International Journal of Informatics Society

02/24 Vol.15 No.3 ISSN 1883-4566

Informatics Society

Editor-in-Chief: Hiroshi Inamura, Future University Hakodate

Associate Editors: Katsuhiko Kaji, Aichi Institute of Technology
Kei Hiroi, Kyoto University
Yoshia Saito, Iwate Prefectural University
Takuya Yoshihiro, Wakayama University
Tomoki Yoshihisa, Shiga University

Editorial Board

Hitoshi Aida, The University of Tokyo (Japan)
Huifang Chen, Zhejiang University (P.R.China)
Christian Damsgaard Jensen, Technical University of Denmark (Denmark)
Teruo Higashino, Kyoto Tachibana University (Japan)
Tadanori Mizuno, Aichi Institute of Technology (Japan)
Jun Munemori, The Open University of Japan (Japan)
Yuko Murayama, Tsuda University (Japan)
Ken-ichi Okada, Keio University (Japan)
Norio Shiratori, Chuo University / Tohoku University (Japan)
Ian Wakeman, University of Sussex (UK)
Ismail Guvenc, North Carolina State University (USA)
Qing-An Zeng, North Carolina A&T State University (USA)
Tim Ziemer, University of Bremen (Germany)
Justin Zhan, University of Cincinnati Computer Science Faculty (USA)
Xuyun Zhang, Macquarie University (Australia)

Aims and Scope

The purpose of this journal is to provide an open forum to publish high quality research papers in the areas of informatics and related fields to promote the exchange of research ideas, experiences and results.

Informatics is the systematic study of Information and the application of research methods to study Information systems and services. It deals primarily with human aspects of information, such as its quality and value as a resource. Informatics also referred to as Information science, studies the structure, algorithms, behavior, and interactions of natural and a rtificial systems that store, process, access and communicate information. It also develops its own conceptual and theoretical foundations and utilizes foundations developed in other fields. The advent of computers, its ubiquity and ease to use has led to the study of informatics that has computational, cognitive and social aspects, including study of the social impact of information technologies.

The characteristic of informatics' context is amalgamation of technologies. For creating an informatics product, it is necessary to integrate many technologies, such as mathematics, linguistics, engineering and other emerging new fields.

Guest Editor's Message

Tetsuya Yokotani

Guest Editor of the Forty-fifth Issue of the International Journal of Informatics Society

We are delighted to have the Fortieth issue of the International Journal of Informatics Society (IJIS) published. This issue includes selected papers from the Sixteenth International Workshop on Informatics (IWIN2022), held online from August 31-September 3, 2022. The workshop was the sixteenth event for the Informatics Society. It was intended to bring together researchers and practitioners to share and exchange their experiences, discuss challenges and present original ideas in all aspects of informatics and computer networks. In the workshop, 25 papers were presented in eight technical sessions. The workshop was successfully finished, and precious experiences were provided to the participants. It highlighted the latest research results in informatics and its applications, including networking, mobile ubiquitous systems, data analytics, business systems, education systems, design methodology, intelligent systems, groupware, and social systems.

Each paper submitted to IWIN2022 was reviewed in terms of technical content, scientific rigor, novelty, originality, and presentation quality by at least two reviewers. Through those reviews, 18 papers were selected for publication candidates of the IJIS Journal, and they were further reviewed as Journal papers. We have three categories of IJIS papers, Regular papers, Industrial papers, and Invited papers, each of which was reviewed from different points of view. This volume includes papers among those accepted papers, which have been improved through the workshop discussion and the reviewers' comments.

We publish the journal in print as well as in an electronic form over the Internet. We hope that the issue would be of interest to many researchers as well as engineers and practitioners all over the world.

Tetsuya Yokotani received B.S., M.S., and Ph.D. in information science from the Tokyo University of Science in 1985, 1987, and 1997, respectively. He joined Mitsubishi Electric Corporation in 1987. Since then, he has researched high-speed data communication, optical access systems, home networks, and network performance evaluation technologies, mainly in the Information Technology R&D Center. In 2015, he moved to the Kanazawa Institute of Technology as a professor at the College of Engineering. Since then, he has engaged in research and education on networks for various IoT services and has proposed standardization in these related areas. Currently, he is an advisory board chair in the technical committee on Communication Quality and Reliability (CQR) in IEEE ComSoc. He is also a distinguished lecturer at IEICE. He is a fellow member of IEICE. He is also a member of IEEE ComSoc and IPSJ.

Regular Paper

Proposal of a Broadcaster Support Method using MR Stamp in 360-degree Internet Live Broadcasting

Yoshia Saito* and Kei Sato*

*Graduate School of Software and Information Science, Iwate Prefectural University, Japan
y-saito@iwate-pu.ac.jp

Abstract - In this study, we investigate the use of Mixed Reality (MR) stamp which supports broadcasters in reducing communication errors from viewers to a broadcaster in 360-degree internet live broadcasting. There is a problem that the broadcaster is unable to grasp the viewer's POV (Point Of View) compared to the conventional broadcasting method. We have confirmed that this problem could be reduced by combining an equirectangular video and 2D stamp which presents the viewer's interests in a simple image on the video. On the other hand, the 2D stamp system has three issues. The issues are (1) the stamp cannot be fixed on the target object, (2) the broadcaster must check a PC display to see the stamp, and (3) it is difficult to understand the position of the stamp in real space.
In this study, we propose MR stamp which can be displayed and fixed on the real space, which enables the broadcaster to check a holographic stamp on the real space through an MR device. To realize our proposal, we implemented the proposed system using Microsoft's HoloLens 2 which is a head mounted display (HMD) as the MR device. It can show holograms in real space by recognizing real space. We also evaluated the effectiveness of the MR stamp compared with the 2D stamp and found that the MR stamp with the spatial audio solved the three issues of the 2D stamp.

