Regular Paper

Feature Data Distribution Methods for Person Re-identification using Multiple Cameras

Satoru Matsumoto[*], Tomoki Yoshihisa[**], Tomoya Kawakami[***], and Yuuichi Teranishi[*,****]

[*]Cybermedia Center, Osaka University, Japan
[**]Department of Data Science, Shiga University, Japan
[***]Graduate School of Engineering, University of Fukui, Japan
[****]National Institute of Information and Communications Technology, Japan
smatsumoto@cmc.osaka-u.ac.jp, yoshihisa@biwako.shiga-u.ac.jp

Abstract - Recently, public cameras are widely used and are deployed in various places. These multiple cameras can be used for tracking lost children or criminals. For person tracking, most systems transmit feature data of people such as feature values or person images to a server. The server compares the data with others and judges whether they are the same person. Artificial intelligence and numerical analyses techniques can be used for the comparison. However, in this conventional scheme, the computational loads of the server is proportional to the data amount that is transmitted to the server. This increases as the number of cameras increases. Hence, in this research, we propose a scheme for distributing the computational loads of the server arose in the conventional scheme. Moreover, we propose two methods to determine the timing for camera devices to transmits feature data to other cameras. We evaluate these proposed methods and compare their performances. The simulation results show that the average traffic for each camera device can be reduced significantly compared to that under the conventional scheme.

Keywords: public cameras, feature data, processing servers, peer-to-peer.

1 INTRODUCTION

Due to the recent trend of Society 5.0 and Smart city, the development of comfortable cities using IT technology has attracted great attention. Understanding how and when people through the city can be useful in solving various social issues, such as marketing and research on human flow. Therefore, obtaining the travelling routes of people moving around the city contributes to the comfortable cities. If feasible to track people using many security cameras deployed in towns and cities, we can track many people widely.

Some schemes to detect a person in multiple images obtained from multiple cameras that do not share the same field of view have been proposed. These schemes track people by identifying the same person recorded in other cameras. The process of determining the same person from multiple images obtained from multiple cameras is called Person Re-identification and has been studied in recent years. These research uses deep learning or deep distance learning [1-4] to obtain feature values with high computational power, as well as using unsupervised learning [5].

Various research has been conducted to improve the accuracy of person re-identification, such as research on methods using unsupervised learning [5, 6]. However, when identifying the same person from multiple images obtained from multiple cameras, a wide communication bandwidth is required if all the images are transmitted to the server. In addition, if all the information obtained from the cameras are transmitted to the server and the person re-identification process is performed on the server, the load on the server increases as the number of cameras and persons tracked increases. Even in the case of using cloud video analysis services, the load on the analysis server increases.

Hence, in this paper, we propose a person tracking method that does not concentrate the load on the server. In the proposed method, a camera network is built by multiple camera devices that can communicate with each other. When a person is captured in a camera's field of view, the feature data of the person image are calculated. The camera device then transmits the calculated feature data to the camera devices where the person is going to be captured next. The camera device that receives the feature data compares the feature data of each captured people with those received before and judges whether the captured person is captured before by other camera devices. For example, in the cases that a person is captured by a camera A and after that captured by a camera B, the person may move to the place where the camera B shoots after the place where the camera A shoots. By deploying many cameras and shooting wider area, the system can enable more accurate tracking. Since each camera device performs the process of person re-identification using the model in the camera, our proposed method has a large possibility to suppress the concentration of the load on the server as the number of cameras and persons increases.

Furthermore, to evaluate the traffic of communication is generated when people are tracked using our proposed method, we develop a simulator. In the simulator, we assume that the camera devices are deployed at each intersection in a grid-shape roads. We also assume that the people through the rads from the left top corner to others randomly. We compare the average communication traffic of the server under a conventional method and that of our proposed method. We confirm that our proposed method can distribute the load. The organization of the paper is as follows. Section 2 inscribes existing research on person re-identification, a problem

deeply related to this research. Section 3 inscribes the proposed method, and the evaluation results are shown in Section 4. Finally, we conclude the paper in Section 5.

2 RELATED WORK

Person re-identification is the problem of identifying the same person from images of people captured by multiple cameras that do not share the same field of view. Given a query image, the person re-identification system searches for a person identical to the query image in the gallery images, as depicted in Fig.1. Numerous research improved the accuracy of person re-identification. Some of them consider person re-identification as a classification problem in which each person in a gallery image is a different class or not and use the SoftMax loss function to train the model. Others use distance learning such as triplet loss, etc. [7-10].

Person re-identification is expected to have a wide range of applications in computer vision, such as surveillance, behavior analysis, and person tracking. But on the other hand, it has a major problem. When using multiple person images captured by multiple cameras that do not share the same field of view to perform person re-identification, the following inter-camera gaps are unavoidable due to the nature that the person images used were captured by different cameras [8].

· Variety of perspectives
· Variety of lighting
· Variety of resolutions for captured people

The variety of perspectives refers to the fact that the characteristics of postures and the characteristics of looks change due to the different angles at which the people are captured in each camera. Variety of lighting refers to the changes in the lighting conditions in cameras' field of views depending on the camera positions and the times when people are captured. The appearances of people captured change under another lighting, such as the appearance of colors, etc. Variety of resolutions for captured people refers to the changes in the size of the bounding boxes for captured people. This changes the resolution of the resulting person image. The variety of resolutions also makes person re-identification difficult in that the resolution of a person captured in faraway positions is relatively low. Therefore, person re-identification systems that can give a higher accuracy even when the influences of these varieties are large.

