pedestrian traffic in public areas (e.g., stations, airports, shopping malls or complex buildings) provides valuable information. For safety management, congestions can be detected to prevent accidents in their early stage by monitoring the pedestrian traffic continuously. Knowing the number of people working in a large building may help to when designing evacuation plans. For marketing purposes, value assessments of shopping areas can be achieved based on traffic data because higher pedestrian traffic is directly linked to more sales. In building management, pedestrian traffic data can be utilized to optimize the number and working hours of staff. Power savings can be achieved by adjusting air-conditioning and heating based on pedestrian traffic.

Over the last decade, various computer vision methods have been studied to automatically measure the pedestrian traffic. One popular approach to pedestrian traffic measurement is the use of top-view cameras. In this approach, a camera is mounted vertically at the top of a gate or over a region of interest. Because of the superior viewpoint of the camera, pedestrians do not obscure each other in video frames. Hence the problem of pedestrian traffic measurement may be solved easily by detecting moving objects using foreground segmentation and tracking the detected blobs (Sexton et al., 1995; Kim et al., 2003). However, these methods fail when a number of people move close or slightly touch each other creating a single blob. Chen et al. resolved this problem by comparing the area of detected moving object with the area of one person to estimate the number of people in the blob (2006). Velipasalar *et al*. employed two-level hierarchical tracking to deal with pedestrians of complex movements interacting with each other (2006).

Pedestrian traffic also can be studied by detecting humans using standard surveillance cameras that do not require a specific viewpoint. Similar to top-view camera based methods, some of these methods perform foreground segmentation to distinguish moving objects. However, for oblique camera angles, multiple pedestrians easily appear as merged blobs in a video frame. The detected foreground blobs are segmented into individuals by modeling humans as ellipsoids (Zhao et al, 2004, 2008) or rectangles (Liu et al, 2005; Beleznai et al., 2006) cooperating with the known camera geometry. Based on based on their shapes and apperances, humans can also be detected directly from image frames without separating out foreground blobs. Viola *et al*. detected humans using appearance and motion information

together in a boosting sceheme (2003). Dalal and Triggs used histograms of oriented gradients as features to describe human shapes (2005). Detection of whole human bodies often suffer due to occlusions in dense crowds. To resolve miss-detection due to occlusions, only upper body shapes (Sidla et al., 2006) or contours around heads (Yuk et al., 2006) may be used in detection. Part-based detection methods have been studied extensively (Wu and Nevatia, 2005; Lin et al., 2007) to improve detection performance in dense crowds. Once pedestrians are detected, they are then tracked to analyze their movement and to collect traffic data. Bayesian inference (Zhao et al., 2004 & 2008), Kalman filter (Sidla, 2006) or other trackers (Yuk et al, 2006) have been used to track individual pedestrians.

Even though various efforts have been made, existing methods are not suitable for measuring pedestrian traffic in large public areas. The top-view camera based method shows good performance with relatively low computational burden. However the top-view camera based methods cannot be applied to exisitng CCTV systems because they require a dedicated camera system of specific angles. Currently, most of large buildings have their own video surveillance system but they have oblique views to enable wide coverage of cameras and to deliver better scene understanding to human operators. Installation of additional video camera system only for pedestrian traffic measurement would be a great burden. Unlike the top-view camera based methods, the detection-based methods can be applied to ordinary CCTV cameras with an oblique view. However, the computational complexity of detection-based methods is relatively high in general. This complexity is a restriction to real systems where the computational power is low and the number of cameras is large. Moreover, the computation time tends to increase as the scene gets more complex with large crowds because more pedestrians should be detected and tracked.

A pedestrian traffic measurement method should satisfy the following requirements to be useful in a practical system that covers a large public area:

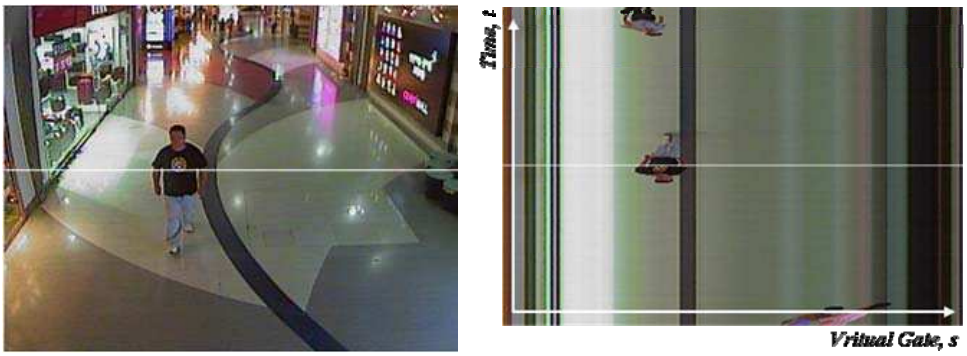
- Low computational complexity: The computational complexity of the algorithm should be as low as possible. Real-time execution on a PC is not sufficient for large systems because tens or hundreds of cameras are often used in a complex building. When such a large number of cameras is involved, the algorithm should be able to process a number of CCTV inputs on a single computer or the method should be executable on an embedded system with a computational power that is much lower than that of a standard PC.
- Compatibility with existing system: Most large buildings have their own video surveillance systems. The pedestrian traffic measurement method should make use of existing surveillance systems. To achieve this compatibility, traffic measurement algorithms solely rely on video camera input, and not require other kinds of input such as range data. This also implies that the algorithms should not be constrained by camera angle.
- Stability under high traffic: In public places, such as railway stations or shopping malls, the number of people can be large. Hence the method should be able to measure pedestrian traffic successfully not only for small numbers of people, but also for large crowds. Moreover, the computation time of the method should not increase for larger numbers of people.