Keywords: 360-degree internet live broadcasting, Mixed Reality, MR stamp, Broadcaster support method

1 INTRODUCTION

YouTube started a 360-degree internet live broadcasting service from 2016 and it enables anyone to easily use the 360-degree internet live broadcasting service now. 360-degree internet live broadcasting is a service that combines internet live broadcasting with 360-degree videos using an omnidirectional camera. In 360-degree internet live broadcasting, a broadcaster can provide a 360-degree video to viewers in real-time without caring about the view angle of the camera. The viewers can change the POV according to their interests and communicate with the broadcaster using text chat.

The 360-degree internet broadcasting, however, has a lot of new problems. One of the problems is that the broadcaster cannot be aware of the viewers' POV. In the conventional internet live broadcasting, it uses a web camera which has a single lens and the single lens definitely shows the viewers' POV. The broadcaster can see what they are watching by the direction of the lens. On the other hand, 360-degree internet live broadcasting uses an omnidirectional camera that has a wide-angle lens or multiple lenses. It prevents the broadcaster from seeing what the viewers are watching by the direction of the lens.

There are many studies about the role of gaze information in remote communication [1][2]. In the studies, it concludes that the communicatee's gaze information indicates the target of interest or center of the topic. The gaze information in the remote communication is similar to the viewers' POV in the 360-degree internet live broadcasting. The broadcaster sometimes cannot understand the context of the viewers' comments and it causes communication errors between the broadcaster and the viewers.

To solve the problem of communication errors, we have proposed stamp functions in 360-degree internet live broadcasting [3]. In this study, the stamp functions help the broadcaster to understand what the viewers are talking about and find the object. However, it remains three issues in the proposed system, which are (1) the stamp cannot be fixed on the target object, (2) the broadcaster must check a PC display to see the stamp, and (3) it is difficult to understand a position of the stamp in the real space. These issues should be solved to realize smooth communication between the broadcaster and the viewers.

In this study, we propose MR stamp which is a new stamp function using MR that can be displayed and fixed in real space. It enables the broadcaster to check a holographic stamp on the real space through an MR device. The MR stamp can solve the issues of the previous stamp function.

The contributions of this paper are summarized as follows:

- We proposed a new broadcaster support method called MR stamp in 360-degree internet live broadcasting.
- We developed and evaluated a prototype system of the MR stamp using HoloLens 2.
- We clarified that the MR stamp enabled the broadcaster to find a target object in a relatively short time and spatial audio could help the broadcaster to find the MR stamp.

ISSN1883-4566 © 2023 - Informatics Society and the authors. All rights reserved.

The rest of this paper is organized as follows. Section 2 describes our previous work about stamp functions in 360-degree internet live broadcasting. Section 3 describes an overview of the proposed system and a use case of the MR stamp. Section 4 describes the implementation of the proposed system. Section 5 describes a first evaluation experiment to clarify the effects of the proposed system and reveal its problem. Section 6 describes a second evaluation experiment improving the proposed system. Section 7 summarizes this study.

2 PREVIOUS STUDY

We have proposed stamp functions in 360-degree internet live broadcasting to support communication between the broadcaster and the viewers. In this section, we explain the stamp functions in the previous study and its known issues.

2.1 Stamp Functions in 360-degree Internet Live Broadcasting

There are two stamps which are “Look” and “Go” stamps in the previous study as shown in Fig. 1. Figure 2 shows a user interface of the viewer in the previous study. The viewers can watch the 360-degree live video in spherical format and change the POV as they want. The viewers are also able to use the Look and Go stamp by selecting the kind of stamp and clicking on the video. The stamps are shown in the same place as the video on the user interface of the viewers and the broadcaster. Figure 3 shows the user interface of the broadcaster. The 360-degree video is displayed in equirectangular format. The broadcaster can check the stamps that are sent from the viewers without changing the POV.

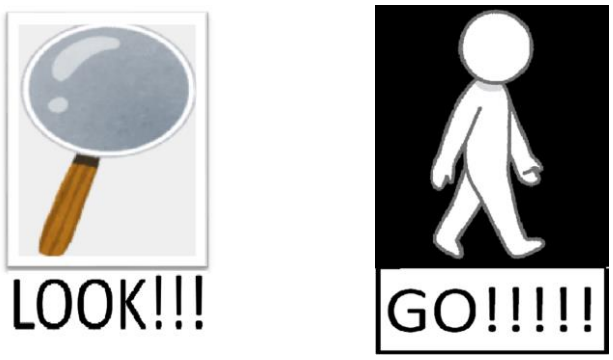
We evaluated the previous stamp function and found their advantages. The stamp function improved the easiness of communication between the broadcaster and the viewers compared with the case that the stamp was not used. The broadcaster could easily understand what the viewer talked about and found the object. Moreover, the stamp function increased the frequency of communication between the broadcaster and the viewers.

2.2 Known Issues

Although the previous study has advantages, there are three issues as follows.

(1) The stamp cannot be fixed on the target object.
A stamp only has information of a direction from the omnidirectional camera at a particular time. Therefore, the stamp cannot be fixed on the target object and it causes a positional shift of the stamp when the omnidirectional camera is moved.

(2) The broadcaster must check a PC display to see the stamp.



Look stamp Go stamp

Figure 1: Stamps in the previous study.