Numerous research efforts have endeavored to mitigate these challenges. In [7], an adversarial network is used to obtain a more accurate feature representation that eliminates gaps between cameras as much as possible. The method proposed in [9] uses StarGAN to transform the styles of people in images. The method transforms the images of the people captured by a camera device to the images that consider the shooting conditions (background, lighting, etc.) of other cameras, then it uses these images as training data to reduce the influence of gaps between cameras. Also, there is a study that investigate how the variety of viewpoints affects the accuracy of person re-identification, as in [8].

Although numerous research has been conducted to reduce the influence of above differences in conditions between cameras, the following problems still exist.

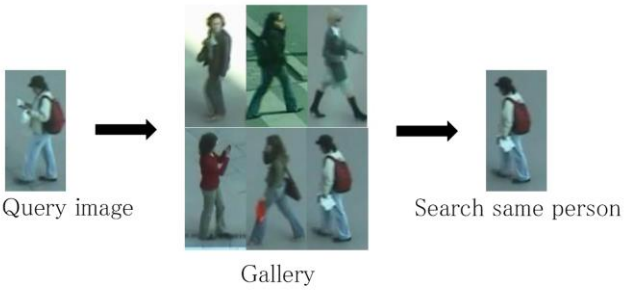


Figure 1: Overview of person re-identification

· Generating pedestrian images using GAN is too time consuming.

· Performance is significantly degraded when multiple people are captured in the field of view.

· Because model learning relies on external features of clothes, which occupies a large area of the human body, performance deteriorates significantly when a human's cloth changes during the process or when there are multiple people wearing the same clothes.

For the case where the camera images overlap, [11] performs partial figure re-identification using local features. Systematically investigating the impact of clothing changes on the accuracy of existing re-identification models, [12] generates pedestrian images with different attire to address this challenge. In [13], a method that person re-identification with removing the external information of clothes and focuses on body shape information is proposed.

However, the systems that adopt these existing methods need to collect all camera images to a computational server. This causes a large communication and processing loads on the server. Even in the traditional approach, wherein cameras solely transmit feature data of identified individuals to the server, the computational loads concentrated on the server. We aim to relief this loads for person re-identification in the paper.

3 PROPOSED METHOD

In this section, we first provide an overview of the proposed person tracking method. After that, we explain the detail.

3.1 Summary

Authors considered the idea of tracking a person through surveillance cameras in a city or facility using a conventional re-identification method, as explained in Section 2. In this case, the method involves transferring images captured by cameras to a server via a computer network. The subsequent processing of person re-identification on the server requires a large amount of communication traffic for image transmission. Moreover, the methods in which information obtained from the video is transferred to the server and the person re-identification process is performed on the server increases the load on the server as the number. In this case, the method of transferring the images captured by the cameras to a server via a computer network and processing the person re-identification on the server requires a large communication traffic for the transmission of the images.

Moreover, the methods in which information obtained from the video is transferred to the server and the person re-identification process is performed on the server increases the load on the server as the number.

The communication and the processing loads of the server increase in proportional to the number of the cameras. Therefore, the server's load becomes excessively high to track people in wide area. The authors propose a person tracking scheme to solve these problems in which features are transmitted among cameras. In our proposed method, a camera network is built using multiple camera devices that can communicate with each other, and the travelling paths of people in the target area are tracked by repeatedly transmitting and receiving feature data between camera devices and re-identifying people. The load on the server itself can be distributed to the clients while the sum of the load is almost the same as the load in the centralized case.

If re-identification fails, the system cannot track the person. Thus, the tracking performance can deteriorate compared with the system that a server manages all the cameras. However, our proposed system can distribute the communication and processing loads arose on the server in the above system.

3.2 Tracking Method

In this section, we describe the process flow of person tracking using camera device network. The proposed method is based on the following four assumptions.

· All camera devices that are connected to the camera device network can communicate with each other and transmit feature data.

· All camera devices have a neural network model that calculate the feature values of a captured person. The input of the model is him/her image. Each camera device gets the images from their connected cameras.

· The locations and the angles of the cameras are fixed, and the positioning of all cameras is assumed to be known in advance.

· All camera devices can estimate the direction of movement of a person using the coordinate and the interframe information.

Under the above assumptions, the camera devices connected to the computer network track the travelling path of a person by repeatedly transmitting feature data and judging whether the person is the same person. The following is an overview of the process flow when a person is successfully tracked between Camera A and B.

1. Camera A captures a new person X.
2. Camera A detects a person, acquires a person image, and computes the feature of the person X using a neural network model.
3. The destination camera device is determined by the destination determination method (detailed in Section 3.4) and the feature X is transmitted.
4. Camera B adds the feature X to the gallery.
5. A person moves and is captured by Camera B.

