Vol. 4.Special Issue ICCSII ISSN 2079-8407
Journal of Emerging Trends in Computing and Information Sciences

http://www.cisjournal.org

Monte Carlo-based Bidirectional Pedestrian Counting Method for Compound-Eye Sensor Systems

Shuto Fujii[1], Yoshiaki Taniguchi[2], Go Hasegawa[2], Hirotaka Nakano[2]

[1]Graduate School of Information Science and Technology, Osaka University, Japan

[2]Cybermedia Center, Osaka University, Japan

E-mail: [1]f-shuto@ist.osaka-u.ac.jp, [2]{y-tanigu, hasegawa, nakano}@cmc.osaka-u.ac.jp

ABSTRACT

We propose a method for estimating counts of pedestrians walking in opposite directions, as in the case of a corridor. The proposed method utilizes a ceiling-mounted compound-eye sensor composed of multiple binary sensors with partially overlapping sensing regions. The output of the compound-eye sensor is sent to a monitoring server, which conducts a number of Monte Carlo simulations. The simulation scenario with the smallest difference from the output history of the compound-eye sensor is selected as an estimate of pedestrian counts. Simulation results show that in crowded situations the proposed method decreases relative error by up to 45%, as compared with an existing method.

Keywords: *Binary Sensor, Compound-eye Sensor, Monte Carlo, Pedestrian Counting, Sensor Network*

1. INTRODUCTION

In various fields such as marketing, traffic control, and safety management, there is a strong demand for a method to estimate pedestrian counts accurately. For example, information on the temporal change in traffic volume of a footway can be used to determine appropriate times for construction or maintenance [1]. In addition, information regarding the number and movement direction of pedestrians at the entrances of commercial facilities, event sites, or food courts could prevent crowding. Manual counting is often used to collect such information, but this entails high labor costs and cannot be used in crowded situations. Therefore, automatic methods for estimating pedestrian counts have attracted considerable attention.

Most studies on estimating pedestrian counts are conducted in the field of computer vision [2]. Processing video images allows estimates of characteristics of moving pedestrians such as height, velocity, flow lines, and counts. However, methods based on video processing require significant processing power, operating memory, storage, and electric power. In addition, brightness of the background influences the estimation accuracy of such methods.

Previous studies on pedestrian counting have used devices such as infrared imaging sensors [3], laser sensors [4], and ultrasonic sensors [5]. Commercial pedestrian counters include devices that use infrared imaging sensors [6], active infrared sensors [7], passive infrared sensors [8], piezo films [9], and laser scanners [10]. Binary sensors such as infrared and ultrasonic sensors are among the simplest devices, capable of detecting only the presence or absence of objects within the sensing region. Binary sensors can neither count pedestrians nor identify individual pedestrians within the sensing region, but they possess advantages such as low cost, simplicity, small size, and energy efficiency, so methods for estimating pedestrian counts using binary sensors have attracted attention. Since the capabilities of a single binary sensor are limited, researchers have considered using combinations of binary sensors for estimating pedestrian counts and movement direction [5, 11–13]. However, the estimation accuracy of such methods is relatively low in crowded situations where many pedestrians move simultaneously.

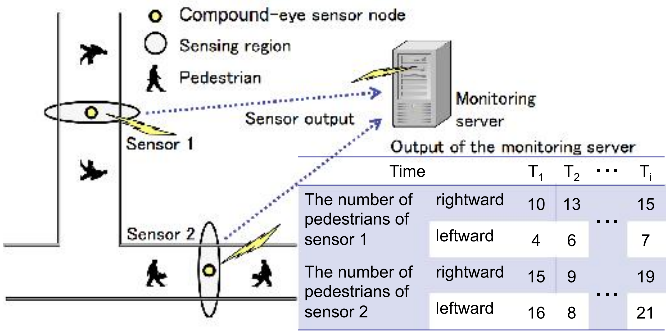

Fig. 1. The compound-eye sensor system.

We previously proposed a method for estimating pedestrian counts and movement directions using combinations of binary sensors in situations where many pedestrians walk in opposite directions, such as in a corridor [14]. Figure 1 shows the system used for estimating pedestrian counts in this study. This sensor system consists of a monitoring server and ceiling-mounted compound-eye sensor nodes. Each node consists of a compound-eye sensor and a wireless transceiver. The system uses passive binary sensors, such as pyroelectric infrared sensors [15]. The sensor output obtained from a compound-eye sensor node is sent to the monitoring server via a wireless network, and the pedestrian count is estimated on the monitoring server. In [14], we proposed a Monte Carlo-based method for estimating pedestrian counts when the compound-eye sensor consists of two binary sensors, and evaluated the method under limited conditions.

The present paper describes a Monte Carlo-based method for estimating pedestrian counts using compound-eye sensors consisting of multiple binary sensors. The proposed method is based on our previous proposal [14], and divides the compound-eye sensor into a set of local compound-eye sensors, each having two binary sensors. The number of pedestrians is estimated for each local compound-eye sensor in parallel, and one of the estimation results is selected. To estimate pedestrian counts for each local compound-eye sensor, the monitoring server maintains a number of simulation fields, each of which contains a virtual compound-eye sensor and virtual pedestrians. While the monitoring server receives information about the presence of pedestrians from the compound-eye sensor nodes, it randomly generates virtual pedestrians in the simulation field. Next, it updates the output history of the virtual compound-eye sensor and counts the number of virtual pedestrians for each simulation field. The simulation field with the smallest difference between the output history of the compound-eye sensor and that of the virtual compound-eye sensor is selected as an estimate of pedestrian counts. We evaluated the accuracy of the proposed method through simulation experiments by changing the number of simulation fields, the specifications of the compound-eye sensor, and the number of binary sensors. We also evaluated the proposed method in comparison with an existing method, which estimates pedestrian counts using the duration in which any of the binary sensors detects the presence of pedestrians.

