# Countermeasure to Human Recognition Error for Agent-based Human Tracking System

Masaru Shiozuka†‡, Tappei Yotsumoto†,
Kenichi Takahashi‡, Masashi Nishiyama‡, Takao Kawamura‡, Kazunori Sugahara‡

†System Engineering Department,
Melco Power Systems Co. Ltd.
Kobe, Japan
email: {Shiozuka.Masaru@zd, Yotsumoto.Tappei@zb}.MitsubishiElectric.co.jp

‡Graduate School of Engineering,
Tottori University
Tottori, Japan
email: {takahashi, nishiyama, kawamura, sugahara }@tottori-u.ac.jp

*Abstract*—**Human monitoring systems are widely deployed in companies, schools and elsewhere for the prevention of crimes. Such systems require operators to monitor information sent from monitoring devices, such as cameras and/or beacon sensors. To reduce the burden of operators' work, we have proposed an automatic human tracking system based on mobile agent technologies. The system succeeded in tracking persons when Radio Frequency IDentifier (RFID) sensors were used as monitoring devices. However, it sometimes causes human recognition error when using a camera. In this paper, we propose a method to address human recognition error. In this method, an agent changes his/her behavior according to a distance computed from pictures taken by a camera. Experiments using the proposed method showed that the rate of successful human tracking improved even in an environment where human tracking error often occurred.**

***Keywords-Human Tracking; Mobile Agent; Camera; Sensor.***

## I. INTRODUCTION

As security measures of companies and our daily lives, various kinds of systems, such as an entrance control system for monitoring suspicious persons, have been introduced. However, if the number of sensors and tracking targets increases, tracking all targets becomes difficult. Therefore, we propose a mobile agent-based tracking system using neighbor relations of sensors in the environment where each sensor is located discretely [1]-[3]. This system consists of sensors, tracking nodes, mobile agents and a monitoring terminal. The system uses cameras and/or beacons as sensors. In this system, a node with a sensor analyzes data received from its sensor, therefore, the processing load of data analysis is distributed in each node.

One agent has the features of one target person. The agent moves among nodes by detecting the features of the target person. An operator can know the location of the target by checking the location of its corresponding agent. Since sensors are installed in discrete locations, such as an entrance and passage crossings, a person is often not caught in any sensors. Therefore, we proposed a method to predict which sensor may catch the target person next [1]. The method calculates neighbor nodes of each sensor based on the value of each sensor's detection range, the map of the floor and the locations of the installed sensors. Since the method enables us to calculate neighbor nodes, the system can predict which sensor may catch the target person next.

The system, however, sometimes fails to track a target person due to the uncertainty of sensors. Even if the system uses RFID sensors, the system sometimes fails to receive a signal from a RFID tag. Therefore, we have proposed a method to find hidden neighbor relations [2]. In this method, when an agent loses a target, a new bypass is constructed between the node where the target is lost and the node where the target is found. This method can achieve continuous tracking of a person.

When beacons or RFID are used as sensors, a unique ID is included in the signal from the sensor. Thus, a person is uniquely identified; we do not need to consider human recognition error. A system using cameras, however, causes human recognition error. When the system uses cameras, the system extracts the features of a person from a picture taken by a camera. Here, the features cannot always be extracted accurately. For example, when tracking a person with brown hair color, his/her hair color may be recognized as black under the intensity of the light. As a result, it may cause human recognition error.

Several research studies on human tracking using cameras have been proposed. Wenxi et al. [4] propose a method to predict the migration route of a person in a crowd by using high-order particle filter and online-learning. Jin et al. [5] propose a group structure to improve tracking accuracy in a situation when the detection ranges of cameras overlap. These are not applicable to a situation where sensors are installed discretely. Babenko et al. [6] and Zhang et al. [7] propose an online classifier to improve tracking accuracy of a single object. Cho et al. [8] propose a method to create neighbor relationships among cameras automatically. However, they require a central server to collect and manage data from cameras. If the number of cameras increases, the system requires expensive machines because of the increased computational cost.

