ESTIMATION OF PEDESTRIAN FLOW SPEED IN SURVEILLANCE VIDEOS

Gwang-Gook Lee, Kee-Hwan Ka, Whoi-Yul Kim

Department of Electronics and Computer Engineering
Hanyang University
Seoul, Korea
E-mail: gglee@vision.hanyang.ac.kr

ABSTRACT

This paper proposes a method to estimate the flow speed of pedestrians in surveillance videos. In the proposed method, the average moving speed of pedestrians is measured by estimating the size of real-world motion from the observed motion vectors. For this purpose, pixel-to-meter conversion factors are calculated from camera geometry. Also, the height information, which is missing because of camera projection, is predicted statistically from simulation experiments. Compared to the previous works for flow speed estimation, our method can be applied to various camera views because it separates scene parameters explicitly. Experiments are performed on both simulation image sequences and real video. In the experiments on simulation videos, the proposed method estimated the flow speed with average error of about 0.1m/s. The proposed method also showed a promising result for the real video.

Keywords: Video surveillance, flow speed estimation, motion vectors

1. INTRODUCTION

For the last decade, there have been diverse efforts to develop intelligence video surveillance systems. Intelligent video surveillance systems aim to interpret scenes, detect suspicious events, and alarm human operators to prevent accidents. Research efforts for such intelligent systems include measurement of crowd density [1], detection of unusual events [2], or recognition of abandoned objects [3]. Although previous research efforts have covered such various issues, measuring pedestrian flow speed has received less attention despite its importance. Flow speed of pedestrians provides a means to detect unusual events (e.g., congestion, blocking or emergency) in a global manner. Also, it is essential information to develop pedestrian traffic models, which is used in designing buildings or planning evacuations [4].

In [5], Teknomo *et al*. proposed a data collection system for pedestrian flow analysis. The proposed system automatically detects moving objects, tracks them and stores their locations with time stamps. Using obtained data of individual movements, it was able to observe characteristics of pedestrian traffic flow such as speed of individuals, flow rate or average speed and directions. However, their method is limited to video sequences obtained from a top-view camera and could not guarantee its performance to videos with lower viewing angles. Shimmura *et al*. proposed a method to estimate human-flow speed [6]. In their method, average moving speed of pedestrians was estimated from motion vectors. To transform motion vectors of a video frame into moving velocity in the real-world, scaling factors were utilized that were acquired from a series of simulation images. However, only one fixed camera setting is used to create simulation images in their method. Moreover, because the formula for conversion between motion vectors and real-world speed implicates camera parameters, it is not proven that the same scaling factor can be effectively applied to different camera views.

In this paper, we propose a method to estimate the flow speed of pedestrians from surveillance videos. Estimation of moving velocity is achieved by converting motion vectors from image domain to physical domain. For this purpose, pixel-to-meter conversion factors are computed from camera geometry directly. Also, to overcome the ambiguity of motion vectors (due to the 2D camera projection), we estimate the real-world height of the observed motion vector statistically. This motion height estimation function is obtained using a set of simulation image sequences. Unlike the flow speed estimation method in [6], our proposed method can be applied to various scenes by separating scene parameters explicitly.

Rest of this paper is organized as follows. In Section 2, we introduce the flow speed estimation method. Section 3 explains simulation experiments to overcome the problem of ambiguity of motion projection. Section 4 gives experimental results and Section 5 concludes this paper.

2. FLOW SPEED ESTIMATION FROM MOTION VECTORS

In this section, the proposed method for pedestrian flow speed estimation is explained. For further explanation, an example of a real-world motion and its projection onto the image plane is illustrated in Fig. 1. As shown in the figure, when a motion $\mathbf{v_R}$ occurs in the real-world, it appears as a motion vector $\mathbf{v_I}$ on the image plane. Our goal is to measure the overall flow speed of pedestrians by estimating the size of $\mathbf{v_R}$ and taking an average of them.