The rest of this paper is organized as follows: Section 2 outlines related work. Section 3 describes the binary sensor model, the compound-eye sensor model, and the mobility model of pedestrians used in this paper. Section 4 describes the Monte Carlo-based method for estimating pedestrian counts. Section 5 uses simulations to evaluate the performance of the proposed method. Section 6 presents our conclusions and areas of future work.

2. RELATED WORK

Several methods have been proposed that use combinations of binary sensors to estimate pedestrian counts and movement directions [11–13]. In [11], the authors proposed a counting system for a mountain trail, where two active photobeam sensors, composed of a photobeam transmitter and a photobeam receiver, are installed parallel to the trail. In that method, pedestrian counts in each direction are estimated based on the order in which two sensors trigger and the duration of the interval in which photobeam sensors detect pedestrians. If the duration is shorter than a certain threshold, the system decides that a single pedestrian has passed by the sensors; otherwise it decides that two pedestrians have passed by the sensors.

Unlike the above works, [12] uses passive binary sensors for tracking pedestrians by means of a ceiling-installed device combining two passive infrared sensors and an active ultrasonic sensor. The movement direction of pedestrians is estimated from the triggering order of the two sensors. In addition, individual pedestrians are identified on the basis of height as measured with the active ultrasonic sensor. Our research group, too, has proposed a lightweight method for estimating counts of pedestrians walking in opposite directions, based on the triggering of two passive binary sensors [13]. Simulations showed that the estimation accuracy of the proposed method decreases significantly in crowded situations where many pedestrians move simultaneously.

Object tracking is another intriguing human sensing problem [16], and a number of related studies have been conducted [17–19]. The multiple hypothesis tracker (MHT) [17] is a well-known method for tracking multiple objects. In MHT, all possible situations ("hypotheses") are maintained based on sensor information, and the hypothesis with the highest event probability is selected as the estimated trajectory of the object. The number of hypotheses increases exponentially with the number of objects, however, so MHT cannot be used in crowded situations where many pedestrians move simultaneously. Methods based on Markov chain Monte Carlo (MCMC) simulations [18, 19] have been proposed as a solution for this problem, and such models have been shown to significantly improve the accuracy of object tracking in crowded situations.

The abovementioned lightweight methods [11–13] consider only a limited set of situations, where only one or two pedestrians pass by any given binary sensor, drastically decreasing estimation accuracy in crowded situations. We propose a method based on combined operation of binary sensors for estimating counts of pedestrians walking in opposite directions in crowded situations. To improve the estimation accuracy in crowded situations, the proposed method considers a variety of situations during the time interval where the binary sensors detect pedestrians by managing hypotheses on pedestrian trajectories ("simulation fields"), similar to object tracking methods. However, as in MHT, since maintaining all hypothesis results in an exponential increase in hypotheses, we use a Monte Carlo-based method similar to MCMC-based methods to limit growth.

3. COMPOUND-EYE SENSOR SYSTEM

This section describes the compound-eye sensor system, binary sensor models, the compound-eye sensor used in the system, and a pedestrian mobility model.

3.1. System overview

Figure 1 shows the compound-eye sensor system, which consists of a monitoring server and compound-eye sensor nodes. A compound-eye sensor node consists of a compound-eye sensor and a wireless transceiver. A compound-eye sensor is composed of multiple ceiling-mounted binary sensors, assumed to be pyroelectric infrared sensors, whose sensing regions partially overlap. Sensor nodes send data packets to the monitoring server when the output of their binary sensor changes. Data packets contain the sensor's output value and a timestamp. The monitoring server uses the presence or absence of information from sensors to estimate counts of pedestrians walking in opposite directions.

For simplicity, we assume that the system has a single compound-eye sensor node, that data packets are reliably sent to the monitoring server, and that transmission latency is negligible.

3.2. Binary sensor model

We assume a rectangular sensing region such as that created by a linear Fresnel lens (Fig. 2). The sensing region is determined for each pedestrian. We refer to the distance from the center of the binary sensor to an edge of the sensing region as the "sensing length." The sensing length differs when a pedestrian enters or exits the sensing region, because pedestrian movement is detected stochastically at the edge of the sensing region [20]. We denote the sensing length on entry and exit as the incoming sensing length r_{in} and the outgoing sensing length r_{out}, respectively. In the case of pyroelectric infrared sensors r_{out} is greater than r_{in}, because the sensor incurs a delay when its output changes from 1 to 0. The incoming sensing length $r_{in,k}$ for pedestrian p_k is therefore determined randomly as $r_{\min} \le r_{in,k} \le r_{\max}$, and the outgoing sensing length $r_{out,k}$ for pedestrian p_k is determined randomly as $r_{\min} + r_{off} \le r_{out,k} \le r_{\max} + r_{off}$, where $r_{\max}$ is the maximum sensing length, $r_{\min}$ is the minimum sensing length, and r_{off} is the distance traveled by the pedestrian during the delay.

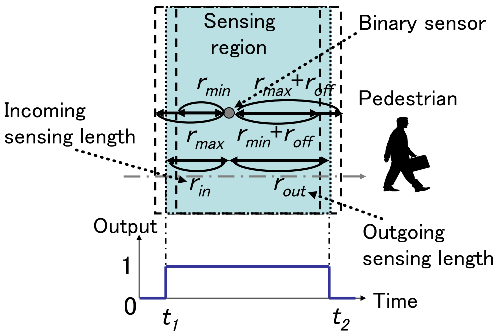

Fig. 2. Sensing model of a binary sensor.

A binary sensor outputs a value of 1 when a pedestrian is inside its sensing region and 0 otherwise. Pedestrian counts are therefore possible when the sensor outputs a 0 only; otherwise the number of pedestrians cannot be determined.