In this paper, we propose a method to address human tracking errors. When we use a camera as a sensor, human recognition error occurs because a person is identified by the difference (hereafter, called "distance") between the features of a target and features extracted from a picture taken by a camera. Therefore, a person who is not a target may be recognized as the target; a person who is a target may not be recognized as being the target. To address such cases, we introduce the concept of *reliability*. Since the reliability is calculated from several pictures, agents can keep tracking even if the distance computed from a picture is accidentally low or high. In experiments, we confirmed that it is possible to track persons with high accuracy even in a situation where human recognition error occurs.

The remainder of this paper is structured as follows. Section II introduces the overview of our human tracking system. Section III proposes a method to correct human recognition error. In Section IV, the proposed method is evaluated, and Section V concludes the paper.

## II. Human Tracking System Based on Mobile Agent Technologies

We have developed an automatic human tracking system based on mobile agent technologies [1]. In this system, a mobile agent tracks one person called "target." All the targets are tracked by each agent automatically. An operator can know the location of each target by its corresponding agent.

### A. System Overview

Figure 1 shows the overview of the system. The system consists of targets, sensors, nodes, agents and a monitoring terminal. A target is a human tracked by agents. A node has a data analysis function and an execution environment for agents. An agent moves across nodes along the migration of a target. The location of a target is displayed on the monitoring terminal through the location of the agent.

### B. Tracking Flow

When a person comes into the detection range of the sensor, the corresponding node catches its signal. If the person is not tracked by any agent, the node generates an agent with his/her features e.g. facial features and beacon IDs. Hereafter, we call it as "a target agent." The target agent distributes its copies called "copy agents" to its neighbor nodes. Neighbor nodes are calculated by a method described in Section II-C. The set of a target agent and copy agents is called as "a group." When the target is detected in the group, a copy agent which detects the target becomes a new target agent. Thus, the target is tracked by the new target agent. The original target and copy agents are subsequently erased. The new target agent distributes its copy agents to its neighbor nodes. In these steps, a person can be tracked by agents.

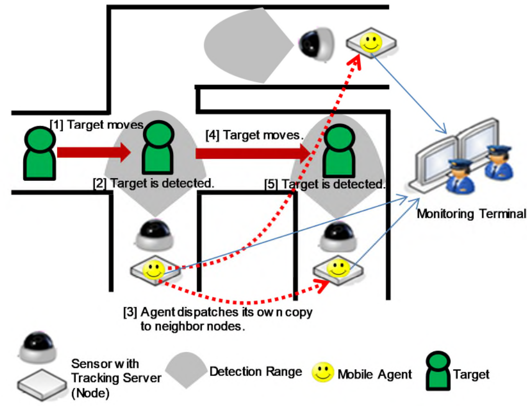


Figure 1. Overview of the Proposed System

### C. Method to Calculate Neighbor Nodes

To predict which sensors may detect a target next, we have proposed a method which calculates neighbor nodes based on each sensor's detection range and the locations where sensors are installed. In this method, the following points are defined:

- Branch points (passage crossings): $B_i$
- Sensor points (sensor locations): $S_i$
- Detection points (between two branch points, between two sensor points and between a branch point and a sensor point): $D_i$

Matrix $X$ of $|S| \times |P|$ is defined from the detection range of all sensors. S is a set of sensor points and P is a set of all points (branch points, sensor points and detection points). Element $X_{ij}$ of matrix $X$ is defined as (1).

$$X_{ij} = \begin{cases} 0, & \begin{array}{c} where\ detection\ range\ of\ sensor \\ S_i\ does\ not\ include\ point\ P_j. \end{array} \\ 1, & \begin{array}{c} where\ detection\ range\ of\ sensor \\ S_i\ includes\ point\ P_j. \end{array} \end{cases} \tag{1}$$

Next, we define an adjacency matrix $Y$ of $|P| \times |P|$. Element $Y_{ij}$ of matrix $Y$ is defined as (2).

$$Y_{ij} = \begin{cases} 0, & \begin{array}{c} where\ point\ P_i\ and\ point\ P_j \\ are\ not\ neighboring\ each\ other. \end{array} \\ 1, & \begin{array}{c} where\ point\ \ P_i\ and\ point\ P_j \\ are\ neighboring\ each\ other. \end{array} \end{cases} \tag{2}$$