An alternative method for measuring pedestrian traffic is introduced in this chapter. The method is a statistical approach which uses feature-based regression. The feature-based regression is widely used for crowd size estimation. In crowd size estimation, the number of people or the level of crowdedness in an image frame is measured by examining image

features. As image features, foreground pixels (Velastin *et al*., 1994; Celik *et al*., 2006), edges (Cho *et al*., 1999), textures (Manara *et al*., 1999) or combinations of various features (Kong *et al*., 2006; Chan *et al*., 2008) are employed. Based on the extracted image features, the count of people or the level of crowdedness is measured by linear relation (Velastin *et al*., 1994), a neural network (Cho *et al*., 1999; Kong *et al*., 2006), a SVM classifier (Xiohua *et al*., 2006) or a Gaussian process regression (Chan *et al*., 2008).
In a similar manner to the crowd size estimation, the size of pedestrian traffic is estimated from the amount of image features. That is, the traffic is measured by setting a relation between image features and the number of pedestrians. To count passing people rather than static humans, the analysis is performed in a spatiotemporal domain rather than an image domain. Because it is a statistical method which is applied in the spatiotemporal domain, it requires very low computation, its performance remains stable under high traffic and it is also less sensitive to camera viewpoints.

## 2. Overview

The basic concept underlying the pedestrian traffic measurement method can be easily understood from Fig. 1. In the video frame, a measurement line, called a *virtual gate*, is set up as in Fig. 1 (a). Here $s$ connects a pixel location $(x, y)$ to the corresponding pixel on the virtual gate. Fig. 1 (b) is an example of a spatiotemporal image.




(a) (b)

Fig. 1. (a) A virtual gate to measure pedestrian traffic size, (b) a spatiotemporal image created using the virtual gate

Observing the image pixels on the virtual gate over time can be interpreted as examining a spatiotemporal image whose two coordinates correspond to time $t$ and the linear coordinate along the virtual gate $s$, respectively. When a person passes the virtual gate, his body shape is produced in the spatiotemporal image as in Fig. 1 (b). The spatiotemporal image, which is obtained over a certain period of time, contains the images of people who passed the gate during that period. Hence the pedestrian traffic or the number of people passing the virtual gate, can be acquired by counting the number of people in this spatiotemporal image.
When counting pedestrians in spatiotemporal images, we cannot use conventional detection or segmentation techniques because human shapes suffer severe distortions in the spatiotamporal images. Fig. 2 shows some examples of these distortions. In Fig. 2 (a) the

shape of objects are slanted because they changed direction while passing the virtual gate. Fig. 2 (b) gives an example of size variations of people in the spatiotemporal image. Because the two people on the left side moved very slowly, their shapes are elnogated resulting in a larger image size that of the people on the right side. Also, in Fig 2. (c), part of some people are occluded by others and their whole body shapes cannot be seen.

(a) (b) (c)

Fig. 2. Distortions occur in the spatiotemporal image. (a) Slanted shape occurred due to a pedestrian passing to the left. (b) Elongated shape caused by very slow pedestrian movement. (c) Occlusions due to dense crowd.

Because of these problems, counting pedestrians as individuals in spatiotemporal images using conventional detection or segmentation method is not feasible. Rather than trying to detect individuals, a statistical method is adopted to count pedestrians as a whole from image features.

Fig. 3 shows the block diagram of the traffic flow measurement method. As shown in the figure, image features are extracted first followed by feature integration process to measure pedestrian traffic. Foreground pixels and motion vectors are extracted as image features. In the traffic flow measurement step, the foreground pixels are accumulated along the virtual gate over continuous frames to calculate pedestrian traffic. In the feature accumulation, a feature normalization process is employed to account for size variation of the human images caused by perspective projection. Also, different moving speeds of individuals are considered to adapt different motions of pedestrians. Because occlusions due to a dense crowd yields under-estimation of pedestrian traffic size, the accumulated feature size is compensated to deliver an accurate estimate of pedestrian traffic.

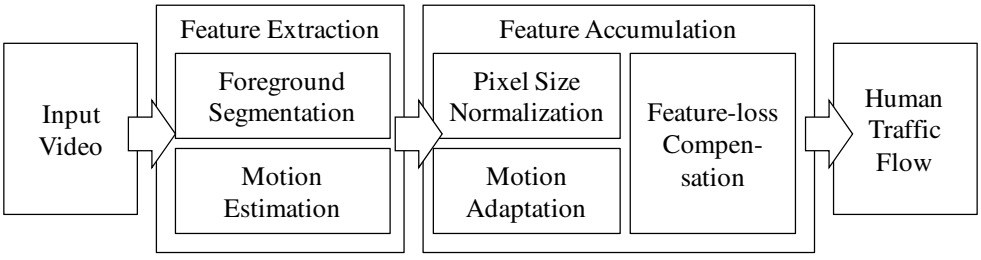



Fig. 3. Block diagram of the traffic flow measurement method

## 3. Feature extraction

### 3.1 Foreground segmentation

As the image feature for human traffic estimation, regions of moving objects are first isolated. We avoided using other features such as edges or textures because of their sensitivity to noise level or lighting changes. Foreground segmentation is achieved by comparing an input frame with a reference background. In creating and updating the reference background, a background modeling method proposed by Stauffer and Grimson (1999) is employed with a light modification.