3.3. Compound-eye sensor model

A compound-eye sensor consists of multiple binary sensors b_i ($1 \le i \le N$). These sensors are linearly placed with a regular spacing at the center of a monitoring area, for example, along a corridor. Figure 3 shows an example of a compound-eye sensor composed of four sensors b_1, b_2, b_3, and b_4. The distance between each sensor is denoted as the sensor distance d. We define the direction of moving from sensor b_1 toward sensor b_N as right, and the opposite direction as left. The output of sensor b_i at time t is denoted as $o_t^i \in \{0,1\}$. Furthermore, the output of the compound-eye sensor at time t is denoted as $[o_t^1, o_t^2, \cdots, o_t^N]$.

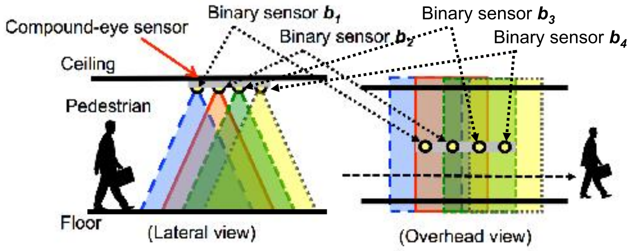

Fig. 3. Deployment of compound-eye sensor.

When the output of the compound-eye sensor is $[0, 0, \cdots, 0]$, the pedestrian count under the compound-eye sensor can be estimated as zero. We refer to this as an observable state. For other outputs the pedestrian count cannot be determined, and this is referred to as an unobservable state.

3.4. Mobility model of pedestrians

We assume that pedestrians move either left ("leftward" pedestrians) or right ("rightward") within the monitoring area, and they do not change their movement direction or velocity. Leftward (rightward) pedestrians arrive from the right (left) side of the monitoring area by a Poisson process with a leftward (rightward) arrival rate λ_l (λ_r). The velocity distribution of pedestrians is a normal distribution with mean v_m and deviation v_s.

4. MONTE CARLO-BASED PEDESTRIAN COUNTING

This section describes a Monte Carlo-based method for estimating pedestrian counts. The proposed method consists of two sub-methods, one for estimating pedestrian counts using estimation results of local compound-eye sensors, and one for estimating pedestrian counts by conducting Monte Carlo simulations.

4.1. Estimating pedestrian counts with local compound-eye sensors

This section describes the first sub-method.

4.1.1. Overview

The proposed method divides the compound-eye sensor into a set of local compound-eye sensors, each having two binary sensors. Hereinafter, we refer to each local compound-eye sensor as a "paired sensor". A paired sensor c_k ($1 \le k \le N-1$) is composed of neighboring binary sensors b_k and b_{k+1}. The output of paired sensor c_k at time t is denoted as $[o_t^k, o_t^{k+1}]_k$. For example, a compound-eye sensor composed of four binary sensors (Fig. 3) is considered as a set of three paired sensors c_1, c_2, and c_3 in the proposed method (Fig. 4).

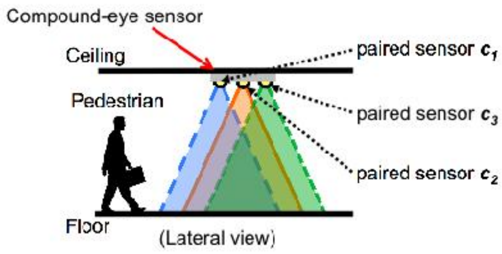

Fig. 4. Three paired sensors.

The number of pedestrians under each paired sensor is estimated in parallel, and one estimate is selected. Figure 5 shows an example of behavior of the proposed method. In that figure, solid and dotted circles indicate observable and unobservable states, respectively. The states for paired sensors c_1, c_2, and c_3 are shown below the states of the compound-eye sensor. The time during which a paired sensor is in an unobservable state is denoted as an "unobservable interval," and depicted as an arrow below the states. The following discussion of the behavior of the proposed method refers to Fig. 5.

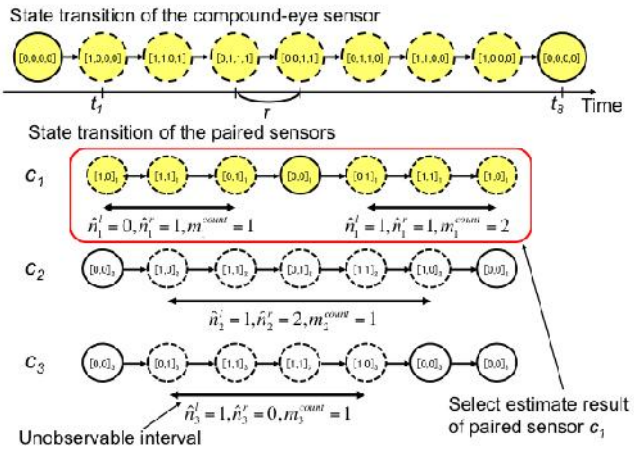

Fig. 5. Example of behavior of the proposed method.

When the monitoring server receives a packet with information about the presence of a pedestrian and the state of the compound-eye sensor undergoes transition from an observable state to an unobservable state, the monitoring server starts to estimate pedestrian counts using paired sensors. While the compound-eye sensor is in an unobservable state, the monitoring server estimates pedestrian counts for each paired sensor in parallel. The method to estimate pedestrian counts in a paired sensor in its unobservable interval is based on our previous proposal [14], and explained in Section 4.2. When the monitoring server receives a packet with information about the absence of pedestrians and the state of compound-eye sensor undergoes transition from an unobservable state to an observable state, the estimate of the paired sensor with the largest number of unobservable intervals is selected as the estimated pedestrian count. This is because our previous study [14] indicated that shorter unobservable intervals achieve higher accuracy. In Fig. 5, the estimation result of paired sensor c_1 is selected, since the number of unobservable intervals is largest.