When $E_{ij} \geq 1$ in (3), the neighboring sensor exists $(n-1)$ points away from the detection range of sensor $S_i$.

$$E = X \cdot Y^n \cdot X^T \tag{3}$$

TABLE I. SENSORS' DETECTION RESULT AND REAL EVENT

| Sensor \ Real Events | No Person Exists | Person P1 Exists | Person P2 Exists | Person P1 and P2 Exist |
|---|---|---|---|---|
| **Undetected** | True Detection | Non-Detection | Non-Detection | Non-Detection |
| **Person P1 is Detected** | False Detection | True Detection | False Detection | True Detection and Non-Detection (※1) |
| **Person P2 is Detected** | False Detection | False Detection | True Detection | True Detection and Non-Detection (※1) |
| **Person P1 and P2 are Detected** | False Detection | True Detection and False Detection (※2) | True Detection and False Detection (※2) | True Detection |

※1 One person is detected but another person is not detected.
※2 Other person except existing persons are detected
Note) False detection never occurs if we use a Beacon/RFID sensor

Even if the neighboring sensors can be calculated in (3), the number of points between the detection ranges of two sensors is unknown. In other words, $n$ is unknown. Therefore, the points which are not included in the detection range of all sensors are eliminated from matrix $X$ and $Y$. Matrix $X'$ is generated from matrix $X$ by eliminating all the points in column $j$ that satisfy (4).

$$\sum_{k=i}^{m} X_{kj} = 0 \qquad (4)$$

Further, $X_{ik}$ is set to 1 if $X_{ij} = 1$ and $X_{jk} = 1$. This prevents a route from being cut off by the elimination of a point. Similarly, matrix $Y'$ is generated from matrix $Y$ by eliminating all the points in column $j$ and row $j$. Then, the neighbor sensors can be calculated in (5).

$$E' = X' \cdot Y' \cdot X'^{T} \qquad (5)$$

In (5), we can find all neighbor nodes calculated from all tracking route.

### D. Issues To Be Tackled

The system can track a person continuously if sensors detect a target person correctly. However, a sensor has uncertainty, thus, it sometimes fails to detect a target.

Therefore, we have to tackle the uncertainty of the sensors. We, first, discuss the uncertainty of sensors by the comparison between a sensor's detection result and a real event, shown in Table I. Real events are put on the column, and sensors' detection results are on the row. In this table, "True Detection" means that a target is correctly detected. "False Detection" means that other person except existing person is detected. "Non-Detection" means that a target is not detected where the target exists. Tracking misses when "False Detection" or "Non-Detection" occurs.

When we use a beacon as a sensor, "False Detection" does not occur. Therefore, we have proposed a hidden neighbor relation in [2]. In experiments using the hidden neighbor relation, we have confirmed that the tracking accuracy improved. However, the problem of "False Detection" is remained. In the next section, we propose a method to improve tracking accuracy even if "False Detection" occurs.

TABLE II. EXAMPLE OF HUMAN RECOGNITION RESULTS

| Input Picture | Pictures Ordered by Distance | | | |
|---|---|---|---|---|
| P1_Front | P2_Right | P1_Right | P3_Back | … |
| | 2.760 | 3.208 | 3.476 | … |
| P1_Left | P1_Right | P3_Back | P2_Right | … |
| | 2.119 | 2.679 | 3.117 | … |

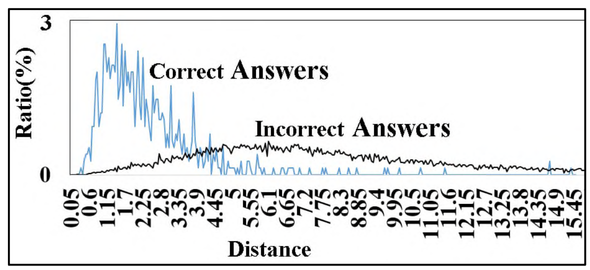


Figure 2. Distribution of Correct/Incorrect Answers

## III. DEALING WITH TARGET RECOGNITION ERROR

When we use a camera as a sensor, target recognition error occurs because a person is not uniquely identified. Therefore, a person who is not a target may be recognized as the target; a person who is a target may not be recognized as the target. The former causes "False Detection", and the latter causes missing the target by "Non-Detection." Also, two or more persons may be recognized as the same target. In this section, we, first, explain an example of target recognition error.