In the Stauffer and Grimson's method, each pixel in a video frame is modeld by mixture of Gaussian distributions. An update of the background model is performed incrementally by the online K-means algorithm given by (1):

$$\Theta_{k,t}(x) = (1-\alpha)\cdot\Theta_{k,t-1}(x) + \alpha\cdot\Lambda\left(I_t(x), \Theta_{k,t-1}(x)\right). \tag{1}$$

In (1), $\Theta_{k,t}$ and $I_t$ are model $k$ and an observation at time $t$ for a pixel $x$. Each model updates its parameter based upon a local estimate $\Lambda(I_t(x), \Theta_{k,t\text{-}1}(x))$. The learning rate $a$ is a small constant which determines the learning speed of the background model. Because the model parameters are updated incrementally using online $K$-means, the background model is adaptable to scene changes such as lighting variation or new background objects.

The incremental update of the model parameter in (1) can be thought as a pixel observation process that uses a temporal window of length $L = 1/a$. The underlying assumption of the model update is that the background pixel occurs most frequently in this temporal window. Hence the model update process tries to find the dominant mode by estimating its density using online clustering. However, such assumption is often violated when high traffic of pedestrians occurs constantly in a scene. For example, if pedestrians pass the observation area continuously leaving only a small time window for backgroud pixels, foreground pixels may occupy the majority of the pixel statistics resulting in a defective background models as shown in Fig. 4.

Because the traffic flow measurement method introduced in this chapter is designed for use in public areas with high traffic rates, the background modeling method must be able to cope with the defective backgrounds with high traffic. To resolve this problem, another assumption is made which is that background pixels are not only the most frequent but also are static. Hence, to avoid creating an erroneous background model, the learning rate of each pixel is adjusted by examining its static level. If a pixel is not static at a time, a lower learning rate is applied because the pixel might belong to a foreground object.

(a) (b) (c)

Fig. 4. (a) a scene with continuous pedestrian movements, (b) method clear background model under low human traffic, (c) a defected background model due to continuous movements of humans.

To identify static pixels, we first define its activity of a pixel as (2). In (2), $A(x, t)$ represents the activity of a pixel x at time t and $I_d(x, t)$ is interframe difference which is defined as $|I(x, t) - I(x, t\text{-}1)|$. Hence, the activity is decided as the maximum value between the interframe difference and the activity of previous frame decreased by a constant ratio $\beta$.

$$A(x,t-1) = \max\left(I_d(x,t), \beta \cdot A(x,t-1)\right) \tag{2}$$

By comparing its activity to a given threshold level $T_{act}$, each pixel is classified as static or non-static. Fig. 5 shows an example of a static pixel classification where (a) is the input frame and (b) is the classification result. In Fig. 5 (b), static pixels and non-static pixels are represented in black and white, respectively. Pixels around moving objects show large activity values and are labelled as non-static pixels. We used 0.2 for $\beta$ and 40 for $T_{act}$.

Even though pixels around moving objects show large activity values, pixels inside a large object or nearly static objects might be labeled as static pixels as shown in Fig. 5 (b). Hence the labeled result is expanded using a morphological operation. The size of the window used for the morphological operation is determined as the expected size of a human at each pixel location, which will be explained in Section 5.1. Fig. 5 (b) shows the result of the morphological operation in which gray pixels indicate non-static pixels reclassified from static pixels.

(a) (b)

Fig. 5. Distinguishing static and non-static pixels: (a) input video frame; (b) black pixels and white pixels correspond to static and non-static pixels, respectively. Gray pixels are expanded from white pixels by morphological operations.

Once static and non-static pixels are distinguished, a low learning rate is used for the model to update to the non-static pixels. The lower learning rate $a_l$ is set to be 10 times lower than the regular learning rate. Controlling the learning rate according to the activity of the pixel can be thought as changing background model update according to the history of the pixel. When a pixel shows static characteristic, its background model is updated by a general Gaussian mixture model. If a pixel is determined to be non-static, its update rate is significantly reduced making the background model similar to a static reference.

To reduce computational complexity, the background model is generated and maintained only for the pixels at the virtual gate. The activity value is computed only for regions around the virtual gate to control the learning rate of the background models. Because computation of activity value is quite simple compared to the maintenance of background models, the overall computational load is significantly reduced.

### 3.2 Motion estimation

To measure pedestrian traffic separately in different directions, the moving directions must be observed. Motion information is also required to obtain the exact traffic size because the size of an object in the spatiotemporal image varies according to the time taken the object to pass the virtual gate. For these reasons, a motion vector is chosen as a feature to examine moving directions and speeds of pedestrians. Hence, motion vectors are employes as image features too.

Motion vectors are obtained by a coarse-to-fine estimation of optical flow using pyramids (Bouguet, 1999). Because the estimation of optical flow includes a differential equation, which is solved iteratively, it introduces computational complexity. Hence, to reduce computation, motion vectors are computed for every two pixels on the line of the virtual gate and then interpolated.

Fig. 6 illustrates an example of motion vector computation. The computed motion vectors are displayed in different colors. The green lines indicate motion vectors that pass the virtual gate in the upward direction and the red lines represents motion vectors in the downward direciton. The lengths of the lines coincide with the maginitudes of the motion vectors.