4.1.2. Details

This section explains the first sub-method in detail. The monitoring server checks the output of the compound-eye sensor at regular intervals τ, which should be determined such that the output of the binary sensor changes at most once within a single interval. We denote as t the ith period after the monitoring server commences operation. The monitoring server tracks the estimated total number of rightward pedestrians $\hat{n}_k^r$, the estimated total number of pedestrians $\hat{n}_k^l$, and the number of unobservable intervals m_k^{count} for each paired sensor c_k. The monitoring server acts according to the preceding output $[o_{i-1}^1, o_{i-1}^2, \cdots, o_{i-1}^N]$ of the compound-eye sensor at time t_{i-1} and the current output $[o_i^1, o_i^2, \cdots, o_i^N]$ of the compound-eye sensor at time t_i as follows:

Transition of a compound-eye sensor between observable states: When the output of the compound-eye sensor does not change from $[0, 0, \cdots, 0]$, the monitoring server does nothing.

Transition of a compound-eye sensor from an observable sate to an unobservable state: When the output of the compound-eye sensor changes from $[0, 0, \cdots, 0]$, the monitoring server initializes the estimated number of rightward pedestrians $\hat{n}_k^r$, the estimated number of leftward pedestrians $\hat{n}_k^l$ and the number of unobservable intervals m_k^{count} to 0 for each paired sensor c_k.

Transition of a paired sensor from an observable state to an unobservable state: When the state of paired sensor c_k changes from an observable state to an unobservable state (the output of paired sensor c_k changes from $[0, 0]_k$), the monitoring server begins estimating pedestrian counts for paired sensor c_k by conducting Monte Carlo simulations. (See Section 4.2 for details.)

Transition of a compound-eye sensor between unobservable states: When the output of the compound-eye sensor undergoes transition between states excluding $[0, 0, \cdots, 0]$, the monitoring server continues estimating pedestrian counts for each paired sensor.

Transition of a paired sensor from an unobservable state to an observable state: When the state of paired sensor c_k changes from an unobservable state to an observable state (the output of paired sensor c_k changes to $[0, 0]_k$), the monitoring server finishes estimating pedestrian counts for paired sensor c_k, and obtains the estimated number of rightward pedestrians $\hat{n}_{k,f}^r$ and the estimated number of leftward pedestrians $\hat{n}_{k,f}^l$ (see Section 4.2 for details). Then, the monitoring server adds the estimated number of rightward pedestrians $\hat{n}_{k,f}^r$ and the estimated number of leftward pedestrians $\hat{n}_{k,f}^l$ to the estimated total number of rightward pedestrians $\hat{n}_k^r$ and the estimated total number of leftward pedestrians $\hat{n}_k^l$, respectively. It also increases the number of unobservable durations m_k^{count} for paired sensor c_k by one.

Vol. 4.Special Issue ICCSII ISSN 2079-8407
Journal of Emerging Trends in Computing and Information Sciences

http://www.cisjournal.org

Transition of a compound-eye sensor from an unobservable state to an observable state: When the output of the compound-eye sensor changes to $[0,0,\cdots,0]$, the monitoring server selects the estimate of paired sensor c_k with the largest number of unobservable intervals as an estimated pedestrian count. If multiple estimates share the largest number of unobservable intervals, the monitoring server selects the estimate of paired sensor c_k with the median value in terms of the estimated total number of pedestrians $\hat{n}_k^l + \hat{n}_k^r$ from the set of candidates. After this step, the monitoring server chooses $\hat{n}_k^l$ as the estimated number of leftward pedestrians $\hat{n}^l$ and $\hat{n}_k^r$ as the estimated number of rightward pedestrians $\hat{n}^r$.

4.2. Estimating pedestrian counts in each local compound-eye sensor

This section describes the second sub-method, which estimates pedestrian counts under a paired sensor during an unobservable interval. This method is based on our previous proposal [14].

4.2.1. Overview

In the second sub-method, a number of Monte Carlo simulations are conducted for paired sensor c_k to estimate pedestrian counts while paired sensor c_k is in unobservable states. Figure 6 shows an example Monte Carlo simulation using a paired sensor. In that figure, solid circles indicate observable states, and dotted circles indicate unobservable states. In the circles, the number of leftward pedestrians n_k^l and the number of rightward pedestrians n_k^r are collectively denoted as $(n_k^l, n_k^r)_k$. When the state of paired sensor c_k undergoes transition from an observable state to an unobservable state, the monitoring server begins estimating pedestrian counts as described in previous section, and the monitoring server generates M simulation fields for paired sensor c_k, each of which contains a virtual paired sensor and virtual pedestrians. While paired sensor c_k is in an unobservable state, the monitoring server randomly generates virtual pedestrians, moves them in the simulation field, updates the output history of the virtual paired sensor, and counts the number of virtual pedestrians for each simulation field. When the state of each paired sensor undergoes transition from an unobservable state to an observable state, the monitoring server finishes estimating pedestrian counts. The simulation field with the smallest difference between the output history of the paired sensor and that of the virtual paired sensor is selected as the estimated pedestrian count.

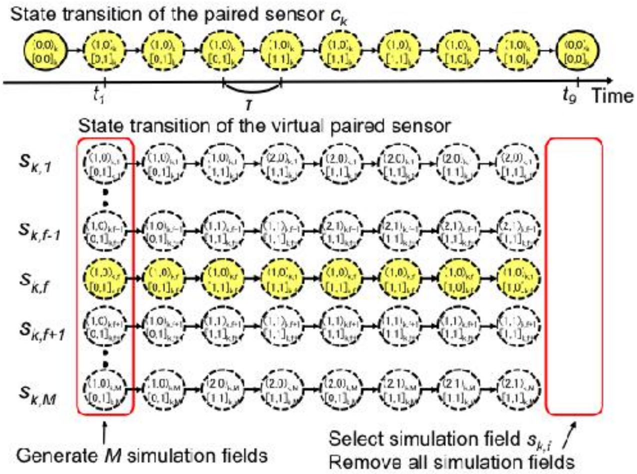

Fig. 6. An example Monte Carlo simulation.