### A. Example of Target Recognition Error

As an example, we show a person recognition method proposed by Nishiyama [9]. It uses a picture database named SARC3D [10] included in PETA dataset [11]. The SARC3D consists of pictures of 50 different persons. Each person has four pictures taken from four directions, front, back, left and right. Therefore, the SARC3D has 200 pictures in total. 100 pictures are used as training data for parameters' setting. Table II shows a part of distance computed from an input picture and other pictures in remained 100 pictures. The smaller value means an input picture and other picture is similar. For example, P1_front picture (taken from the front of person P1) is most similar to P2_Right (taken from the right side of person P2), and second similar is P1_Right (take from the right side of person P1).

The distribution of distances are shown in Figure 2. Correct answer means the distance between two pictures of the same person. Incorrect answer means the distance between the pictures of other person. From Figure 2, we can see the distance of the same person tends to be low, and the distance of other person tends to be high. However, the distance of correct answers and incorrect answers overlap. Even if a distance is low, it is not always a correct answer. Even if a distance is high, it is not always an incorrect answer. In other words, when the distance between the pictures of other person becomes accidentally low, human recognition error occurs. This causes "False Detection."

*B. Reliability*

Each picture taken by a camera is certainly different even if they are the pictures of the same person. This sometimes causes that a distance computed from a person who is not a target is lower than a distance computed from a target person. As shown in Figure 2, the graphs of correct and incorrect answers have an overlapping part. That is, even if the distance is low, it may be an incorrect answer. This causes "False Detection." To avoid "False Detection," we introduce *reliability* which uses $n$ pictures instead of one picture. Reliability is defined as (9).

$$reliabilitiy(n) = 1 - \prod_{i=1}^{n}(1 - P(x_i)) \quad (9)$$

In (9), $x_i$ is a distance computed from a picture, $n$ is a number of pictures used for the calculation of reliability. When $n$ is 3, we use continuous three pictures taken by one camera to recognize a target or not. $P(n)$ is a function that returns a precision for a distance $x_i$. Since the function $P(n)$ depends on a person recognition method, we have to make function $P(n)$ based on a person recognition method. An agent recognizes a person as his/her target if equation (10) is satisfied.

$$reliability(n) > p_n \quad (10)$$

In (10), $p_n$ is the threshold value. The threshold value enables us to control "False Detection" rate. If the threshold value $p_n$ is high, "False Detection" decreases, but missing a target increases. The threshold value $p_n$ is low, "False Detection" increases.

In addition, even if the "False Detection" rate is low, two or more agents which are tracking different targets may recognize the same person as their each targets. In this case, "False Detection" should occur except on one agent. Therefore, we adapt group expansion described in the next section without the decision of their target. In the same reason, we adapt group expansion when two or more different persons are recognized as the same target.

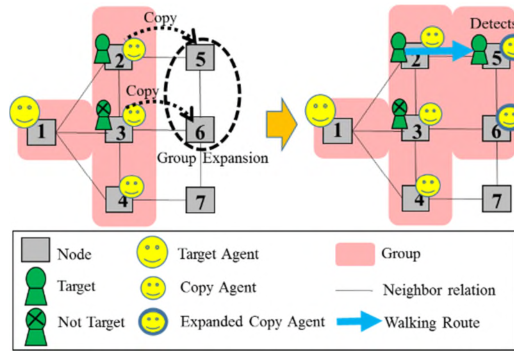


Figure 3. Example of Group Expansion

*C. Group Expansion*

If $p_n$ is high, "False Detection" decreases, but missing a target increases. To decrease the problem of missing a target, we expand a group. A group consists of a target agent in a node which found a target last, and copy agents in its neighbor nodes. The agents try to find a target within their nodes. When an agent misses a target, the target may go out of the monitoring area covered by their nodes. Therefore, we expand a group to cover the outside of the monitoring area. An agent misses a target in the following cases.