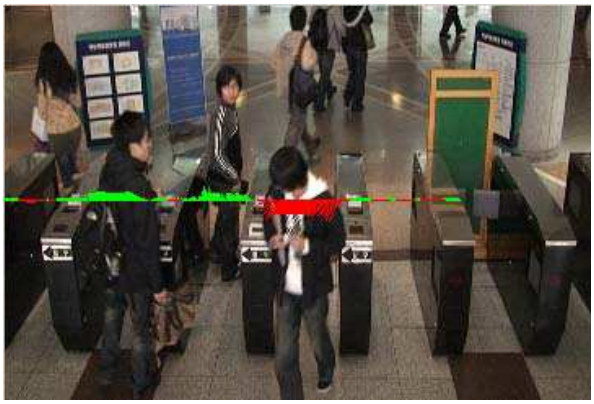


Fig. 6. An example of extracted motion vectors

## 4. Feature accumulation

### 4.1 Estimating human traffic flow from image features

As a result of the feature extraction described in the previous section, a foreground map $fg(t, s)$ and a motion vector map $v(t, s)$ for the spatiotemporal image are created. 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 $t$. Fig. 7 gives an example of the foreground map and the motion vector map. Fig. 7 shows an example of feature extraction.

For convenience, the traffic away from the camera is refered to as the upward direction and the opposite as the downward direction. To determine the traffic flow size of humans for upward and downward directions separately, the direction of traffic flow $k \in \{+1, -1\}$ is introduced. The direction of traffic flow is defined as +1 when the inner product of the motion vector and the normal vector of the virtual gate line is equal to or greater than zero and vice versa.

(a) (b) (c)

Fig. 7. Feature extraction results: (a) spatiotemporal image, (b) foreground map, (c) motion vector map

Based on the assumption that the number of people in the image is proportional to the amount of image features, the traffic flow size for a direction $k$ during the time from $t_i$ to $t_j$ is obtained by integrating the extracted image features by using the following formula.

$$F_k(t_1,t_2)=\sum_{t=t_1}^{t_2}\sum_{s=1}^{N}\alpha\cdot\rho(s)\cdot fg(t,s)\cdot\delta\big(k,d(t,s)\big). \tag{3}$$

In (3), $N$ is the number of pixels on the virtual gate and $d(t, s)$ is the direction of the traffic flow for a pixel $s$ at time $t$. A delta function $\delta(i, j)$ (which equal one if $i = j$, but otherwise is zero) is used to integrate the image features from one direction only. Hence, the summation of $fg()$ multiplied by $\delta()$ gives us the amount of foreground pixels that have the same direction and occur between time $t_1$ and $t_2$.

The amount of image features (i.e., foreground pixels) is converted into the number of pedestrians by introducing two scaling factors $a$ and $\rho(s)$ in (3). To determine $\rho(s)$, humans are modeled as rectangles with sizes that vary linearly with vertical image coordinates as shown in Fig. 8. The rectangle size for each pixel position can be easily calculated by annotating the human size manually at several locations and interpolating. Then, for a pixel $s$, $\rho(s)$ is set to $1/W(s)\cdot H(s)$ where $W(s)$ and $H(s)$ are the width and height of a rectangle. Because the area covered by a human is generally smaller than its bounding box, another scaling factor $a$ is employed to fill this gap. The scaling factor $a$ can be determined using a short video sequence with a known number of pedestrians.


Fig. 8. Pixel size normalization

### 4.2 Adaptation to motions of pedestrians

As mentioned in Section 2.1, different moving speeds and directions of people influences to feature observation in the spatiotemporal domain. For example, a person who moves slowly produces larger traffic estimate by taking a longer time to pass through the virtual gate. To deal with the different moving speeds and direction of pedestrians, the feature accumulation in (3) is modified to (4).

$$F_k(t_1,t_2)=\sum_{t=t_1}^{t_2}\sum_{s=1}^{N}\alpha\cdot\rho(s)\cdot\left\|\mathbf{v}(t,s)\right\|\cdot\left|\cos\theta_{\mathbf{v}}\right|\cdot fg(t,s)\cdot\delta\left(k,d(t,s)\right). \tag{4}$$

In this equation, the motion magnitude is multiplied to include the moving speeds of people in the traffic flow. Also, to consider only the motion components that contribute to passing by the virtual gate, the motion vector is projected onto the normal vector of the virtual gate where $\theta_v$ is the angle between the motion vector $v(t, s)$ and the normal vector of the line of the virtual gate.

Fig. 9 shows some examples of this pixel counting process. Fig. 9 (a), (b) and (c) are examples of the test sequence in which one person passes the gate. Fig. 9(d), (e) and (f) are the results of pixel counting obtained by integrating $F()$. Therefore, the feature integration results approached one because the pixel count was normalized by the average area of one person using $a$ and $\rho(s)$. Note that the moving speeds of (a) and (b) are different (21 frames vs. 16 frames, respectively). Also, the viewpoints are different in (a) and (c). However, the traffic flow obtained by (4) approach to one in all three sequences proving its robustness to changes in camera angle and different speeds of pedestrians.