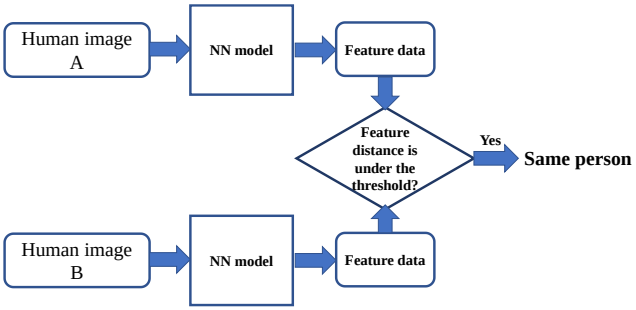


Figure 2: An image of a person re-identification process

6. Camera B computes the feature values and compares them with the feature values X in the gallery to determine that they are the same person or not.
7. The fact that the person captured by Camera A was also captured by Camera B indicates that the person moved from A to B.

Figure 2 shows an image of a person re-identification process. In Fig. 2, there are two separate images of people on the left side, and they are input to the same neural network model (NN model). The distance between the output features is calculated. The distance is between the features is used to judge whether the persons in the images are the same person or not.

3.3 Processes for Each Camera

The flow of processes executed by each camera device is shown below.

1. A person is captured by the camera.
2. Obtain bounding boxes and calculate features with NN models.
3. Person identification by comparing the calculated features with those in the gallery.
4. If these match, go to 7.
5. If these do not match, the feature data are transmitted to a camera device that is determined using the destination determination method (see Section 3.4) because the person is a newly detected person. The received camera device adds the feature data to the gallery.
6. Return to 1.
7. Notifies the server that a person has been detected.
8. The feature data of the person are again transmitted to the camera device determined using the destination determination method. The received camera device adds the feature data to the gallery.
9. Return to 1.

3.4 Destination Determination Method

In this section, we describe a method for determining the destination camera device for transmitting feature data to another camera. When a person is captured by one camera, it is assumed that its feature data needs to be transmitted to neighboring camera devices to track the person. This is because the cameras neighboring to the camera device that a

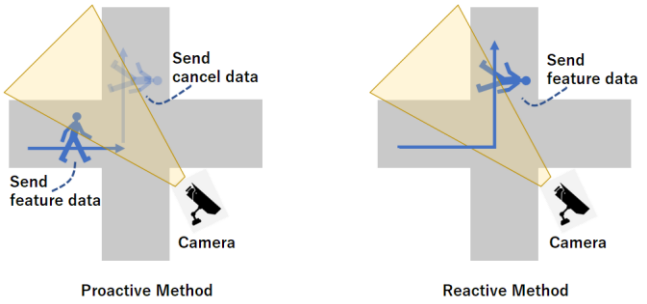


Figure 3: Time to start communication under our proposed method.

person captured is likely to be captured in the next. However, in the method where the feature data are transmitted only to the neighboring cameras, there is a possibility that the tracking of a person fails if the neighboring cameras fail to detect the person. One of the solutions for avoiding the failures is transmitting the feature data to further neighboring camera devices (the neighboring camera devices of the neighboring camera devices, etc.). Therefore, in the proposed method, we introduce a parameter *N* that indicates the number of the communication hops from the source camera device to transmit the feature data.

As described in Section 3.2, camera devices can predict the direction of moving people, and therefore, it is possible to limit the transmission destinations by using the direction. That is, the direction of movement can be used to limit the transmission destination. The transmitted feature data are deleted after a certain time has elapsed, preventing feature data that are not used for tracking from remaining in the gallery.

Based on the above approach, we propose two types of methods for determining the transmission destination. The image of each method is shown in Fig.3. The first one is to transmit feature data to all the neighboring cameras within *N* hops when a person is captured by a camera device. The value of the parameter *N* influences the success rate of the person tracking. However, it is difficult to get the success rate by mathematical analysis from the value of *N*. Therefore, *N* should be determined so that the success rate satisfies the application requirement by the trial and error. The transmission timing is when the moving direction of the person is predicted. After the reception, the camera devices that are not likely to capture the person need to delete the feature data from the gallery (the proactive method). The other one does not transmit feature data when a person is captured by a camera device but transmits feature data to the camera devices that exist in the destination direction when the direction of the person is predicted (the reactive method).

3.4.1 Proactive Method

The flow of the proactive method is shown in Fig. 4. The gray areas in the figure represent roads. The people walk on those areas. For simplicity, the roads are grid-shaped as shown in the figure, but the same process can be applied to roads that are not grid-shaped. The camera devices are assumed to be located at each intersection, and the locations of the camera devices are marked with the numbers (1 to 6).