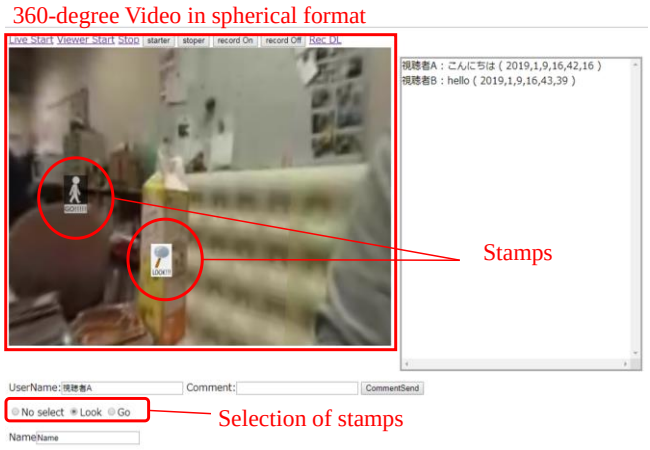


Figure 2: User Interface of viewer in the previous study.

Figure 3: User Interface of broadcaster in the previous study.

In the previous study, the broadcaster has to carry a laptop PC and check stamps from the viewers through the display. The check of the laptop PC display consumes time and prevents smooth communication between the broadcaster and the viewers. Besides that, it is dangerous to see the laptop PC while walking and it is inconvenient not to be able to use a hand which keeps the laptop PC.

(3) It is difficult to understand the position of the stamp in the real space.

The stamps are shown on the 360-degree video in equirectangular format. When a stamp appears on the video, the broadcaster must find the position of the stamp in real space. The distortion of the 360-degree video in equirectangular format makes it difficult to find the direction of the stamp.

3 PROPOSED SYSTEM

The MR stamp aims to solve the issues of the previous study using MR and realize smooth communication between the broadcaster and the viewers.

3.1 Effectiveness of MR

MR is a technology that displays holograms of virtual objects in real space and the users can interact with the holograms. Several researches show the effectiveness of MR in remote communication between users.

Lee [4] developed an MR remote collaboration system that shared 360-degree live video. In this system, a hologram of the remote user's hand is displayed in real space through the MR device. The hand gestures by the hologram help to understand each other's focus and improve their communication. Johnson [5] studied the effect of MR guidance. An experiment was conducted to understand how to provide explicit spatial information in a collaborative MR environment. The experiment result showed the MR guidance realized effective referencing through deixis. Other several researches show the effectiveness of the hologram for remote communication [6, 7]

From the related work, the reduction effect of communication errors can be also expected by introducing the MR technology for the stamp function in 360-degree internet live broadcasting.

3.2 System Model

Figure 4 shows a model of the proposed system. The proposed system is included in the existing 360-degree internet live broadcasting system. A broadcaster provides 360-degree live video to viewers using the 360-degree internet live broadcasting system. The viewers can send a 2D stamp which is implemented in the previous study to the broadcaster. The proposed system receives the 2D position information of the stamp and transforms it into 3D position information. The proposed system displays an MR stamp using the 3D position information and the broadcaster can check the MR stamp in real space through an MR device.

The proposed system can solve three issues in the previous section. The first issue which is that "The stamp cannot be fixed on the target object" can be solved by fixing the stamp on the real space using MR. The second issue which is that "The broadcaster must check a PC display to see the stamp" can be solved by using an MR device which is a type of HMD. The third issue which is that "It is difficult to understand the position of the stamp in the real space" can be solved by directly displaying the stamp on the target object in the real space using MR.

Table 1: Comparison between 2D stamp and MR stamp.

	Time to find	Errors to find	Easiness to find
2D stamp	Long time	Many errors	Difficult
MR stamp	Short time	Fewer errors	Easy

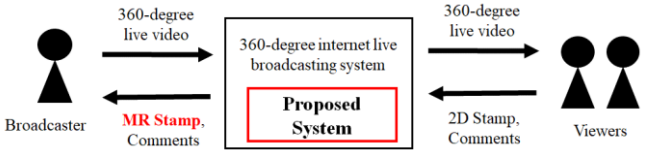


Figure 4: A model of the proposed system.

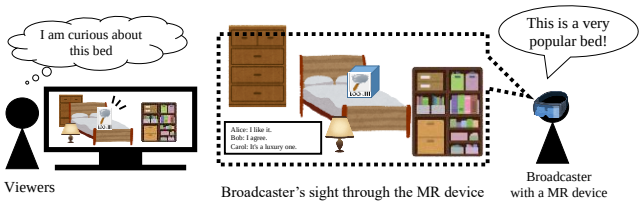


Figure 5: A use case of the proposed system.

Table 1 shows a comparison between the 2D stamp of the previous study [3] and the MR stamp. In terms of time to find the target object, the MR stamp would shorten the time because it does not need to check a PC display and understand the position of the stamp in the real space. In terms of the number of communication errors to find the target object, the MR stamp would also reduce the errors because it can be fixed on the target object. Moreover, the broadcaster would find the MR stamp easier to find the target object than 2D stamps because it has the advantage of less time and errors to find the target object besides that the MR device frees a hand which keeps the laptop PC.

3.3 Use Case

In this study, we suppose that an omnidirectional camera is fixed on an arbitrary position and 360-degree internet live broadcasting is performed indoors. The reason is that the usage environment should be simple for the first step of this study. We also suppose it is used at the showroom and the exhibition.

Figure 5 shows a use case of the proposed system. The viewers can send an MR stamp to the broadcaster if they want to see a particular object in a showroom. The broadcaster can understand the request from the viewers easier than when only comments are used. Since 2020, online virtual events have been increasing because of COVID-19. In online virtual events, it is said that more opportunity for real two-way communication between the broadcaster and viewers [8]. In online conferences, it is reported that they cannot have smooth relationships with each other between participants [9]. The MR stamp would realize smooth two-way communication between the broadcaster and the viewers in various online virtual events.