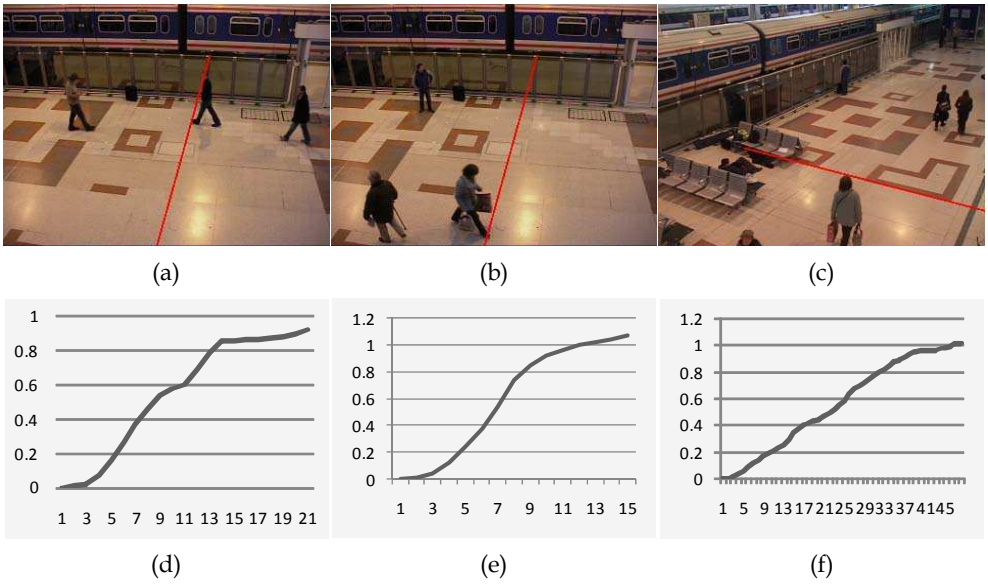



Fig. 9. Examples of traffic flow estimation for one person

### 4.3 Compensating feature loss due to a dense crowd

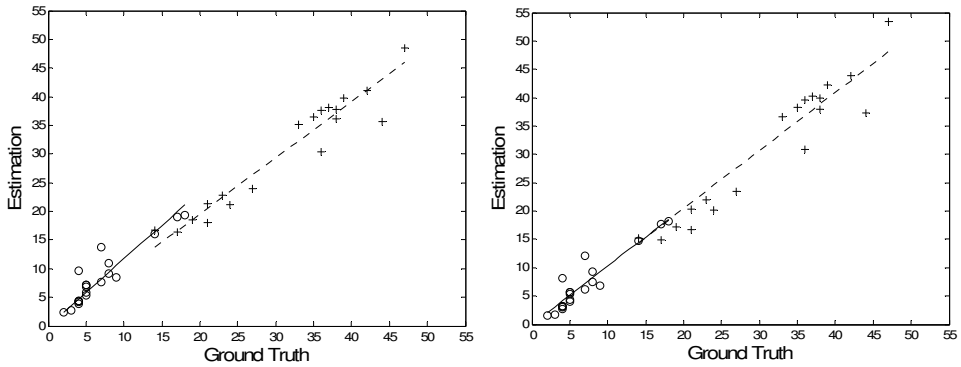
Although different pedestrian speeds and directions can be handled using motion vectors, the pedestrian traffic given by (4) cannot handle the problems caused by a dense crowd. When a scene is crowded, occlusions take place between individuals that make foreground pixels less observable. Hence, the traffic estimates obtained from (4) tend to underestimate the actual traffic value when the scene becomes crowded.

To compensate for the loss of feature observation due to occlusion, the traffic flow computed by (4) is compensated using nonlinear regression,

$$F_k^{'}(t_1, t_2) = a \cdot F_k(t_1, t_2)^b. \tag{5}$$

where $a$ and $b$ are the regression parameters that are learned during initial training. Because the loss of feature observation increases as the crowd level in the scene grows, a function of the power form is chosen for the regression. The measurement duration $t_2$-$t_1$ must be fixed because the feature integration result of (4) is used as input to the nonlinear regression. It was set to 60 seconds in our experiments. For parameter learning, the gradient descent method is employed as the optimization algorithm.

Fig. 10 shows an example of nonlinear regression used to compensate for the under-estimation in a dense crowd. In the graphs, 40 sample data are displayed. The points were obtained by estimating the sizes of the pedestrian traffic in one minute video segments. The $x$-axis represents the actual number of people passing in a video segment and the $y$-axis indicates the estimated human traffic size for the same video segment. Sample data were obtained from the two different video sequences containing low and high pedestrian traffic, which are represented as "o" and "+" in the graphs. Samples from each video were fitted linearly, as illustrated by solid and dotted lines for better comparison. The estimated flow size (obtained by (4)) versus ground truth is given in Fig. 10 (a). As shown in the figure, the slope of the fitted line for the video containing high pedestrian traffic is lower than that of the video for low pedestrian traffic. This indicates that the pedestrian traffic was underestimated for the high traffic segments. On the other hand, we can see that the lines nearly coincide in Fig. 10 (b) where the flow estimation results were adjusted by nonlinear regression as in (5).




(a) (b)

Fig. 10. Result of the nonlinear regression to compensate for feature loss. Graphs of ground truth vs. estimates (a) without and (b) with feature loss compensation as in (5).

## 5. Experiments

For the evaluation, an experimental dataset of 4 hours of video sequences was used. The video data were acquired at two different locations of the most crowded shopping mall in Korea. Fig. 11 shows an example of the test video of two different locations ((a) Video 1 and (b) Video 2). Because the characteristics of traffic flow in the shopping mall differ early and late in the day, we recorded video sequences at two different times (10:00 – 11:00 AM and 7:00 – 8:00 PM).

(a) (b)

Fig. 11. Test sequences for the evaluation: (a) Video 1 and (b) Video 2.