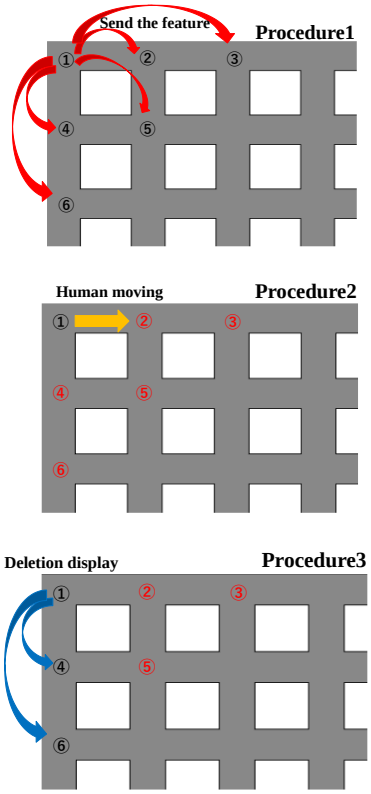


Figure 4: The flow of the proactive method

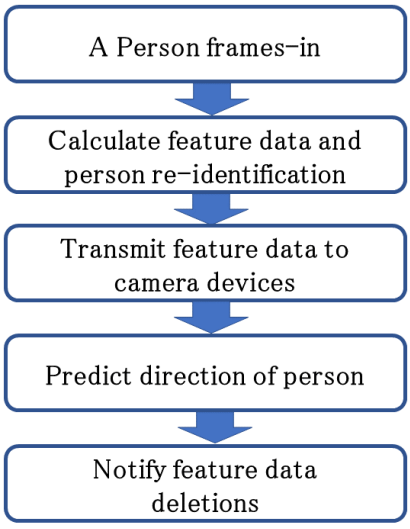


Figure 5: Processes for each camera device in the proactive method

The parameter *N* is set to 2, which indicates how many cameras are to transmit the feature values to the next camera.

Procedure 1 shows how the features are transmitted when a person is detected by Camera 1. Camera 1 transmits the feature data of the person to the surrounding *N* (= 2) camera devices when it detects a person. (Cameras with red numbers are the those hold the feature data.)

In Procedure 2, the person moves from the area that Camera 1 shoots to the area that Camera 2 shoots. Camera 1 judges that Camera 2 is the camera that may capture the person in the next based on its direction.

In Procedure 3, Cameras 4 and 6 are notified to remove the feature values from the gallery. This avoids the cameras that are unlikely to capture the person from continuing to have the feature values and reduces the number of candidates for the person re-identification.

Figure 5 shows the processes for each camera device in the proactive method. In the proactive method, when a new person is captured in the field of view of a camera, it calculates the feature values of the person. Then, the camera device determines whether the captured person is the same person that other camera devices capture before, by calculating the distance among feature values. When it finds the same person, it transmits only the information that the person was captured to the server. In the proactive method, after that, the camera device transmits the feature data of the captured person as well as own Camera ID to N (at maximum) neighboring camera devices. In this case, the system can avoid duplicate transmissions because it is possible to find the camera devices to which the feature data has been transmitted in the past from the list of camera IDs. If the same person is not found, it is assumed that the person is a new person and the feature data is transmitted to all camera devices to N (at maximum) neighbors. If the direction of the person is predictable from the direction and the location information at the time of frame-out, the number of galleries for person re-identification can be reduced by notifying the camera devices to delete the feature data stored that is not likely to capture the person.

3.4.2 Reactive Method

The flow of the reactive method is shown in Fig. 6. The road and the camera devices deployment are the same as the example for the proactive method in the previous subsection. Unlike the proactive method, the reactive method starts transmitting feature data after the moving direction of the person is found.

Procedure 1 shows the movement of the person from the area that Camera 1 shoots to that of Camera 2. In Procedure 2, the feature values are transmitted only to the camera device that exist in the direction of the person when Camera 1 detects it. We assume that the direction is predictable based on the travelling path of the person in the camera's field of view, such as the trajectory of the person and the position at which the person frames out.

The reactive method has the advantage of reducing the amount of communication because each camera device predicts the direction in which a person is moving and transmits the feature data only to the camera devices that exist in the direction. On the other hand, if the direction of the person cannot be predicted correctly, the feature data are not transmitted to the camera devices in the direction of the person, thus the tracking fails. If the terrain is complex, or if it is considered difficult to correctly predict the direction of a person due to the positional relationship among cameras, the probability of tracking failures can be high.

Figure 7 shows the processes for each camera device in the reactive method. In the reactive method, as in the proactive method, each camera device calculates feature data and re-identifies people when a new person is captured in the field

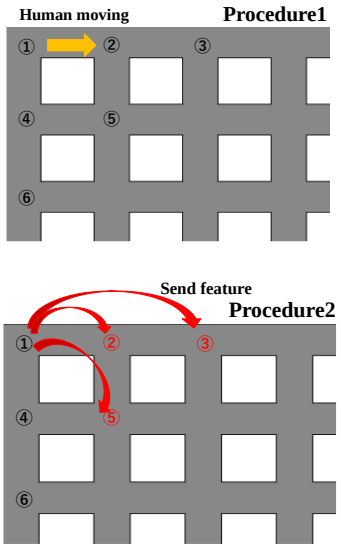


Figure 6: The flow of the reactive method

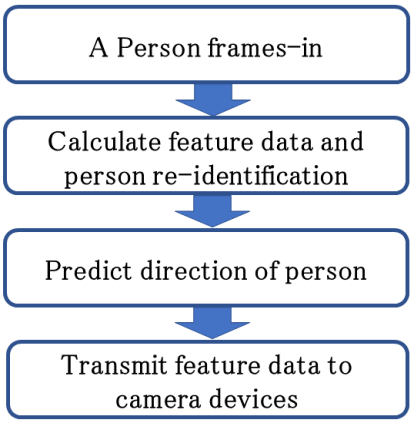


Figure 7: Processes for each camera device in the reactive method

of view. However, the feature data are not transmitted immediately, but only to the N neighboring cameras in the direction of their movement after predicting them based on their trajectories.