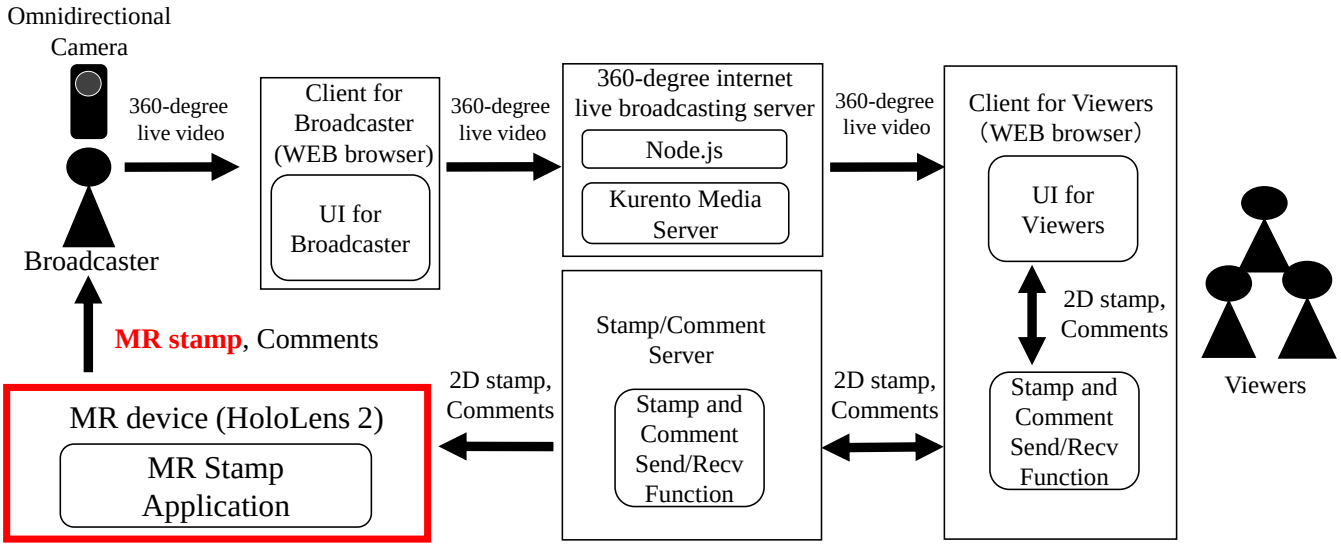


Figure 6: Architecture of the prototype system.

4 IMPLEMENTATION

We implemented a prototype system of the MR stamp using Hololens2. In this section, we describe the architecture of the prototype system and its main application.

4.1 System Architecture

The prototype of the proposed system is based on the 360-degree internet live broadcasting system of the previous study. Figure 6 shows the architecture of the prototype system. The red square shows new implementation in this study and the other parts are diverted from the previous study. A broadcaster can start 360-degree internet live broadcasting using the client for broadcaster on a web browser. The 360-degree internet live broadcasting server distributes it to viewers. The viewers can watch the 360-degree live video and send 2D stamp and comments in the same manner as the previous study. The stamp/comment server forwards it to all clients for viewers and an MR device of the broadcaster. We use Microsoft HoloLens 2 as the MR device. In the HoloLens2, the MR stamp application is running. The application presents the MR stamp and comments to the broadcaster. The broadcaster can check the MR stamp in real space through the HoloLens 2.

4.2 MR Stamp Application

The MR stamp application receives 2D position information of the stamp and needs to transform it to 3D position information for the MR stamp. To realize the coordinate transformation, we use a Raycast function in Unity, which irradiates a 3D ray from an origin point to a target direction and detects intersecting collisions. The origin point is the coordinates of the omnidirectional camera. The target direction can be given by the coordinates of the stamp. The HoloLens 2 has a function of spatial mapping. The spatial mapping provides a detailed representation of real space surfaces in the environment around HoloLens 2. The raycast detects the intersecting collisions with the real space surfaces and returns the 3D coordinates. The MR stamp is displayed on the 3D coordinates.

Figure 7 shows an example of the MR stamp. The MR stamp is shown as a 3D square frame so that it is easy to check the target object from the broadcaster. The MR stamp disappears after a period of time (10 seconds in this implementation).

The comments from the viewers are displayed on a comment window as shown in Fig. 8. The comment window tracks the broadcaster's sight.

Figure 7: An example of the MR stamp.

Figure 8: Comment window through HoloLens 2.

5 FIRST EVALUATION

We conducted an experiment to evaluate the effectiveness of the MR stamp using the implemented prototype system. The purpose of the evaluation is to check whether the three issues of the previous study can be solved or not by using the MR stamp.

5.1 Environment and Procedure

We compare the prototype system (MR stamp system) with the stamp system of the previous study (2D stamp system). In the experiment, A broadcaster performs 360-degree internet live broadcasting and two viewers watch the broadcasting. In the room of the broadcaster, there are various objects. The viewers talk about an object in the broadcaster's room using the MR/2D stamp and comments. The broadcaster looks for the target object and communicates with the viewers. The evaluation items are as follows; (1) time to find the target object, (2) number of communication errors, and (3) subjective easiness to find the target object.

The experiments were conducted 4 times. There were one broadcaster and two viewers per time and the participants were students at our university. The broadcasting time was 30 minutes. Figure 9 shows the procedure of the experiment. At first, a viewer sends a stamp to the broadcaster. The target object and the viewer who performs the task are predetermined by the task instruction. The broadcaster looks for the target object referring to the stamp. After the target object is found, the broadcaster confirms whether the object is correct or not by speaking to the viewer. The viewer replies whether it is correct or not. If it is not correct, the broadcaster continues to find the target object. The search task with stamp is performed 2 times. At last, we ask the broadcaster how easy to find the target object in 5-point scale with a questionnaire. This procedure was performed for both the MR stamp system and the 2D stamp system. The order of the MR stamp and 2D stamp system was random to keep fairness.