- Case1: Non-Detection occurs.
- Case2: Reliability is less than a threshold value.
- Case3: Two or more different persons are recognized as the same target.
- Case4: Two or more agents which are tracking different persons recognized the same person as their target.

We do not mention Case 1 in this paper, since this case is already addressed in [2]. We, first, mention Case 3. Here, we suppose that a target agent stays in node 1, copy agents stay in node 2, 3 and 4 in Figure 3. Then, both agents in node 2 and 3 find target candidates. Since a target is one person, we cannot determine which person in node 2 or 3 is the target. Therefore, when the target moves to node 5 or 6, the agents lose the target. Then, a group is expanded to cover neighbor nodes of node 2 and 3 (node 5 and 6 in Figure 3). Since node 5 and 6 are monitored by group expansion, the target can be tracked continuously. In a similar way, a group is expanded in Case 4.

Case 2 is different from Case 3 and 4. We cannot find expanded nodes because there is no person who is likely to be a target. Therefore, we make a group expansion condition. The group expansion condition is defined as (11).

$$reliability(i) > p_i \ (0 < i < n) \quad (11)$$

An agent uses $n$ pictures to calculate reliability. When reliability calculated from $n$ pictures of the target is

 ISBN: 978-1-61208-676-7 

accidentally low, the agent recognizes the target as a non-target. In equation (11), pictures less than *n* are used as group expansion condition. Therefore, the group expansion condition is satisfied, if the distance of one picture is high. When the group expansion condition is satisfied at some, a group is expanded to cover its neighbor nodes. By group expansion, missing rate of a target can be reduced.

## IV. EXPERIMENTS

We implemented a simulator to evaluate our proposed method. In this simulator, we evaluated whether persons can be tracked correctly in situations where person recognition error occurs. As a picture recognition method, we use the person recognition method proposed in [9], but we can apply any other person recognition method.

### A. Simulation Settings

Figure 4 shows the map used in this simulation. There are nine cameras installed in each node. Since our system is implemented in distributed manner, the number of cameras does not affect our system performance. Maximum eight persons are walking at the same time. Table III shows the walking route of each person. Each person moves between nodes in 5 seconds. For example, person P1 starts at node 1, reaches at node 2 in 5 seconds and finally reaches at node 9 in 40 seconds.

In this simulation, we fix *n* on 3, therefore, three pictures are used to calculate reliability. The threshold $p_n$ is set up to be "False Detection" rate under 5%.

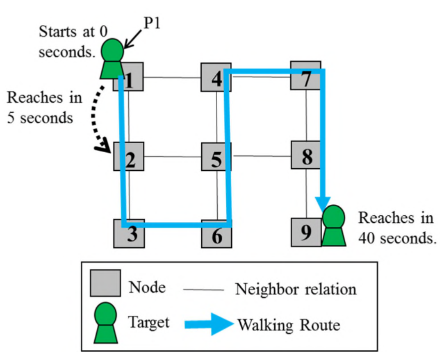


Figure 4. Simulation Map

TABLE III. WALKING ROUTE

| *Target ID* | *Walking Route* |
|---|---|
| P1 | 1→2→3→6→5→4→7→8→9 |
| P2 | 9→8→7→4→5→6→3→2→1 |
| P3 | 3→2→1→4→5→6→9→8→7 |
| P4 | 7→8→9→6→5→4→1→2→3 |
| P5 | 1→4→7→8→5→2→3→6→9 |
| P6 | 9→6→3→2→5→8→7→4→1 |
| P7 | 3→6→9→8→5→2→1→4→7 |
| P8 | 7→4→1→2→5→8→9→6→3 |

In this simulator, a distance computed from a picture is given as follows:

Rule1: If the target of an agent is in the detection range of a sensor, an agent randomly get a distance from the set of correct answers in Figure 2.