As the ground truth for evaluation, the number of people passing the virtual gate was counted manually each minute. The initial 20 minutes of each sequence were employed as a training set to calculate parameters (i.e., $a$, a and $b$ in (4) and (5)) and the remaining 40 minutes of the video sequences were used for evaluation. The same coefficients were maintained across experiments for video sequences obtained from the same camera.
Table 1 summarizes the evaluation results. The relative accuracy of the proposed method was 95% to 100% and 97.20% on average. The processing speed of the proposed method reached 70 frames/second on an Intel Pentium IV 2.67 GHz PC. Figs. 12 and 13 provide evaluation results for Video 1 and Video 2 in a graphical representation. It should be noted that the accuracy remained stable in spite of the significant differences of traffic levels between video sequences of different times (200 at minimum and 1,200 at maximum in for 40 minutes).

| | | Upward | | | Downward | | |
|---|---|---|---|---|---|---|---|
| | | Ground Truth | Estimation | Accuracy | Ground Truth | Estimation | Accuracy |
| Video 1 | 10 AM | 268 | 257 | 95.98 | 522 | 522 | 100 |
| | 7 PM | 910 | 901 | 98.98 | 1025 | 1054 | 97.12 |
| Video 2 | 10 AM | 813 | 785 | 96.58 | 211 | 201 | 95.22 |
| | 7 PM | 1194 | 1238 | 96.32 | 1215 | 1284 | 97.44 |

Table 1. Evaluation results

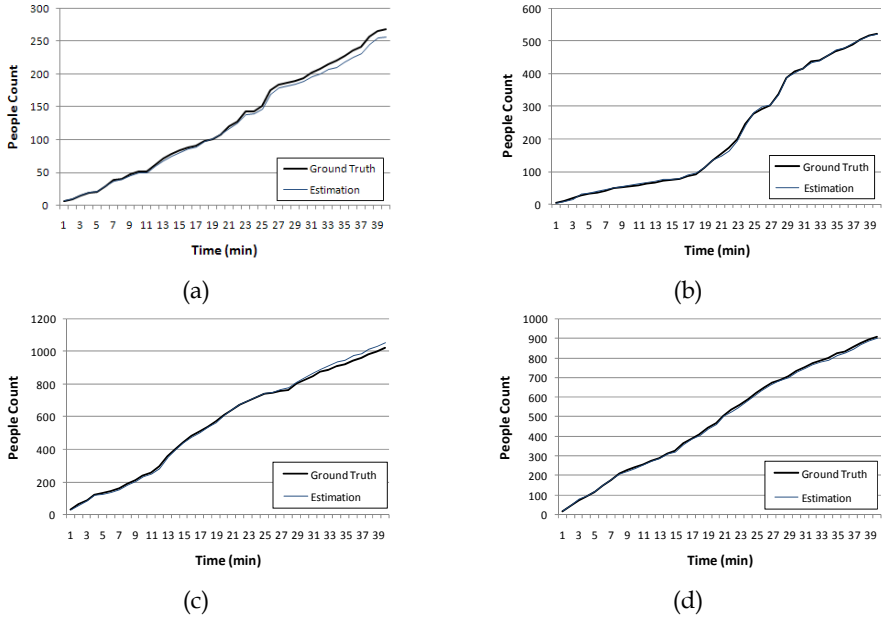



Fig. 12. Evaluation results for Video 1: (a) 10 AM and upward, (b) 10 AM and downward, (c) 7 PM and upward, (d) 7 PM and downward

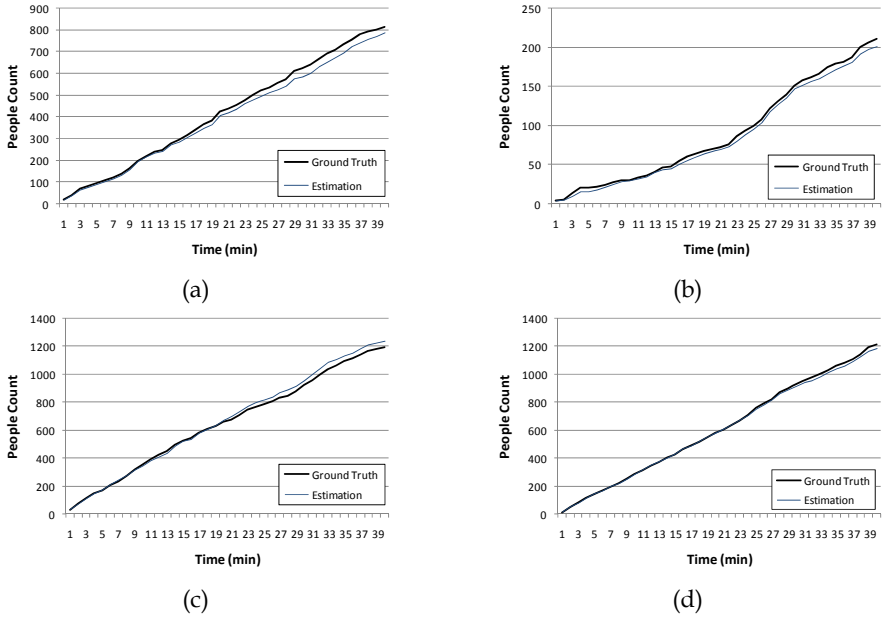



Fig. 13. Evaluation results for Video 2: (a) 10 AM and upward, (b) 10 AM and downward, (c) 7 PM and upward, (d) 7 PM and downward

## 6. Discussions

The method introduced in this chapter is a statistical approach that estimates the size of human traffic flow from the amount of image features. This basic concept, estimating traffic size from image features, is discovered from (3), which integrates foreground pixels of the same directions. This process in (3) is similar to that used for crowd size estimation, but the proposed method performs an online update. Instead of gathering image features from a whole frame, the proposed method extracts image features only from the virtual gate line and accumulates them over sequential frames. This incremental accumulation makes the traffic measurement process the same as image analysis in the spatiotemporal domain.