Figure 10 shows the target objects used in the experiment. Target A and C are comprehensible ones to find because there is nothing around the object (hereafter, single object). Target B and D are mistakable ones to find because there are several similar objects around the object (hereafter, multiple objects). In addition to these target objects, there are several dummy objects in the broadcaster's room. We locate the target and dummy objects scattered in all directions to make the broadcaster look around to find the object. Figure 11 shows the location of all objects in the broadcaster's room.

5.2 First Evaluation Results

Figures 12 and 13 show the evaluation results of the first evaluation. There are the results of 4 broadcasters with 2D and MR stamp systems. In terms of Target, "Single" means the single object such as Target A and C, and "Multiple" means multiple objects such as Target B and D. "Time to find" denotes the time from when a stamp was displayed to when the broadcaster found the correct target object. "Errors" denotes the number of times when the broadcaster indicated incorrect objects.

The average time to find the target object in the 2D stamp system is 11.25 seconds in the case of the single object and 20.25 seconds in case of the multiple objects. The total of the average time in the 2D stamp system is 15.75 seconds. In the MR stamp system, it takes 18 seconds in the case of the single object and 13.25 seconds in case of the multiple objects. The total of the average time in the MR stamp

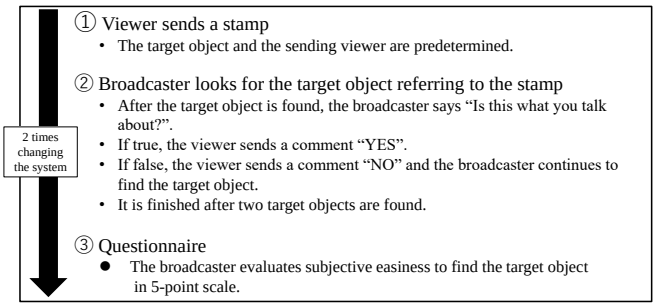


Figure 9: Procedure of the experiment.

Figure 10: Target objects.

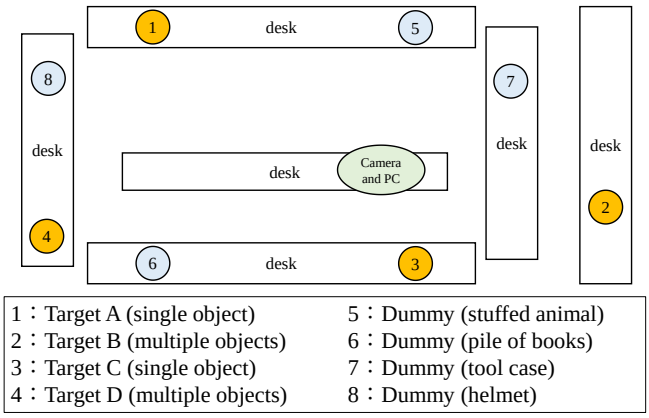


Figure 11: Location of all objects in the broadcaster's room.

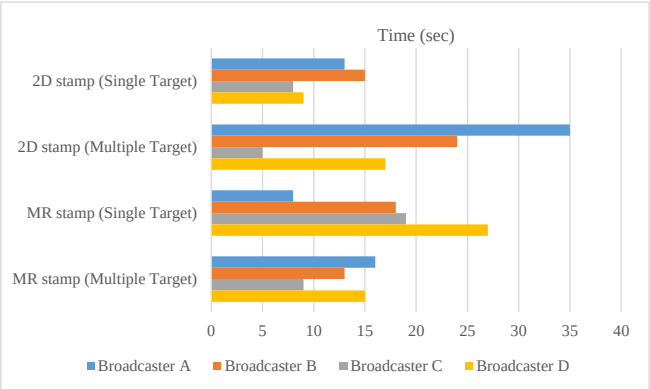


Figure 12: Evaluation result of time to find the target object in the first evaluation.

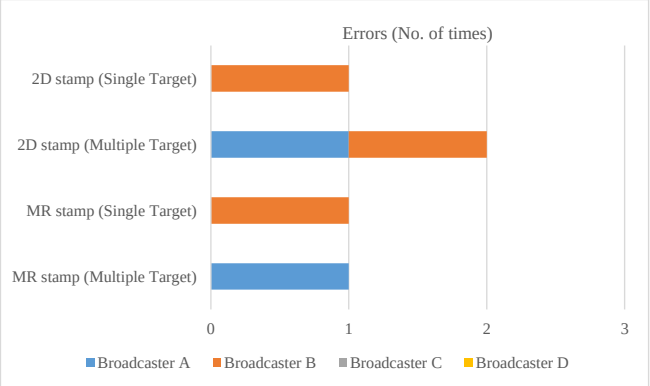


Figure 13: Evaluation result of the number of errors to find the target object in the first evaluation.

system is 15.625 seconds. The average number of errors in the 2D stamp system shows 0.25 and 0.5 in the single and multiple objects respectively. The total of the average number of errors in the 2D stamp system is 0.375. The average number of errors in the MR stamp system shows 0.25 and 0.25 in the single and multiple objects respectively. The total of the average number of errors in the MR stamp system is 0.375. We conducted a Student's t-test for the results of the total average and there is no significant difference in the average time to find ($p = 0.98$) and the average number of errors ($p = 0.63$) between the 2D and MR stamp systems.

Figure 14 shows the result of the questionnaire which is about the subjective easiness to find the target object in 5-point scale. The average points of the 2D stamp system are 4.5 and 3.25 in the single and multiple objects respectively. The total average in the 2D stamp system is 3.85. The average points of the MR stamp system are 4.25 and 3.75 in the single and multiple objects respectively. The total average in the MR stamp system is 4. We conducted a Student's t-test for the results of the total average and there is no significant difference in the subjective easiness to find the target object ($p = 0.83$).