4 EVALUATION

To evaluate the amount of communication traffic generated when tracking a person under our proposed method, we created a simulator and measured the performances. This section inscribes the simulator specifications, evaluation items, and the results. Because there are no existing methods that transmit feature data in camera networks, we show only the performance of our proposed method.

4.1 Simulation Specifications

To systematically evaluate the performance of our proposed methods, we assume that the roads are grid-shape as shown in Fig. 8. A camera network is built with camera devices that can communicate with each other and are located at each intersection.

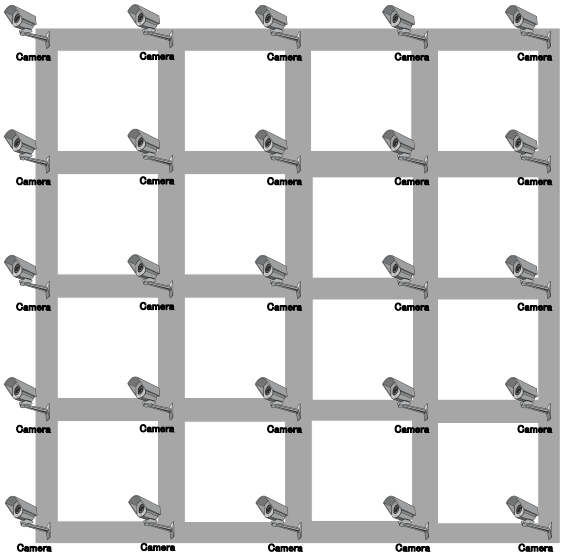


Figure 8: A map for the simulation
(25 camera devices)

All camera devices have a neural network model for feature extraction. The input data of the model is the face images of the humans recorded by the camera devices. The output data of the model is the feature values of the inputted face images. Accordingly, we assume that each camera device judges whether the persons in the images are the same person based of the distance values between their feature values. The feature values are the output data of the model. Regarding about the features used in the neural networks, they depend on the models.

We use three different maps to simulate various map sizes, as shown in Fig. 8. The figure shows cameras (assuming these can capture both vertical and horizontal streets) arranged in a grid where the square of the number of streets is the number of intersections. One section of the grid is fixed by 10 meters. The map becomes larger as the number of cameras increases.

4.1.1 Parameters

We change the following five parameters in the simulator.

- The parameter to determine the number of the camera devices that receive the feature data. When the value is *N*, the feature data are transmitted to *N* neighboring camera devices.
- The number of persons flowing into the tracking area per a second.
- The number of camera devices deployed in the tracking area was assumed to be either between 4 and 49.
- Since a person entering an intersection is not always detected by the camera, the detection probability can be changed as a parameter ranging from 0.0 to 1.0. This value depends on perspective, lighting, and resolutions in real situations, but these conditions are various and thus we give the probability as a parameter.
- We establish the communication bandwidth allocated for transmitting feature data, facilitating the calculation of transmission delay time.

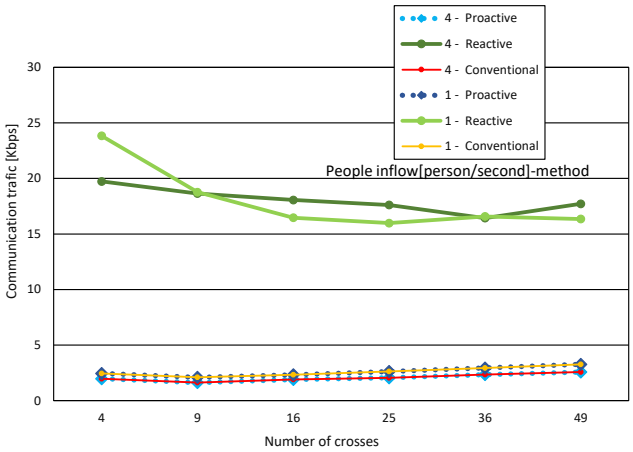


Figure 9: Communication traffic changing the number of crosses (number of camera devices) and the people inflow

4.1.2 Performance Indexes

As one of the indexes of the communication load on the cameras, we use the amount of communication traffic. The communication traffic of the feature data generated when a person moves in a map under the conditions of set parameters. Our developed simulator can calculate and output the delay time required for transmission by setting the communication bandwidth. We calculate the success rate of tracking from the delay. If the delay is longer than the time needed to move a person one block, the tracking fails.

4.1.3 Person Travelling Model

People walk at a speed of 1 meter per a second. Since one section of the grid is 10 meters long, the time between one camera capturing the person and the next is 10 seconds. A person enters the map at the upper left corner and exits at the lower left, the upper right, and the lower right corners. The number of people exiting from each exit is adjusted to be the same.

4.2 Evaluation Results

We get the results under the following situations.

4.2.1 Evaluation Items

The change in the communication traffic under the condition of different number of the camera devices and different people inflow.

The change in the tracking success rate in the proposed method changing the communication bandwidth. The tracking success rate is the rate that the number of the people that are tracked from the time to enter the tracking area to the time to exit divided by the number of the entered people. When the communication delays among the camera devices are all shorter than the one block travelling time of a person, the person is tracked in the tracking area.