The use of statistical analysis in the spatiotemporal domain yields some advantages in the pedestrian traffic measurement method. First, the computational burden of the method is greatly reduced. Human traffic is measured by extracting image features and accumulating them, not by detection or tracking. Instead of analyzing an entire video frame, only pixels on the line of the virtual gate are reqruied to be processed, requiring much less computation. Hence the proposed method incurs much lower computational complexity than previous methods. Second, the performance of the method remains stable in high traffic areas in terms of both accuracy and computation time. Because previous methods are based on the identification of objects, as the number of people in a scene is increased, the accuracy of previous methods decreases while the computation time increases. In the proposed method, the measurement process is not related to individual objects; hence the same execution time can be maintained regardless of the number of people in the scene. Because of the statistical method relies on training, it shows good performance even for scenes with high traffic.

When comparing to previous methods, the human traffic measurement method introduced in this chapter provides similar or even a higher accuracy with much less computation. It has been reported that the top-view camera method proposed by Chen et al. (2006) showed an accuracy of 100% with simple movements of a few people. However, the accuracy was reduced to 85% for pedestrians with complex moving patterns. The frame rate of their method varied from 10 Hz to 30 Hz depending on the number of people in the scene. The detection-based by Sidla *et al.* (2006), which used a head-shoulder shape for human detection, counted passing people with 98% accuracy with a 15 Hz frame rate. However, they applied a linear regression to the result of human detection because the automatic count was overestimated. Without the aid of linear regression, the accuracy fell to 85%–90%. Since all of their test sequences contain only one hour of video, it is not guaranteed that the same linear regression could be applied to other video showing a different level of pedestrian traffic. Zhao *et al*. (2008) employed elliptical human models to detect pedestrians from foreground area and to track located humans. Because Zhao et al. evaluated the accuracy of tracking rather than pedestrian count, their accuracy was relatively low as 62%. Their method also could process about 2 frames per second on a 2.8GHz Pentium IV PC.

Besides the accuracies, it is also be noted that the statistical method is tested for video sequences of highly different traffic levels. Previous methods rarely tested for videos of different crowd levels. Hence the stability in varying level of pedestrian traffic, which is important for practical use, is not guranteed for the previous methods. On the contrary, the

statistical method have been verified using two video sequences of mild and heavy traffic. Even though the test sequences showed huge differences in the level of crowdedness (six times at most), both the computation time and accuracy of the statistical method remained stable.

The main drawback of the statistical method is that it is based on training. The problems of training based methods are twofold. First they require human intervention during initial training process. This could be an obstacle applying the same method to multiple different locations. Another problem is that the performance could be dependent to the amount of training data and might not be guaranteed to a new input which is far from training data. Celik *et al.* (2006) proposed a pixel counting method for crowd size estimation that does not include training phase. It is expected that a similar concept could be applied to the statistical method to relieve its shortcomings.

## 7. Conclusions

In this chapter, a statistical method for measuring human traffic flow was introduced. Unlike previous methods that tried to count individuals by detection and tracking, the statistical method count pedestrians based on image features. Because it is a statistical method which does not include time consuming detection and tracking, it requires much smaller computation compared to previous methods. Through experiments on video data from real environments, it is shown that the proposed method gives similar or higher accuracy compared to previous methods even with the low computational cost. Because it does not rely on a specific camera viewpoint unlike blob-based methods, the method can be applied to existing CCTVs with oblique views. The low complexity, high accuracy and flexibility in viewpoints make the proposed method highly applicable to real systems.

## 8. References

Beleznai, C.; Fruhstuck, B. & Bishof, H. (2006). Human tracking by fast mean shift mode seeking. Journal of Multimedia 1(1): 1.

Bouguet, J. (1999). Pyramidal implementation of the Lucas Kanade feature tracker description of the algorithm. Technical Report, Intel Corporation, Microprocessor Research Labs.

Celik, H.; Hanjalic, A. & Hendriks, E. (2006). Towards a robust solution to people counting, *Proceedings of International Conference on Image Processing*, pp. 2401-2404

Chan, A.; Liang, Z. & Vasconcelos, N. (2008). Privacy preserving crowd monitoring: Counting people without people models or tracking. *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, Anchorage. 30: 40-50.

Cho, S.; Chow, T. & Leung, C. (1999). A neural-based crowd estimation by hybrid global learning algorithm, *IEEE Transactions on Systems, Man, and Cybernetics--Part B: Cybernetics*, 29(4): 535.

Dalal, N.; Triggs, B., Rhone-Alps, I. & Montbonnot, F. (2005). Histograms of oriented gradients for human detection, *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, vol. 1, pp. 8860893.

Kim, J.-W.; Choi, K.-S., Choi, B.-D., Lee, J.-Y. & Ko, S.-J. (2003). Real-Time System for Counting the Number of Passing People Using a Single Camera. *Pattern Recognition*, Springer Berlin / Heidelberg. 2781: 466-473.

Kong, D.; Gray, D & H. Tao. (2006). A viewpoint invariant approach for crowd counting, Proceedings of International Conference on Pattern Recognition, pp. 1187-1190.

Lin, Z.; Davis, L., Doermann, D. & DeMenthon D. (2007). Hierarchical part-template matching for human detection and segmentation,*Proceedings of International Conference on Computer Vision*, pp. 1-8.