From these results, we didn't find the effectiveness of the MR stamp system. In the free descriptive answer of the questionnaire, there were several same answers that "It was difficult to find the MR stamp". Even if the MR stamp can help the broadcaster find the target object, the broadcaster

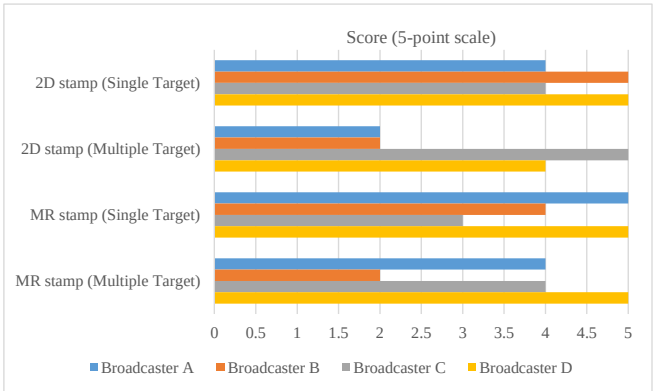


Figure 14: Questionnaire result of subjective easiness to find the target object in the first evaluation.

cannot find the location of the MR stamp itself in the experiment. We expect that an additional function to find the location of the MR stamp will improve the effectiveness of the MR stamp system.

6 SECOND EVALUATION WITH SYSTEM IMPROVEMENT

In the first evaluation, there was no difference in the effectiveness between the 2D stamp system and the MR stamp system. The viewing angle of HoloLens2 is approximately 52 degrees diagonal (28.5 degrees vertical, 43 degrees horizontal), which is narrower than the human viewing angle. Because it can only see MR stamps within a limited range. it is difficult to find the location of the MR stamps within a certain time. Since the prototype system had a problem in that it was difficult to find the location of the MR stamp itself, we tried to improve the prototype system to find the MR stamp easily and conduct a second evaluation using the improved prototype system.

6.1 System Improvement

To help the broadcaster find the MR stamp, we introduce spatial audio into the MR stamp. The spatial audio enables the broadcaster to perceive sound all around and be aware of the location of the sound source. Titus [10] studies the effectiveness of spatial audio in finding the location of a maker. He finds that the spatial audio helps the user with an HMD to find a rough location of the maker although it is not enough to specify the exact location of the maker. In our MR stamp system, the broadcaster would find the MR stamp if he/she can be aware of a rough location of the stamp.

We implemented the spatial audio to the MR stamp using a spatial audio function of MRTK (Mixed Reality Toolkit). When the MR stamp is displayed, the spatial sound is generated from the location of the MR stamp. The broadcaster can hear the spatial sound through the HoloLens 2 and find the MR stamp quickly.

We also conducted a preliminary experiment to check how accurately the user could grasp the direction of the spatial sound using HoloLens 2. From the experiment, the

user could grasp the horizontal direction of the spatial sound approximately. Although the accuracy was low and it was not sufficient to grasp the accurate direction, it could support the MR stamp. On the other hand, it was hard to grasp the vertical direction of the spatial sound in this implementation. Since the target objects were placed horizontally in the first evaluation, the problem that the vertical direction could not be grasped would not be affected.

6.2 Second Evaluation Results

We conducted a second evaluation using the improved prototype system with spatial audio. The environment and the procedure are the same as the first evaluation.

Figures 15 and 16 show the evaluation results of the second evaluation. The average time to find the target object in the 2D stamp system is 17.5 seconds in the case of the single object and 19.75 seconds in case of the multiple objects. The total of the average time in the 2D stamp system is 18.625 seconds. In the MR stamp system, it takes 5 seconds in the case of the single object and 5 seconds in case of the multiple objects. The total of the average time in the MR stamp system is about 5 seconds. The average number of errors in the 2D stamp system shows 0.25 and 0.5 in the single and multiple objects respectively. The total of the average number of errors in the 2D stamp system is 0.375. The average number of errors in the MR stamp system shows 0 and 0 in the single and multiple objects respectively. The total of the average number of errors in the MR stamp system is 0. We conducted a Student's t-test for the results of the total average. There is a significant difference in the average time to find ($p = 0.01$) and a significant trend in the average number of errors ($p = 0.08$) between the 2D and MR stamp systems. Although the errors in the 2D stamp would be caused by the first issue "The stamp cannot be fixed on the target object", no errors occurred in the MR stamps.

Figure 17 shows the result of the questionnaire which is about the subjective easiness to find the target object in 5-point scale. The average points of the 2D stamp system are 3 and 2.25 in the single and multiple objects respectively. The total average in the 2D stamp system is 2.625. The average points of the MR stamp system are 4.75 and 4.5 in the single and multiple objects respectively. The total average in the MR stamp system is 4.625. We conducted a Student's t-test for the results of the total average and there is a significant difference in the subjective easiness to find the target object ($p = 0.0007$).

From these results, we can find the effectiveness of the MR stamp system with special audio. In terms of the three issues of the previous study, the first issue which is that "The stamp cannot be fixed on the target object" is solved because the number of errors decreases in the proposed system. The second issue which is that "The broadcaster must check a PC display to see the stamp" is solved because the time to find the target object decreases. The third issue which is that "It is difficult to understand the position of the stamp in the real space" is solved because both the number of errors and the time to find the target object decreases.

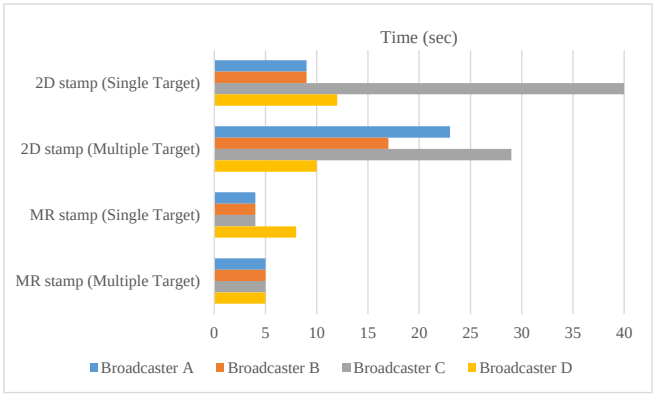


Figure 15: Evaluation result of time to find the target object in the second evaluation.