4.2.2. Details

This section describes the second sub-method in detail. To simulate pedestrian movement and detection values for the paired sensor constituting the compound-eye sensor in the simulation fields, we assume that the monitoring server knows the maximum sensing length $r_{\max}^{ver}$, the minimum sensing length $r_{\min}^{ver}$, the sensor delay distance r_{off}^{ver}, the sensor distance d^{ver}, the number of binary sensors N^{ver}, the leftward pedestrian arrival rate λ_l^{ver}, the rightward pedestrian arrival rate λ_r^{ver}, the mean pedestrian velocity s v_{m}^{ver}, and the velocity deviation v_{s}^{ver}. The monitoring server uses the preceding output $[o_{i-1}^k, o_{i-1}^{k+1}]_k$ of the compound-eye sensor at time t_{i-1} and the current output $[o_i^k, o_i^{k+1}]_k$ of the compound-eye sensor at time t_i as follows:

Starting Monte Carlo simulations: When the output of paired sensor c_k changes from $[0,0]_k$, the monitoring server begins estimating pedestrian counts for paired sensor c_k. The monitoring server first initializes the output history O_k of the paired sensor as the current output of the paired sensor as $O_k = \{[o_i^k, o_i^{k+1}]_k\}$, then generates a set of simulation fields $S_k = \{s_{k,j} \mid 1 \le j \le M\}$. For each simulation field $s_{k,j} \in S_k$, the monitoring server maintains the number of virtual rightward pedestrians $n_{k,j}^r$, the number of virtual leftward pedestrians $n_{k,j}^l$, and the output history $O_{k,j}$ of the virtual paired sensor. The initial values of these parameters are set as follows. If the current output is $[1,0]_k$, the monitoring server generates one virtual rightward pedestrian at the left edge of its virtual sensing region, and sets $n_{k,j}^r$ to 1 and $n_{k,j}^l$ to 0 for all simulation fields. If the current output is $[0,1]_k$, the monitoring server generates one virtual leftward pedestrian at the right edge of its virtual sensing region, and sets $n_{k,j}^r$ to 0 and $n_{k,j}^l$ to 1 for all simulation fields. If the output of the current output is $[1,1]_k$, the monitoring server generates one virtual rightward pedestrian at the left edge of its virtual sensing region and one virtual leftward pedestrian at the right edge of its virtual sensing region, and sets $n^r_{k,j}$ and $n^l_{k,j}$ to 1 for all simulation fields. The number of virtual leftward and rightward pedestrians in simulation field $s_{k,j}$ are collectively denoted as $(n^l_{k,j}, n^r_{k,j})_{k,j}$. The output history of the virtual compound-eye sensor is set as $O_{k,j} = \{[o^k_{i,j}, o^{k+1}_{i,j}]_{k,j}\}$, where $[o^k_{i,j}, o^{k+1}_{i,j}]_{k,j}$ is the output of the virtual compound-eye sensor at time t_i in simulation field $s_{k,j}$.

Continuing Monte Carlo simulations: When the output of paired sensor c_k transitions between any two of the three states $[1,0]_k$, $[1,1]_k$, and $[0,1]_k$, the monitoring server continues estimating pedestrian counts for paired sensor c_k. The monitoring server first updates the output history of the paired sensor as $O_k \leftarrow O_k \cup [o^k_i, o^{k+1}_i]_k$, after which it randomly generates a virtual pedestrian, moves the virtual pedestrians in the simulation field, and updates the output history of the virtual paired sensor for each simulation field based on the information given. If a pedestrian enters the virtual sensing region in simulation field $s_{k,j}$, the monitoring server updates the number of virtual leftward pedestrians $n^l_{k,j}$ or the number of virtual rightward pedestrians $n^r_{k,j}$. Then, the output history of the virtual paired sensor in simulation field $s_{k,j}$ is updated as $O_{k,j} \leftarrow O_{k,j} \cup [o^k_{i,j}, o^{k+1}_{i,j}]_{k,j}$.

Finishing Monte Carlo simulations: When the output of paired sensor c_k changes to $[0,0]_k$, the monitoring server finishes estimating pedestrian counts for paired sensor c_k. The monitoring server chooses one simulation field as follows: First, the monitoring server calculates the difference $\Delta(O_{k,j})$ between the output history of paired sensor O_k and the output history of the virtual compound-eye sensor $O_{k,j}$ in simulation field $s_{k,j}$ as follows:

$$\Delta(O_{k,j}) = \sum_{[o^k_{i,j}, o^{k+1}_{i,j}]_{k,j} \in O_{k,j}} \left(\left| o^k_i - o^k_{i,j} \right| + \left| o^{k+1}_i - o^{k+1}_{i,j} \right| \right) \quad (1)$$

Equation (1) indicates the sum of the Hamming distance between the output history of binary sensor c_k and that of the corresponding virtual binary sensor, and the Hamming distance between the output history of binary sensor c_{k+1} and that of the corresponding virtual binary sensor. Next, the monitoring server chooses a set of feasible simulation fields $S_{k,fs}$ as follows:

$$S_{k,fs} = \left\{ s_{k,j} \middle| \min_{1 \le j \le M} \Delta\left(O_{k,j}\right) \right\} \quad (2)$$

Subsequently, the monitoring server selects the most feasible simulation field $s_{k,f}$ with the median value in terms of the total number of virtual pedestrians $n^l_{k,f} + n^r_{k,f}$ from the set of feasible simulation fields $S_{k,fs}$. After this step, the monitoring server chooses $n^l_{k,f}$ as the estimated number of leftward pedestrians $\hat{n}^l_{k,f}$ and $n^r_{k,f}$ as the estimated number of rightward pedestrians $\hat{n}^r_{k,f}$. Finally, the monitoring server deletes all simulation fields.

5. PERFORMANCE EVALUATION

This section evaluates the accuracy of the proposed method through simulation experiments.