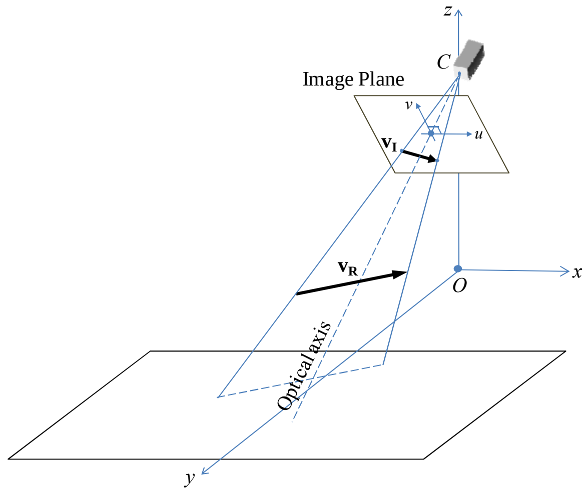


Fig. 1: Projection of a motion onto an image plane

To estimate the speed of real-world motion (the size of motion vector $\mathbf{v_R}$), we first compute conversion factors that transforms pixel units to physical units (meters). Assume that a motion vector of unit length (from pixel p_i to p_j) is observed as in Fig. 2. It is assumed that the camera focal length f and camera tilt angle θ_c are already known. With known camera viewing angle and image resolution, we can compute the tilted viewing angle from optical center θ_i for each pixel position p_i.

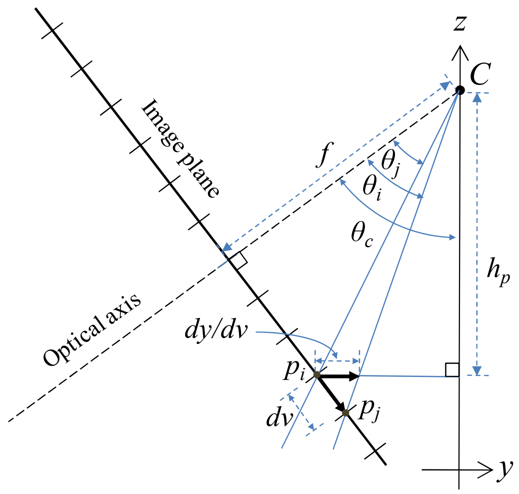


Fig. 2: Conversion from pixels to meters

From these parameters, we can easily compute the pixel-to-meter conversion factor dy/dv as

$$\frac{dy}{dv} = h_p \left\{ \tan(\theta_c - \theta_j) - \tan(\theta_c - \theta_i) \right\}. \tag{1}$$

Also, h_p in Eq. (1) is calculated by

$$h_p = \frac{\cos(\theta_c - \theta_j)}{\cos\theta_j} f. \tag{2}$$

In Fig 2, we only showed the conversion factor for vertical direction, dy/dv. The conversion factor for horizontal direction, dx/du can also be easily obtained by dividing the width of CCD sensor with image resolution. These pixel-to-meter conversion factors (dx/du and dy/dv) and distance from optical center h_p are pre-computed and stored for all pixel positions in the image frame.

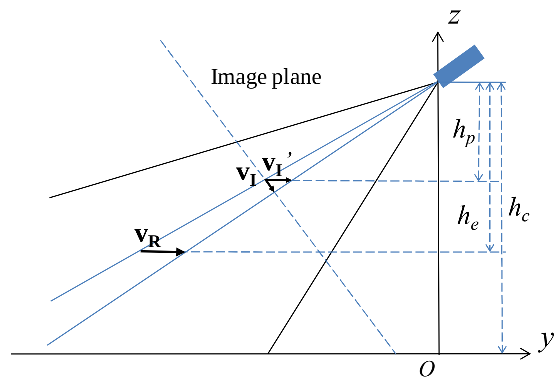


Fig. 3: Estimation of the size of the real-world motion

Using the conversion factors, we can expect the size of real-world motion by a simple trigonometry as shown in Fig. 3. In the figure, $\mathbf{v_I}$**’** is obtained from $\mathbf{v_I}$ by the pixel-to-meter conversion factor. Hence, the size of $\mathbf{v_R}$ can be expected by

$$\left\| \mathbf{v_R} \right\| = \frac{h_e}{h_p} \left\| \mathbf{v_I}' \right\| = \frac{h_e}{h_p} \left\| \left(\mathbf{v_I}(u) \frac{dx}{du}, \mathbf{v_I}(v) \frac{dy}{dv} \right) \right\|. \tag{3}$$