Rule2: If the target of an agent is not in the detection range of a sensor, an agent randomly get a distance from the set of incorrect answers in Figure 2.

In these rules, a target recognition error can be produced. For example, when a target is not appeared in, an agent gets distance from the set of incorrect answers according to rule 2. If the distance is low, the agent recognizes that the target is there even if the target does not exist. Then, a target recognition error occurs.

Regarding group expansion, we implemented Case 2 and 3 in Section III-C except Case 1 and 4.

### B. Tracking Results of Two Persons

In this simulation, two persons, P1 and P2 are walking in the map at the same time. Figure 5 shows the tracking result of P1. We can see an agent can track P1 exactly behind 3 seconds. The reason of being 3 seconds behind is that 3 seconds are necessary to take three pictures. Figure 6 shows the tracking result of P2. When P2 moves to node 4, the reliability did not exceed the threshold $p_n$. Therefore, the agent cannot follow with P2. However, copy agents are distributed to node 1 and 5 (Figure 7) which are neighboring nodes of node 4, since the group expansion condition is satisfied at node 4. After that, when P2 moves to node 5, P2 is detected at node 5. As a result, the agent can continue the tracking.

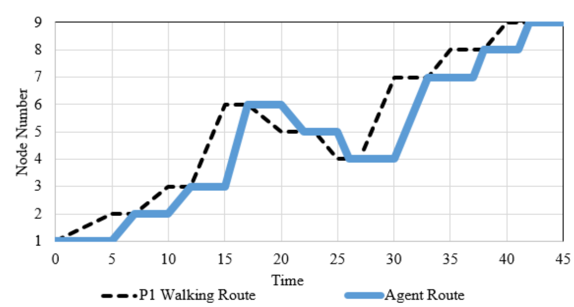


Figure 5. Tracking Result of P1

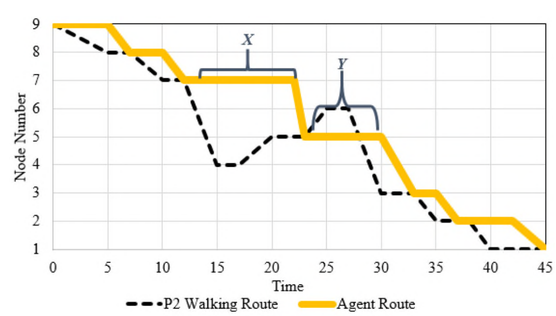


Figure 6. Tracking Result of P2

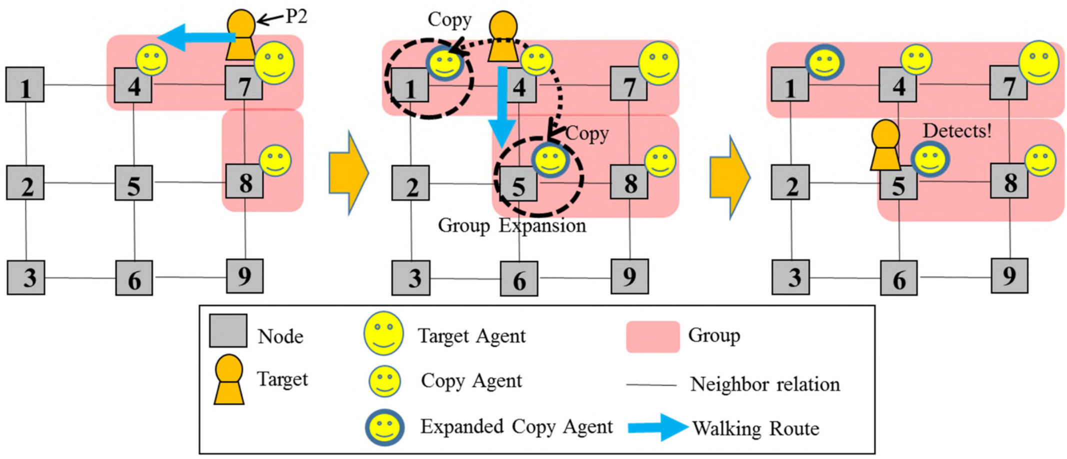


Figure 7. Tracking Results of P2 Group Expansion at Node 4

### C. *Tracking Results of One to Eight Persons*

Figure 8 shows the tracking results of one to eight persons. The results are the average of 20 simulations. Figure 8 shows the rate of that a target and its corresponding agent are on same node. As shown in Figure 8, even if the number of targets increases, the tracking success rate does not fall. In the case of 8 persons, the tracking success rate is 92.5%. For comparison, we make a system that regards a person with smallest distance as a target. The comparative system regards a person with smallest distance as a target. In this comparative system, when the number of persons increases, the tracking success rate falls greatly. In the case of 8 persons, the tracking success rate is 67%.