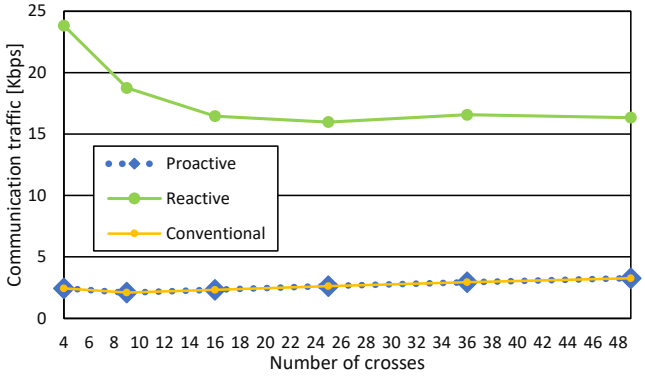


Figure 10: Communication traffic changing the number of crosses (number of camera devices)

Comparison of the average communication traffic of the server in a conventional method, in which the camera devices transmit the feature data to the server with that under our proposed method.

4.2.2 Communication Traffic for Transmitting Feature Data

We evaluated how the communication traffic changes when the number of cameras is changed to between 4 and 49, and when the number of people per second is changed to 1, 2, 3, or 4. In this evaluation, we assume that the communication traffic for one set of feature values is 22.586 [Kbit] assuming a 50-dimensional vector of float32 as the feature data. In addition, three 16-bit regions are allocated to record IDs for identifying the person and the cameras that have passed through. This results in a total of 48 bits of header information being appended. This value is the average data size of the actual features in the data set. Also, this is an example setting. The detection probability for each camera device is set to 0.8.

The results are shown in Fig. 9. The vertical axis represents the communication traffic for transmitting feature data. The unit is Kbps. The horizontal axis represents the number of crosses. From the results, it can be considered that there is a proportional relationship between the number of people and the communication traffic. In the proposed method, the communication traffic ranges from 15 [Kbps] to 24 [Kbps] when the number of the camera devices is between 4 and 49 and the number of people per second is between 1and 4. The number of cameras can be calculated from the number of crosses, as in Fig. 8.

Figure 10 shows a graph of the communication traffic per number of cameras when the number of people per second is set to 1. The vertical axis indicates the communication traffic, and the horizontal axis represents the number of crosses. From this graph, it can be considered that there is a proportional relationship between the number of cameras and the number of transmissions. When the number of crosses exceeds 4, the communication volume reaches a certain limit, which, according to the experimental results of proactive method, is 15.9 [Kbps] to 16.4 [Kbps]. In this experiment, camera bandwidth was constant at 4.6 [Kbps]. It is considered that if the number of cameras is increased, after the start of the simulation, the communication delay will increase, and

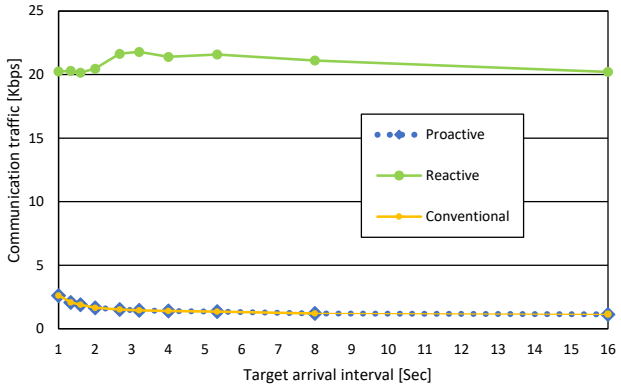


Figure11: Communication traffic of changing the target arrival interval

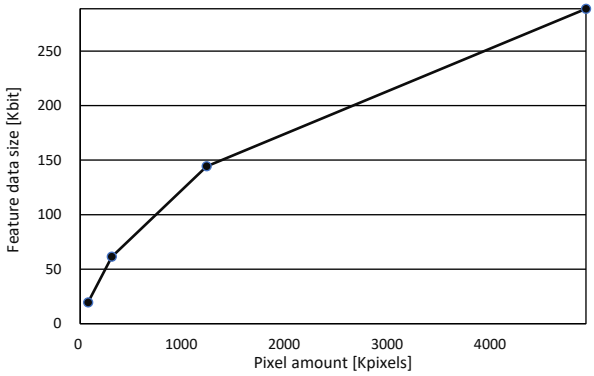


Figure12: Number of pixels and features

the tracking will not be successful. The communication traffic seems to have reached a certain limit.

4.2.3 Tracking Success Rate

To track a person without tracking failures due to latency, it is necessary to provide more bandwidth than the amount of communication generated. If the amount of communication per second generated by tracking exceeds the bandwidth provided, the delay in transmitting feature data will increase as tracking continues, and the delay will diverge to infinity.