5.1. Simulation setting

We evaluate the proposed method through simulations. To evaluate the basic characteristics of the proposed method, we assume that the monitoring server knows the maximum and minimum sensing lengths of the binary sensors, the sensor delay distance, the sensor distance, and the mean and deviation of the velocity of pedestrians, where $r^{ver}_{\max} = r_{\max}$, $r^{ver}_{\min} = r_{\min}$, $r^{ver}_{off} = r_{off}$, $d^{ver} = d$, $v^{ver}_m = v_m$, $v^{ver}_s = v_s$, respectively. We also assume that the leftward pedestrian arrival rate and the rightward pedestrian arrival rate are the same, in other words, $\lambda_l = \lambda_r = \lambda$ and $\lambda^{ver}_l = \lambda^{ver}_r = \lambda^{ver}$. We consider two types of binary sensor: ideal and probabilistic. For ideal binary sensors, we assume that the maximum and minimum sensing lengths are the same ($r_{\max} = r_{\min} = r$). The sensing length r is set to 0.5 m, and r_{off} is set to 0 m. For probabilistic binary sensors, $r_{\max}$ is set to 0.5 m, $r_{\min}$ is set to 0.4 m, and r_{off} is set to 0.1 m. The distance d between sensors is set to 0.1 m, the mean velocity of pedestrians v_m is set to 1.39 m/s, the deviation v_s of the velocity is set to 0.21, and the interval of the proposed method τ is set to 0.01 s.

We use an unobservable interval as one evaluation interval. To evaluate the estimation accuracy of the proposed method, we use relative error ε_{rel} as an evaluation index:

$$\varepsilon_{rel} = \frac{\left|\hat{n}^l - n^l\right| + \left|\hat{n}^r - n^r\right|}{n^l + n^r} \quad (3)$$

Here, a smaller relative error indicates higher estimation accuracy. We also use the duration of the unobservable interval as an index of the proposed method. We refer to the duration as the delay time in this section, meaning the delay from the moment when the proposed method starts estimating pedestrian counts to the moment when estimation completes.

5.2. Evaluation of basic characteristics

We first evaluate the basic characteristics of the proposed method with two ideal binary sensors by changing the number of simulation fields M, the arrival rate of pedestrians λ, the sensor distance d, and the sensing length r. In the evaluation, we assume that the monitoring server knows the arrival rate of pedestrians λ. Figure 7 shows the relative error with a 95% confidence interval against the number of simulation fields and the arrival rate of pedestrians over 1000 evaluations. The relative error decreases when the number of simulation fields increases, since the difference Δ of feasible simulation fields becomes smaller as the number of simulation fields increases. A simulation field with a smaller difference Δ indicates that the simulation field is more suitable as a simulation of the monitoring area. Therefore, the relative error decreases when the number of simulation fields increases. In this evaluation, 1000 is a sufficiently large value for the number of simulation fields M, since the relative error converges (Fig. 7). The relative error increases with arrival rate because the unobservable interval increases with arrival rate.

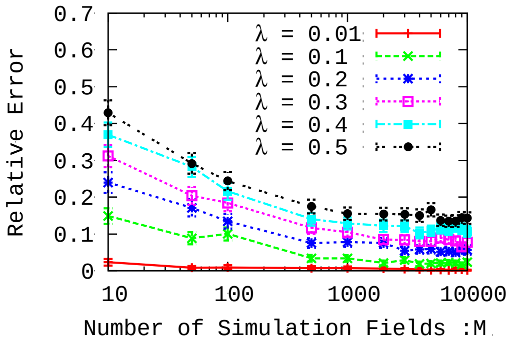

Fig. 7. Relative error against the number of simulation fields.

Figures 8 and 9 show the relative error against the sensor distance and the sensing length, respectively, when the number of simulation fields M is set to 2000. The relative error increases with sensor distance (Fig. 8) and with sensing length (Fig. 9). These trends follow from unobservable intervals increasing with distance between sensors or sensing length. Therefore, the sensor distance and the sensing length should be minimized.

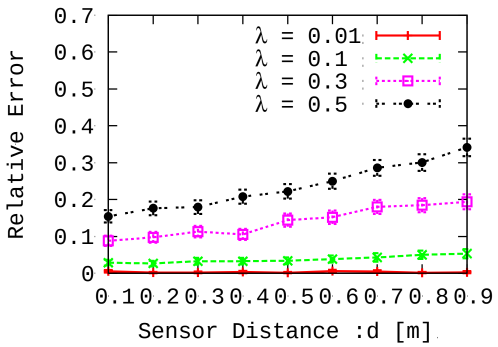

Fig. 8. Relative error against sensor distance.

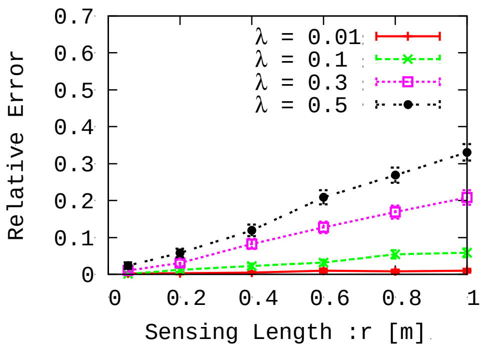

Fig. 9. Relative error against sensing length.

5.3. Evaluation against the number of binary sensors

We next evaluate the basic characteristics of the proposed method by changing the number of binary sensors N. We set the sensor distance d to 0.1 m, the sensing length r to 0.5 m, and the number of simulation fields M to 2000. We conduct 10000 evaluations for each result.