Liu, X.; Tu, P., Rittscher, J., Perera, A., & Krahnstoever, N. (2005). "Detecting and counting people in surveillance applications." *Proceedings of Advanced Video and Signal Based Surveillance*, pp. 306-311.

Marana, A.; Costa, L. & Velastin, S. (1999). Estimating crowd density with Minkowski fractal dimension, *Proceedings of IEEE International Conference on Acoustics, Speech, and Signal Processing*, vol. 6, pp. 3521-3524.

Sexton, G.; Zhang, X., Redpath, G. & Greaves, G. (1995). Advances in automated pedestrian counting, Proceedings of European Convention and Security and Detection, pp. 106-110.

Sidla, O.; Lypetskyy, Y. Brandle, N. & Seer, S. (2006). Pedestrian detection and tracking for counting applications in crowded situations, *Proceedings of International Conference on Video and Signal Based Surveillance*, pp. 70-70.

Stauffer, C. & Grimson, W. (1999). Adaptive background mixture models for real-time tracking, Proceedings of IEEE Conference on Computer Vision and Pattern Recognition, pp. 246-252.

Thou-Ho, C.; Tsong-Yi, C. & Chen, Z.-X. (2006). An Intelligent People-Flow Counting Method for Passing Through a Gate, *Proceedings of International Conference on Robotics, Automation and Mechatronics*, pp. 1-6.

Velastin, S.; Yin, J., Davies, A., Vicencio-Silva, M., Allsop, R. & Penn. A. (1994). Automated measurement of crowd density and motion using image processing, Proceedings of International Conference on Road Traffic Monitoring and Control, pp. 127-132.

Velipasalar, S.; Ying-Li, T. & Hampapur, A.(2006). Automatic Counting of Interacting People by using a Single Uncalibrated Camera, Proceedings of International Conference on Multimedia and Extp, pp. 1265-1268.

Viola, P.; M. Jones & D. Snow. (2005). Detecting pedestrians using patterns of motion and appearance, *Proceedings of International Journal of Computer Vision* pp. 153-161.

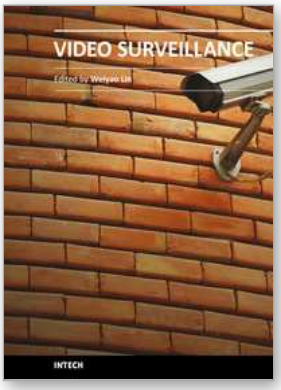
Wu, B. & Nevatia, R. Detection of multiple, partially occluded humans in a single image by bayesian combination of edgelet part detectors,Proceedings of International Conference on Computer Vision, pp. 90-97.

Xiaohua, L.; Lansun, S. & Huanqin, L. (2006). Estimation of crowd density based on wavelet and support vector machine, *Transactions of the Institute of Measurement & Control*, Vol. 28, pp. 299.

Yuk, J.; Wong, K, Chung, F & Chow, K. (2006). Real-time multiple head shape detection and tracking system with decentralized trackers, *Proceedings of Intelligent Systems Design and Applications*, pp. 382-389.

Zhao, T. & R. Nevatia (2004). Tracking multiple humans in crowded environment, *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, pp. II-406 – II-415.

Zhao, T. & Nevatia, R. and Wu, B. (2008). Segmentation and tracking of multiple humans in crowded environments, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 30, pp. 1198-1211.




**Video Surveillance**
Edited by Prof. Weiyao Lin

ISBN 978-953-307-436-8
Hard cover, 486 pages
**Publisher** InTech
**Published online** 03, February, 2011
**Published in print edition** February, 2011

This book presents the latest achievements and developments in the field of video surveillance. The chapters selected for this book comprise a cross-section of topics that reflect a variety of perspectives and disciplinary backgrounds. Besides the introduction of new achievements in video surveillance, this book also presents some good overviews of the state-of-the-art technologies as well as some interesting advanced topics related to video surveillance. Summing up the wide range of issues presented in the book, it can be addressed to a quite broad audience, including both academic researchers and practitioners in halls of industries interested in scheduling theory and its applications. I believe this book can provide a clear picture of the current research status in the area of video surveillance and can also encourage the development of new achievements in this field.

**How to reference**
In order to correctly reference this scholarly work, feel free to copy and paste the following:

Gwang-Gook Lee and Whoi-Yul Kim (2011). Estimation of Human Traffic Flow Using Feature-based Regression in the Spatiotemporal Domain, Video Surveillance, Prof. Weiyao Lin (Ed.), ISBN: 978-953-307-436-8, InTech, Available from: http://www.intechopen.com/books/video-surveillance/estimation-of-human-traffic-flow-using-feature-based-regression-in-the-spatiotemporal-domain

INTECH
open science | open minds

**InTech Europe**
University Campus STeP Ri
Slavka Krautzeka 83/A
51000 Rijeka, Croatia
Phone: +385 (51) 770 447
Fax: +385 (51) 686 166
www.intechopen.com

**InTech China**
Unit 405, Office Block, Hotel Equatorial Shanghai
No.65, Yan An Road (West), Shanghai, 200040, China
中国上海市延安西路65号上海国际贵都大饭店办公楼405单元
Phone: +86-21-62489820
Fax: +86-21-62489821

© 2011 The Author(s). Licensee IntechOpen. This chapter is distributed under the terms of the Creative Commons Attribution-NonCommercial-ShareAlike-3.0 License, which permits use, distribution and reproduction for non-commercial purposes, provided the original is properly cited and derivative works building on this content are distributed under the same license.