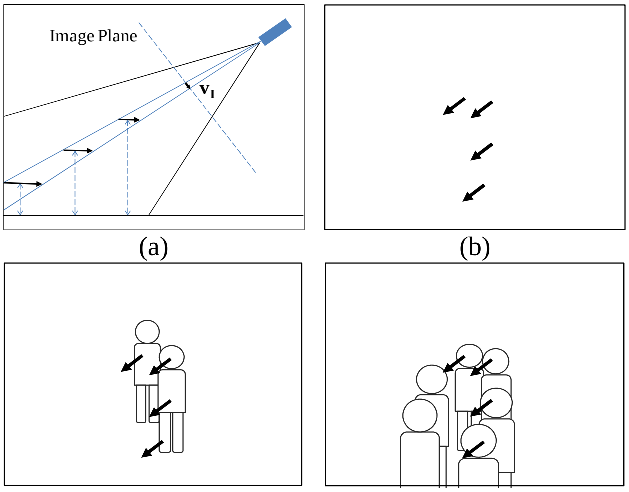
Finally, the overall speed of pedestrians is computed by averaging all motion vectors for image frames as Eq. (4). Here, $\|\mathbf{v_R}(i, t)\|$ indicates the size of motion caused by i_{th} motion vector at frame t. Also, notice that the motion sizes are weighted by w_i to compensate the pixel size variation caused by camera projection as in [1].

$$flow(t) = \frac{\sum_{t=k}^{k+T} \sum_{i=0}^{N} w_i \left\| \mathbf{v_R}(i,t) \right\|}{\sum_{i=0}^{N} w_i}. \tag{4}$$

3. EXPECTING THE HEIGHT OF REAL–WORLD MOTION

Using the method introduced in Section 2, we can estimate the flow speed of pedestrians from motion vectors. However, there still is one more problem that the vertical distance between camera position and the real-world motion h_e cannot be determined from input video. It is because depth (distance from camera) or height of the real-world motion is missing due to camera projection. Fig. 4 shows some examples of this problem. As illustrated in Fig. 4 (a), a vector $\mathbf{v_I}$ on the image plane can come from various motions with different depths or heights. The observed motion vectors in Fig. 4 (b) may be due to motion of different parts of pedestrians as in Fig. 4 (c) and (d).

One of the easiest ways to solve this problem is simply to assume that all motions are occurred on the same plane having the same value of h_e. By setting the value of h_e as the average height for all motion vectors in the frame, the flow speed could be estimated properly. However, the average value of h_e for all motion vectors will be changed for different scenes. Hence, we expect the height of motion vector in real-world statistically.



(a) (b)

(c) (d)

Fig. 4: Ambiguity of projected motion

To begin with, we consider the factors that influence the observation of motion vectors and their expected height in the real-world. Fig. 5 shows an example of motion observation for two pedestrians. In the figure, motions occurred in shaded area will not be observed in the video frame. The amount of visible area for the occluded pedestrian can be expressed as d / tan θ, where d is the distance between two people and θ is the viewing angle. As the visible area decreases, the expected height of observed motion vector will arise because motions from lower part of the human are missed. Hence, the expected real-world height of an observed motion vector follows a relationship in Eq. (5). Here, the crowd density defined by the number of people in a unit area becomes inversely proportional to the average distance between pedestrians.

$$height \propto \frac{1}{d/\tan\theta} \propto \tan\theta \cdot \sqrt{crowd\ density} \tag{5}$$

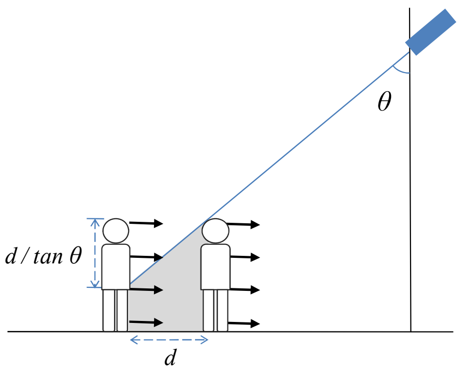


Fig. 5: Relationship between the amount of occlusion, inter-person distance and camera angle