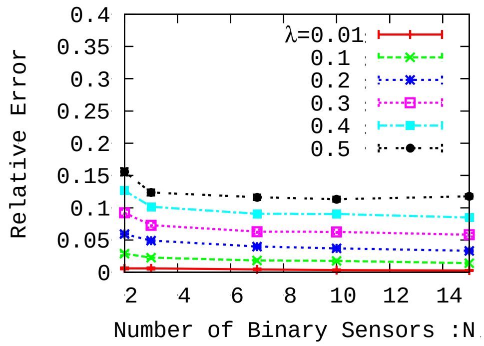

Fig. 10. Relative error against the number of ideal binary sensors.

Figure 10 shows the relative error against the number of binary sensors when ideal binary sensors are used. The relative error decreases when the number of binary sensors increases, because pedestrian counts are estimated for each paired sensor in parallel, and one of the estimation results is selected. Since velocities differ among pedestrians, the unobservable interval differs for each paired sensor. The relative error decreases when the unobservable interval is shorter, as previously described, and increasing the number of binary sensors increases the probability that a paired sensor has a shorter unobservable interval than other sensors. Relative error thus decreases when the number of binary sensors increases. For example, the relative error of the proposed method using three binary sensors is about 21% smaller than when using two binary sensors when the arrival rate is 0.5. When seven sensors are used the error is about 24% smaller, but beyond that the relative error converges (Fig. 10).

Figure 11 shows the delay time against the number of binary sensors. Delay time increases with the number of binary sensors, because the sensing region of the compound-eye sensor increases. Delay time also increases with arrival rate, because pedestrians are more likely to be within the sensing region. Relative error converges when the number of binary sensors exceeds seven, but the delay time increases (Fig. 11). The number of binary sensors should therefore not exceed 7.

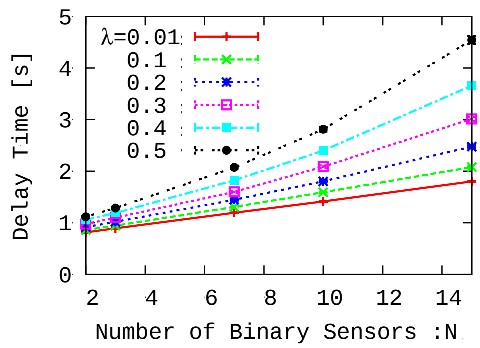

Fig. 11. Delay time against the number of binary sensors.

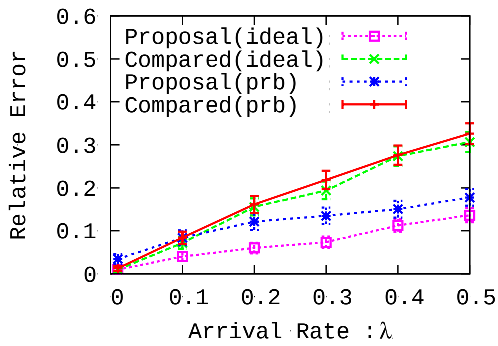

Fig. 13. Comparison with an existing method.

Figure 12 shows the relative error against the number of binary sensors when probabilistic binary sensors are used. The relative error decreases when the number of binary sensors increases, even if the sensors use stochastic pedestrian detection. The relative error for probabilistic binary sensors is higher than for ideal binary sensors (Figs. 10 and 12); estimating pedestrian counts with probabilistic binary sensors is a complex task, since pedestrians are stochastically detected at the edge of the sensing region in both the monitoring area and each simulation field.

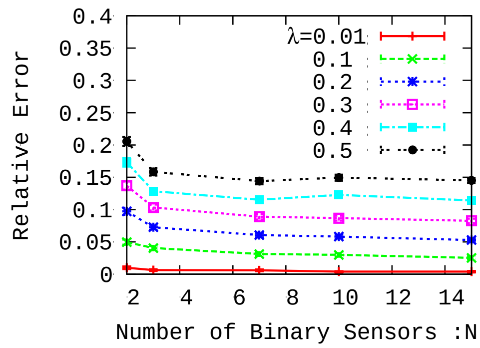

Fig. 12. Relative error against the number of probabilistic binary sensors.

5.4. Comparison with an existing method

Lastly, we evaluate performance of the proposed method in comparison with an existing method proposed in [11]. In the existing method, two binary sensors are considered and pedestrian counts are estimated based on the duration of the unobservable interval. If the duration is shorter than a given threshold, the pedestrian count is estimated as 1, and 2 otherwise. We use 1.41 as the threshold value in the comparison method, since the relative error is smallest at this value. For the evaluation of the proposed method, we assume that the arrival rate is unknown, and the arrival rate λ^{ver} in each simulation field is selected randomly from the interval between the minimum arrival rate λ_{min} (0) and the maximum arrival rate λ_{max} (0.5). We use both ideal and probabilistic binary sensors, setting the number of binary sensors N to 2, and the number of simulation fields M to 2000. We conduct 1000 evaluations for each result.

Figure 13 shows the relative error for the proposed method and the existing method against the arrival rate. The relative error for the proposed method is smaller than that for the existing method when ideal binary sensors are used, and this difference is particularly notable when the arrival rate is 0.5; there, the relative error for the proposed method is about 45% smaller than that for the existing method. This might be because the existing method assumes only one or two pedestrians simultaneously passing under the compound-eye sensor, and that two pedestrians will be moving in the same direction. In contrast, the Monte Carlo simulations allow the proposed method to consider a variety of situations in terms of the number and direction of pedestrians, decreasing the relative error in crowded situations where the arrival rate is high.

The relative error using probabilistic binary sensors is higher than for ideal binary sensors in both methods (Fig. 13), and the difference is nearly constant regardless of the arrival rate. The relative error for the proposed method with probabilistic binary sensors therefore exceeds that for the existing method when the arrival rate is 0.01, despite having lower error in crowded situations. This is a result of the stochastic nature of pedestrian detection in the monitoring area and each simulation field when probabilistic binary sensors are used, and makes pedestrian count estimation in the proposed method a rather complex task. Since the existing method assumes only one or two pedestrians simultaneously pass under the compound-eye sensor, the increase in relative error when using probabilistic binary sensors is prevented when the arrival rate is small. A potential solution to this problem is use of a hybrid method that uses the existing method when unobservable intervals are short, and the proposed method when long. A detailed study of such a hybrid method is reserved for future work.