For example, if the number of the camera devices is 20 and the number of people per second is 1, the amount of communication generated by the proactive method is 73.9 [Kbps]. If the bandwidth is only 46 [Kbps], the tracking of a person travelling at the beginning will succeed, but the tracking of a person travelling after a certain time will not succeed because the transmission delay will be too large. Figure 11 shows the simulation result for this situation. The horizontal axis represents the number of people per second, and the vertical axis is the total of average communication traffic for successful tracking. Assuming that the communication protocol is LoRa, we set the bandwidth by 46 [Kbps]. One of the merits of LoRa is low power consumption. LoRa can contribute to the recent trend of energy saving. Therefore, we assume the system environment in that such an energy saving communication protocols are used. These protocols unfortunately have a drawback that the communication speed is also low. The detection probability is set to 0.9. The success rate of the tracking becomes 0 when the number of people flowing into the tracking area increases

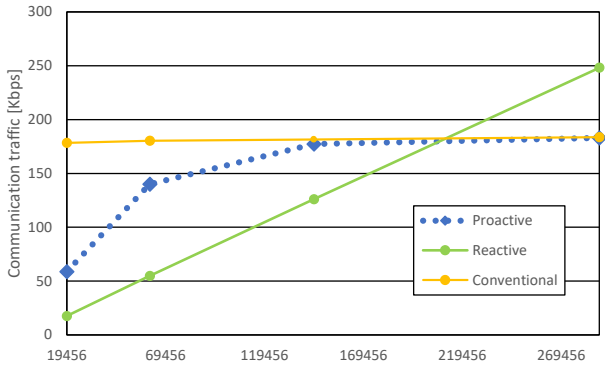


Figure13: Communication traffic for features by resolution

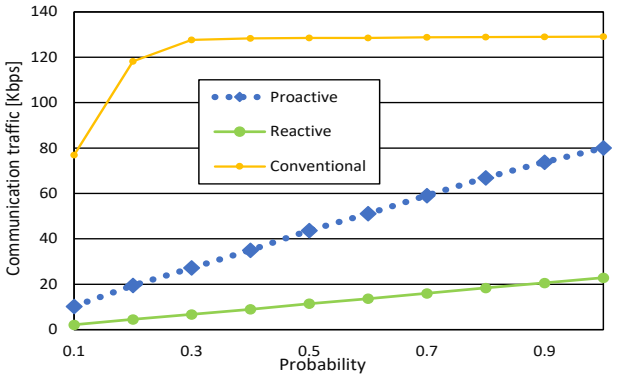


Figure 14: Comparison of communication traffic

and the amount of communication exceeds the bandwidth. This indicates that if the bandwidth is not sufficient for the number of people through the area, the tracking will fail due to delay.

4.2.4 Features Data Size by Resolution

To compare the amount of communication by features according to the number of pixels, the average feature data were calculated for each image size of 320 × 240, 640 × 480, 1280 × 960, and 2560 × 1920, assuming face recognition using the OpenCV library's cascade classifier. The feature data sizes for each of the four resolutions are shown in Fig. 12. The average communication traffic (amount of data received per unit time) for the proposed method was simulated and compared. The video bandwidth was set to 460 [Kbps], which is 10 times the roller video bandwidth. The arrival time interval was set to 1 second. The simulation results are shown in Fig. 13. The horizontal axis is the average feature data size. The vertical axis is the communication traffic, which ranged from 17.5 [Kbps] to 248.1 [Kbps]. From this figure, it can be observed that the proposed reactive method can communication traffic according to the feature data size if the video bandwidth is 46.0 [Kbps], whereas the conventional method can only perform to 58.7 [Kbps] to 183.7 [Kbps].

4.2.5 Comparison of Communication Traffic

We simulated and compared the average communication traffic of the server (the amount of data received per unit of time) and that under the proposed method. This traffic arises when all the feature data of the people captured by a camera device are transmitted to the server or other camera devices. The simulation results are shown in Fig. 14. The horizontal axis is the person detection probability explained in Section 4.1.1.

The communication traffic for each camera device in our proposed method is reduced about 12th at most compared to the average communication traffic under the conventional method, in which the server identifies the same person on the server. This indicates that the load that was concentrated on the server in the conventional method is distributed to each camera device under our proposed method.

5 CONCLUSION

A human tracking scheme in which each camera device transmits the camera image to a server causes a large communication and processing loads on the server. This lengthens the delay for tracking and deteriorates the tracking success rate. Hence, in this research, we proposed a human tracking method in which each camera device transmits feature values of captured people among camera devices. We focus on the problem that the processing load of the server increases in proportional to the number of the cameras, not the absolute value of the processing load itself. We proposed two methods to determine the timing for camera devices to transmits feature data to other cameras. We developed a simulator for the evaluation and simulated the situation in that the number of the camera devices is between 4 and 49, and the tracking area is a grid-shape roads. From the simulation results, we have found a possibility that it is possible to significantly reduce the average traffic per camera device compared to the average traffic on the server.

Our future work includes the evaluations of the recognition rate and the comparison between that of centralized case and that of the decentralized case. Moreover, we will focus on the processing load reduction of the cameras in the future.

ACKNOWLEDGEMENT

This research was supported by a Grants-in-Aid for Scientific Research(C) numbered 21H03429, 20K11829, 20H00584 and by G-7 Scholarship Foundation.

REFERENCES

[1] D. Wu, S.-J. Zheng, X.-P. Zhang, C.-A. Yuan, F. Cheng, Y. Zhao, Y.-J. Lin, Z.-Q. Zhao, Y.-L. Jiang, and D.-S. Huang, “Deep learning-based methods for person re-identification: a comprehensive review”, Neurocomputing, vol. 337, pp. 354-371 (2019).