As in Eq. (5), the height information of an observed motion vector can be estimated using a function of viewing angle and crowd density. To train the function for height estimation, we used a simulated video consisted of a set of synthetic image sequence. The simulation video is obtained by modeling humans as ellipsoids and controlling their movements using the microscopic pedestrian model proposed in [5]. About 10,000 simulation images are created with varying camera angles and crowd density. Using the simulated video, a quadratic function as the height of motion vectors in real-world from viewing angle and crowd density is trained. Fig. 7 shows the resulting function for height estimation of real-world motions. As expected, the trained function in Fig. 7 increases to coincide with Eq. (5). That is, it increases similar to a tangent function along the axis of viewing angle and to a square root function with the axis of crowd density, respectively.

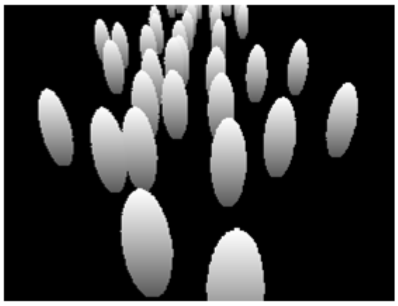


Fig. 6: An example of a synthetic image for simulation

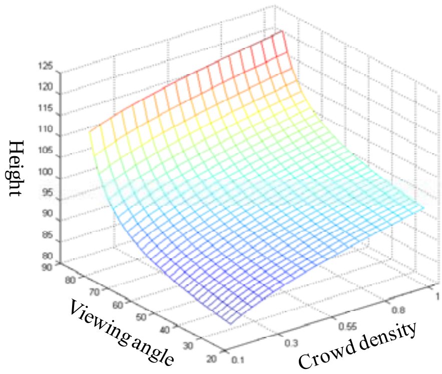


Fig. 7: Height estimation function trained from synthetic images

4. EXPERIMENTS

In this section, we present results of experiments to verify the proposed method. One of the difficulties in the experiments is that the actual speed of pedestrian flow is not easy to measure. Hence, we used two types of dataset: simulated video and real one. The simulated video was prepared by replacing human image to the ellipsoids while moving them according to the pedestrian movement model in [5]. Fig. 8 shows examples of dataset for simulated one and real video, respectively.

(a) (b)

Fig. 8: Examples of (a) a synthetic image and (b) a scene from the real video for experiment

Fig. 9 shows flow speed estimation results for the simulated video. Since positions of each pedestrian are known in advance during the generation stage, accurate flow speed can be easily obtained for every frame of the simulation. Image sequences from three different camera angels (30°, 45° and 60°) are used for the test. Average errors of the flow speed were 0.102 m/s, 0.085 m/s and 0.067m/s for each set.

Fig. 10 gives the speed estimation result for the real scene. Because it is very difficult to obtain actual flow speed from real video frames, the actual flow speed is measured manually by tracking head positions of people in a frame and projecting them inversely with known geometry. The manual estimation is performed for every 500 frames. The solid line in Fig. 10 indicates the estimated flow speed obtained by the proposed method. Also, the result of manual estimation is marked with red squares.

5. CONCLUSION

In this paper, we proposed a method for pedestrian flow speed estimation. Proposed method estimates the flow speed by predicting the size of motion in the real-world from motion vectors in image domain. Pixel-to-meter conversions factor are employed to convert motion vectors from the image domain to the physical domain. Also, the height information, which is missing because of camera projection, is predicted statistically using a simulated video. Unlike the previous method that is limited to a certain view point, our proposed method incorporates camera geometry to cope with the parameters needed for various scenes. For experiments, other sets of simulated and real video were used. Experiments on the simulated video showed that our proposed method can effectively estimate the flow speed with lesser error than or about 0.1m/s. Similar performance was measured for simulated videos at different camera angles. The proposed method also showed a promising result for the real video.

6. REFERENCES

[1] D. Kong, D. Gray and H. Tao, "A Viewpoint Invariant Approach for Crowd Counting," ICPR'06, vol. 03, 2006, pp. 1187-1190.