6. CONCLUSION AND FUTURE WORK

We proposed a Monte Carlo-based method for estimating counts of pedestrians moving in opposite directions based on information regarding the presence of pedestrians detected by a compound-eye sensor. Simulations showed that the relative error in crowded situations decreased by up to 45% as compared with an existing method. In future work, we plan to evaluate the performance of the proposed method in real-world environments through implementation and experiments. We also plan to extend the proposed method to handle multiple movement directions.

ACKNOWLEDGMENTS

This work was partly supported by the Grant-in-Aid for Young Scientists (B) 23700079, JSPS, Japan.

REFERENCES

[1] W. Leutzbach, Introduction to the Theory of Traffic Flow. SpringerVerlag New York Inc., 1987.

[2] O. Masoud and N. P. Papanikolopoulos, "A novel method for tracking and counting pedestrians in real-time using a single camera," IEEE Transactions on Vehicular Technology, vol. 50, no. 5, pp. 1267–1278, Sep. 2001.

[3] A. Leykin and R. Hammoud, "Robust multi pedestrian tracking in thermal-visible surveillance videos," in Proceedings of CVPR Workshop 2006, Jun. 2006.

[4] J. Cui, H. Zha, H. Zhao, and R. Shibasaki, "Laser-based detection andtracking of multiple people in crowds," Computer Vision and Image Understanding, vol. 106, pp. 300–312, May 2007.

[5] Q. Chen, M. Gao, J. Ma, D. Zhang, L. M. Ni, and Y. Liu, "MOCUS: Moving object counting using ultrasonic sensor networks," International Journal of Sensor Networks, vol. 3, no. 1, pp. 55–65, 2008.

[6] "IRISYS people counter," available at http://www.irisys.co.uk/people-counting/.

[7] "PCW-2BX03 directional people counter," available at http://www.sensourceinc.com/.

[8] "Pyroelectric sensor," available at http://www.eco-computer.com/.

[9] "Acoustic slab sensor," available at http://www.eco-computer.com/.

[10] "LOTraffic," available at http://www.logobject.ch/.

[11] B.-R. Son, S.-C. Shin, J.-G. Kim, and Y.-S. Her, "Implementaiton of the real-time people counting system using wireless sensor networks," Internaitonal Journal of Multimedia and Ubiquitous Engineering, vol. 2, no. 3, Jul. 2007.

[12] S.-W. Lee, "A room-level indoor location system for smart houses," in Proceedings of SeNAmI 2009, Dec. 2009.

[13] Y. Taniguti, Y. Gouda, G. Hasegawa, and H. Nakano, "Bi-directional people counting with compound-eye binary sensors," IEICE SIG Notes (IN2009-71), pp. 17–22, Nov. 2009, in Japanese.

[14] S. Fujii, Y. Taniguti, G. Hasegawa, and H. Nakano, "Monte Carlo-based bi-directional pedestrian counting with compound-eye sensor," in Proceedings of ICCSII 2012, pp. 1–6, Dec. 2012.

[15] "RE-210," available at http://www.nicera.co.jp/.

[16] T. Teixeira, G. Dublon, and A. Savvides, "A survey of human sensing: Methods for detecting presence, count, location, track and identity," submitted to ACM Computing Surveys, 2012, available at http://www.eng.yale.edu/enalab/publications/human sensing enalabWIP.pdf.

[17] D. B. Reid, "An algorithm for tracking multiple targets," IEEE Transactions on Automatic Control, vol. 24, pp. 843–854, Dec. 1979.

[18] H. Pasula, S. Russell, M. Ostland, and Y. Ritov, "Tracking many objects with many sensors," in Proceedings of IJCAI 1999, pp. 1160–1167, 1999.

[19] S. Oh, S. Russell, and S. Sastry, "Markov chain Monte Carlo data association for general multiple-target tracking problems," in Proceedings of IEEE CDC 2004, pp. 735–742, Dec. 2004.

[20] N. Shrivastava, R. Mudumbai, U. Madhow, and S. Suri, "Target tracking with binary proximity sensors: Fundamental limits, minimal descriptions, and algorithms," in Proceedings of SenSys 2006, pp. 251–264, Nov. 2006.

AUTHOR PROFILES

Shuto Fujii received his B.E. degree from Osaka University, Japan, in 2012. He is currently a master's degree student. His research interests include pedestrian counting systems.

Yoshiaki Taniguchi received his B.E., M.E., and Ph.D. degrees from Osaka University, Japan, in 2002, 2004, and 2008, respectively. Since 2008 he has been an Assistant Professor at the Cybermedia Center, Osaka University. His research interests include wireless sensor networks, wireless mesh networks, and object tracking systems. He is a member of IEEE, IPSJ, and IEICE.

Go Hasegawa received his M.E. and Ph.D. degrees from Osaka University, Japan, in 1997 and 2000, respectively. From 1997 to 2000 he was a Research Assistant at the Graduate School of Economics, Osaka University. He is currently an Associate Professor at the Cybermedia Center, Osaka University. His research is in the area of transport architecture for future high-speed networks. He is a member of IEEE and IEICE.

Hirotaka Nakano received his B.E., M.E., and D.E. degrees from the University of Tokyo, Japan, in 1972, 1974, and 1977, respectively. He joined NTT Laboratories in 1977 and has been engaged in research and development of videotex systems and multimedia-on-demand systems. He was an executive manager of the Multimedia Systems Laboratory at the NTT Human Interface Laboratories from 1995 to 1999. He was head scientist of the Multimedia Laboratory at NTT DOCOMO until 2004, and is now a Professor at the Cybermedia Center, Osaka University. His research is in the area of ubiquitous networks. He is a member of IEEE, IEICE, and ITE.