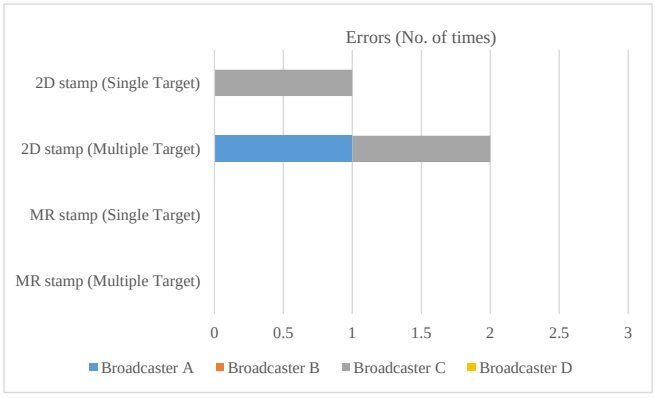


Figure 16: Evaluation result of the number of errors to find the target object in the second evaluation.

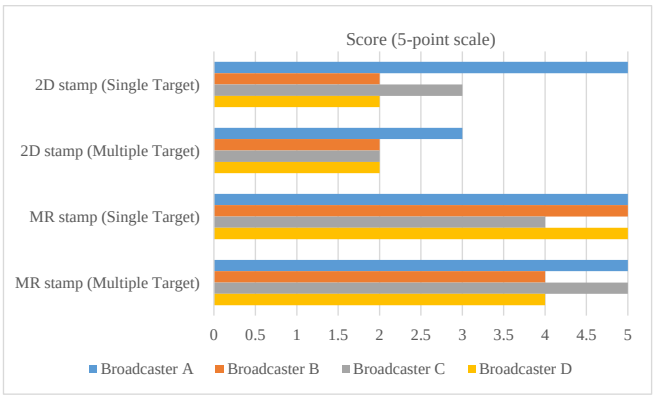


Figure 17: Questionnaire result of subjective easiness to find the target object in the second evaluation.

7 CONCLUSION

In this study, we proposed MR stamp which can be displayed and fixed on the real space, which enables the broadcaster to check a holographic stamp on the real space through an MR device. We implemented a prototype system of the MR stamp with special audio. In the evaluation of this study, we verified the effectiveness of the MR stamp compared with the 2D stamp using the prototype system.

We found that the MR stamp with the spatial audio made it easier for the broadcaster to intuitively grasp the location of the stamp. In addition, a comparison with the 2D stamp system revealed that there was a significant difference in the time to find the target object and the subjective easiness to find the target object. There was also a significant trend in the average number of errors between the 2D and MR stamp systems. From the result, we found that the MR stamp could solve the issues of the 2D stamp. In the future, it is necessary to improve the accuracy of the stamp display.

REFERENCES

[1] R. Vertegaal, “The GAZE groupware system: mediating joint attention in multiparty communication and collaboration”, Proc. of SIGCHI conference on Human Factors in Computing Systems, pp.294-301 (1999).
[2] D. M. Grayson and A. F. Monk, “Are you looking at me? Eye contact and desktop video conferencing”, ACM Transactions on Computer-Human Interaction (TOCHI), Vol. 10, Issue 3, pp.221-243 (2003).
[3] Y. Saito, A. Kuzumaki, Y. Yahata and D. Nishioka, “Development of a Communication Support System with Stamp Functions in 360-degree Internet Live Broadcasting”, Proc. of DICOMO2019, pp.895-900 (2019) (in Japanese).
[4] G. A. Lee, T. Teo, S. Kim and M. Billinghurst, “Mixed Reality Collaboration through Sharing a Live Panorama”, Proc. of SA’17, pp.1-4 (2017).
[5] J. G. Johnson, D. Gasques, T. Sharkey, E. Schmitz and N. Weibel, “Do You Really Need to Know Where “That” Is? Enhancing Support for Referencing in Collaborative Mixed Reality Environments”, Proc. of SIGCHI conference on Human Factors in Computing Systems, pp.1-14 (2021).
[6] O. Oda, C. Elvezio, M. Sukan, S. Feiner and B. Tversky, “Virtual Replicas for Remote Assistance in Virtual and Augmented Reality”, Proc. of the 28th Annual ACM Symposium on User Interface Software & Technology, pp.405-415 (2015).
[7] T. Piumsomboon, A. Day, B. Ens, Y. Lee, G. Lee and M. Billinghurst, “Exploring enhancements for remote mixed reality collaboration”, SIGGRAPH Asia 2017 Mobile Graphics & Interactive Applications, pp.1-5 (2017).
[8] StateOfVirtualEvents2021, available from <https://s3.amazonaws.com/media.mediapost.com/uploads/StateOfVirtualEvents2021.pdf> (accessed 2024-1-19).
[9] C. Misa, D. Guse, O. Hohlfeld, R. Durairajan, A. Sperotto, A. Dainotti and R. Rejaie, “Lessons Learned Organizing the PAM 2020 Virtual Conference”, ACM SIGCOMM Computer Communication Review, Vol.50, Issue 3, pp.46-54 (2020).
[10] T. J. J. Tang and W. H. Li, “An Assistive EyeWear Prototype that interactively converts 3D Object Locations into Spatial Audio”, Proc. of ISWC’14, pp.119-126 (2014).

(Received: October 29, 2023)
(Accepted: January 29, 2024)

Yoshia Saito received his Ph.D. degree from Shizuoka University, Japan, in 2006. He had been an expert researcher at the National Institute of Information and Communications Technology (NICT) from 2004 to 2007, in Yokosuka, Japan. He was a lecturer from 2007 to 2011 at Iwate Prefectural University and he is currently an associate professor at the University. His research interests include computer networks and internet broadcasting.