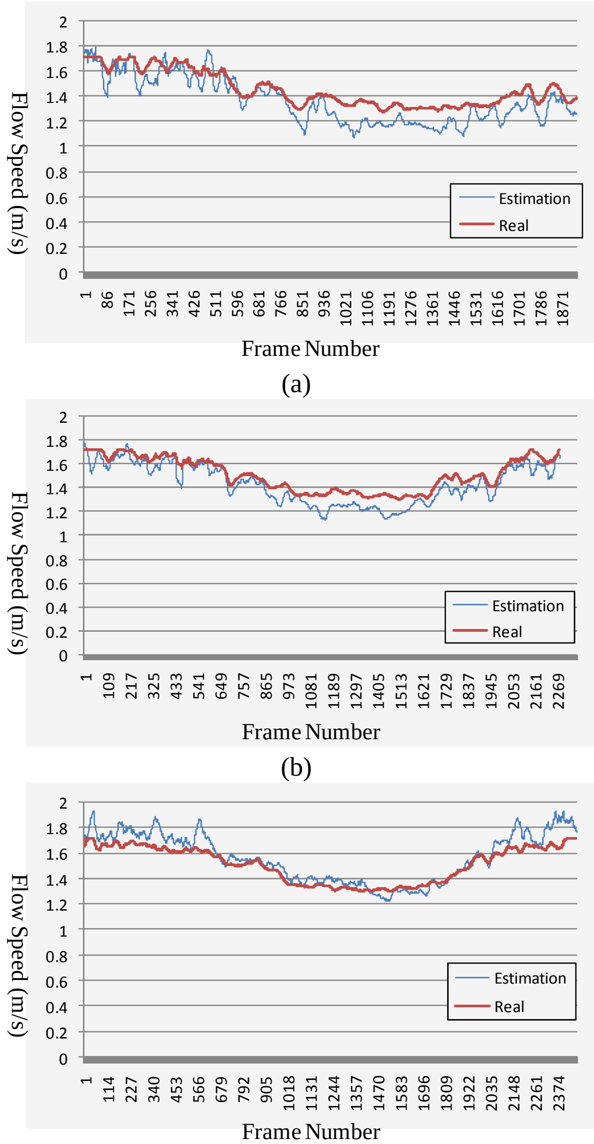
[2] S. A. Velastin, B. A. Boghossian, B. P. L. Lo, S. Jie, M. A. Vicencio-Silva, "PRISMATICA: toward ambient intelligence in public transport environments," Trans. On Systems, Man and Cybernetics, vol. 35, issue 1, pp. 164-182, Jan. 2005.

[3] F. Porikli, Y. Ivanov and T. Haga, "Robust abandoned object detection using dual foregrounds," EURASIP Journal on Advances in Signal Processing, vol. 2008, Issue 1, No. 30, pp. 1-11, 2008.

[4] P. A. Thompson and E.W. Marchant, "Simulex; Developing New Computer Modeling Techniques for Evaluation," Fire Safety Science, vol. 4, pp. 613-624, 1994.

[5] K. Teknomo, "Microscopic Pedestrian Flow Characteristics: Development of an Image Processing Data Collection and Simulation Model," Ph.D. Dissertation. Tohoku University, 2002.

[6] T. Shimmura, H. Arai, and U. Inoue, "Estimating human-flow speed for video surveillance by probabilistic stands," in Cybernetics and Intelligent Systems, 2004 IEEE Conference on. vol. 1, 2004.



(a) (b) (c)

Fig. 9: Estimation results for synthetic image sequences with camera tilt angle of (a) 30°, (b) 45° and (c) 60°

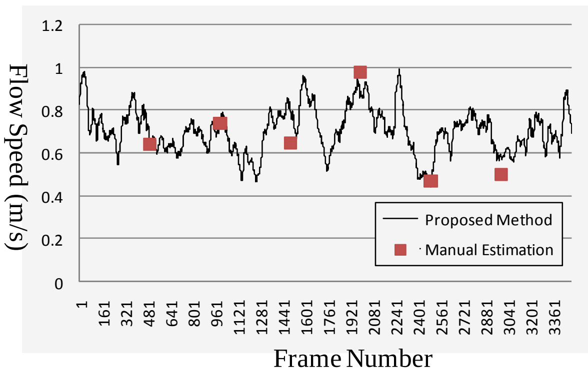


Fig. 10: Flow speed estimation result for real video