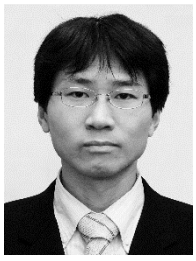
[2] S. Liao and L. Shao, “Graph sampling based deep metric learning for generalizable person re-identification”, arXiv:2104.01546, 11pages (2021).

[3] M. Ye, J. Shen, G. Lin, T. Xiang, L. Shao, and S. C. Hoi, “Deep learning for person re- identification: A survey and outlook”, IEEE Transactions on Pattern Analysis and Machine Intelligence, 20pages (2021).

[4] W. Wu, D. Tao, H. Li, Z. Yang, and J. Cheng, “Deep features for person re-identification on metric learning”, Pattern Recognition, vol. 110, 12pages (2021).
[5] T. Isobe, D. Li, L. Tian, W. Chen, Y. Shan, and S. Wang, “Towards discriminative represen-tation learning for unsupervised person re-identification”, in Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV), pp. 8526-8536 (2021).
[6] C. Yang, F. Qi, and H. Jia, “Survey on unsupervised techniques for person re-identification”, in 2021 2nd International Conference on Computing and Data Science (CDS), pp. 161-164, (2021).
[7] Y. Li, R. Xue, M. Zhu, J. Xu, and Z. Xu, “Angular triplet loss-based camera network for reid”, in 2021 International Joint Conference on Neural Networks (IJCNN), pp. 1-7 (2021).
[8] X. Sun and L. Zheng, “Dissecting person re-identification from the viewpoint of the viewpoint”, in Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 608-617 (2019).
[9] Y. Lin, Y. Wu, C. Yan, M. Xu, and Y. Yang, “Unsupervised person re-identification via cross-camera similarity exploration”, IEEE Transactions on Image Processing, vol. 29, pp. 5481- 5490 (2020).
[10] G. Wang, S. Gong, J. Cheng, and Z. Hou, “Faster person re-identification”, in European Conference on Computer Vision, pp. 275-292 (2020).
[11] X. Zhang, Y. Yan, J.-H. Xue, Y. Hua, and H. Wang, “Semantic-aware occlusion-robust net- work for occluded person re-identification”, IEEE Transactions on Circuits and Systems for Video Technology, vol. 31, no. 7, pp. 2764-2778 (2021).
[12] F. Wan, Y. Wu, X. Qian, Y. Chen, and Y. Fu, “When person re-identification meets change-ing clothes”, in Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops, pp. 830-831 (2020).
[13] X. Qian, W. Wang, L. Zhang, F. Zhu, Y. Fu, T. Xiang, Y.-G. Jiang, and X. Xue, “Long-term cloth-changing person re-identification”, in Proceedings of the Asian Conference on Computer Vision, 17pages (2020).

(Received: June 1, 2023)
(Accepted: December 20, 2023)

Satoru Matsumoto
received his Diploma’s degrees from Kyoto School of Computer Science, Japan, in 1990. He received his Master’s degree from Shinshu University, Japan, in 2004. From 1990 to 2004, he was a teacher in Kyoto School of Computer Science. From 2004 to 2007, he was Assistant Professor of The Kyoto College of Graduate Studies for informatics. From 2007 to 2010, he was Assistant Professor of Office of Society Academia Collabo-ration, Kyoto University. From 2010 to 2013, he was Assistant Professor of Research Institute for Economics & Business Administration, Kobe University. From 2015 to 2016, he was a specially appointed assistant professor of Cybermedia Center, Osaka University. From April 2016 to September 2016, he became a specially appointed researcher. Since November 2016, he became an assistant professor. His research interests include distributed processing systems, rule-based systems, and stream data processing. He is a member of IPSJ, IEICE, and IEEE.



Tomoki Yoshihisa
received the Bachelor's, Master's, and Doctor's degrees from Osaka University, Osaka, Japan, in 2002, 2003, 2005, respectively. Since 2005 to 2007, he was a research associate at Kyoto University. In January 2008, he joined the Cybermedia Center, Osaka University as an assistant professor and in March 2009, he became a full professor at Shiga University from April 2023. His research interests include video-on-demand, broadcasting systems, and webcasts. He is a member of the IPSJ, IEICE, and IEEE.

Tomoya Kawakami
received his B.E. degree from Kindai University in 2005 and his M.I. and Ph.D. degrees from Osaka University in 2007 and 2013, respectively. Since April 2022, he has been an associate professor at the University of Fukui. His research interests include distributed computing, rule-based systems, and stream data processing. He is a member of the Information Processing Society of Japan (IPSJ), the Institute of Electronics, Information and Communication Engineers of Japan (IEICE), and IEEE.

Yuuichi Teranishi
received his M.E. and Ph.D. degrees from Osaka University, Japan, in 1995 and 2004, respectively.
From 1995 to 2004, he engaged Nippon Telegraph and Telephone Corporation (NTT).From 2005 to 2007, he was a Lecturer of Cybermedia Center, Osaka University.From 2007 to 2011, he was an associate professor of Graduate School of Information Science and Technology, Osaka University.Since August 2011, He has been a research manager and project manager of National Institute of Information and Communications Technology (NICT).He received IPSJ Best Paper Award in 2011.His research interests include technologies for distributed network systems and applications.He is a member of the IPSJ, IEICE, and IEEE.