## V. CONCULSION AND FUTURE WORK

In this paper, we extend the method proposed in [2] to address target recognition error. In this extension, we introduced two concepts, reliability and group expansion. The simulation results show the success rate of target tracking is improved. We plan to evaluate the soundness of our method in a real environment.

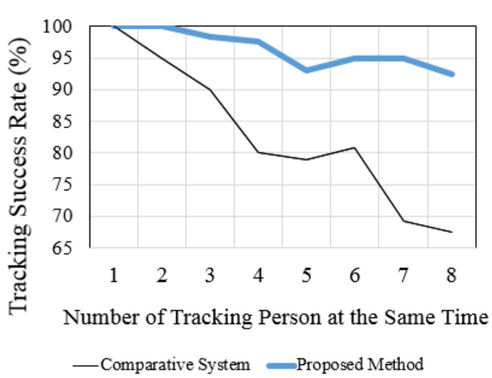


Figure 8. Tracking Result of One to Eight Persons

## REFERENCES


[1] T. Yotsumoto et al., "Automatic Human Tracking System using Localized Neighbor Node Cluculation," Sensors & Transducers, Vol. 194, No. 11, pp. 54-61, 2015.

[2] T. Yotsumoto, M. Shiozuka, K. Takahashi, T. Kawamura, and K. Sugahara, "Hidden neighbor relations to tackle the uncertainness of sensors for anautomatic human tracking," 2017 Second IEEE International Conference on Electrical, Computer and Communication Technologies (ICECCT 2017), Coimbatore, India, pp. 690-696, 2017.

[3] M. Shiozuka, T. Yotsumoto, K. Takahashi, T. Kawamura, and K. Sugahara, "Implementation Example with Ultra-Small PCs for Human Tracking System Based on Mobile Agent Technologies," 11th International Conference on Mobile Ubiquitos Computing, Systems, Sercies and Technologies (UBICOMM2017), pp. 73-78.

[4] L. Wenxi, C. Antoni, L. Rynson, and M. Dinesh, "Leveraging long-term predictions and onlinelearning in agent-based multiple person tracking," IEEE Transactions on Circuits and Systems for Video Technology, 25.3, pp. 399-410, 2015.

[5] Z. Jin and B. Bhanu, "Multi-camera Pedestrian Tracking using Group Structure," International Conference on Distributed Smart Cameras, Article No. 2, 2014.

[6] B. Babenko, M.-H. Yang, and S. Belongie, "Robust Object Traccking with Online Multiple Instance Learning," IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol. 33, Issue 8, pp. 1619-1632, 2011.

[7] L. Zhang and L. van der Maaten, "Preserving Structure in Model-Free Trackin," IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol. 36, No. 4, pp. 756-769, 2014.

[8] Y.J. Cho, S.A. Kim, J.H. Park, K. Lee, and K.J. Yoon, "Joint Person Re-identification and Camera Network Topology Inference in Multiple Camera," arXiv:1710.00983, 2017.

[9] M. Nishiyama et al., "Person Re-identification using Co-occurrence Attributes of Physical and Adhered Human Characteristics," 23rd International Coference of Pattern Recognition (ICPR), pp. 2086-2091, 2016.

[10] SARC3D, http://www.openvisor.org/sarc3d.asp, September, 2018.

[11] Y. Deng, P. Luo, C. Loy, and X. Tang, "Pedestrian attribute recognition at far distance," ACM Multimedia, pp. 3-7, 2014.