Kei Sato received his master's degree from Iwate Prefectural University, Japan, in 2022. His research interests include 360-degree internet live broadcasting.

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

ISSN1883-4566 © 2023 - Informatics Society and the authors. All rights reserved.

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.

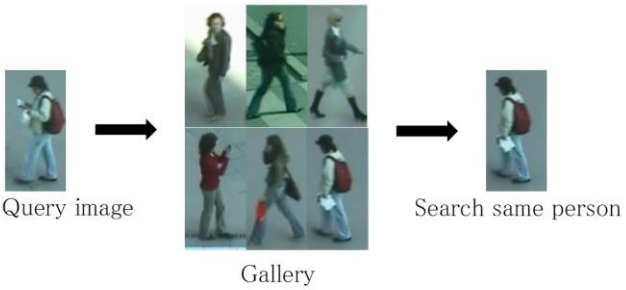


Figure 1: Overview of person re-identification

· 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.

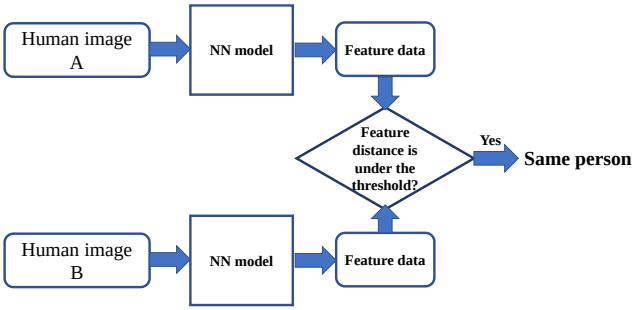


Figure 2: An image of a person re-identification process

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

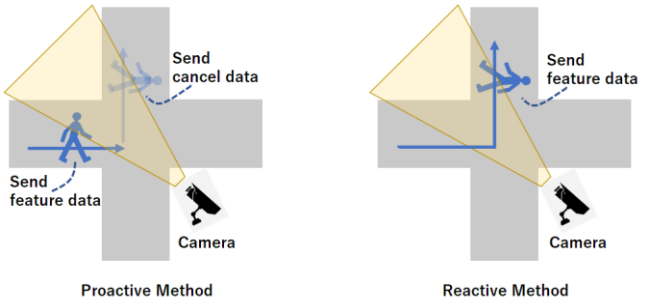


Figure 3: Time to start communication under our proposed method.

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).

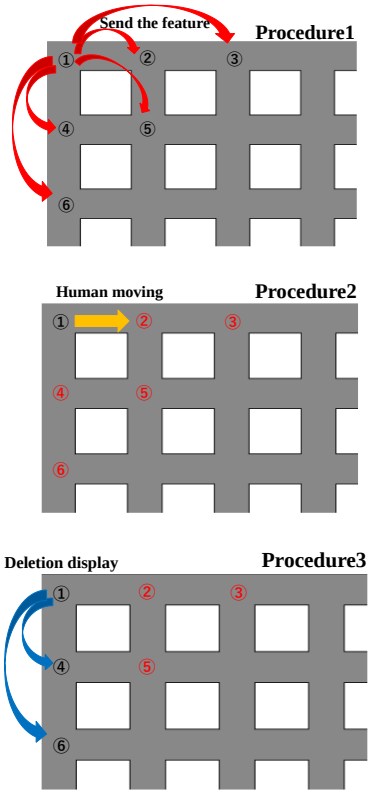


Figure 4: The flow of the proactive method

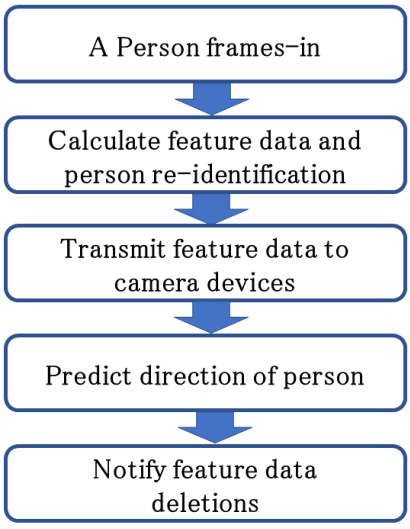


Figure 5: Processes for each camera device in the proactive method

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

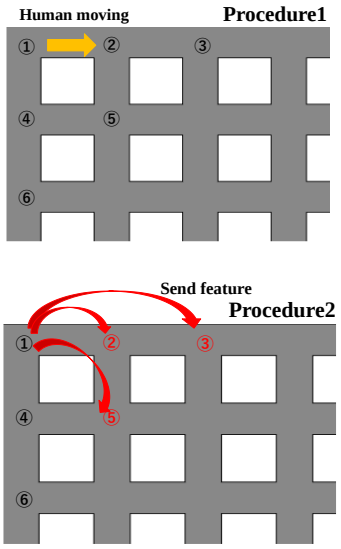


Figure 6: The flow of the reactive method

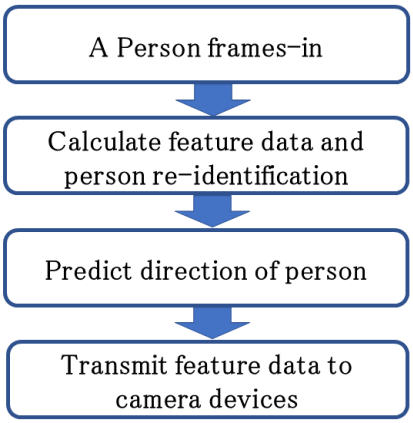


Figure 7: Processes for each camera device in the reactive method

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.

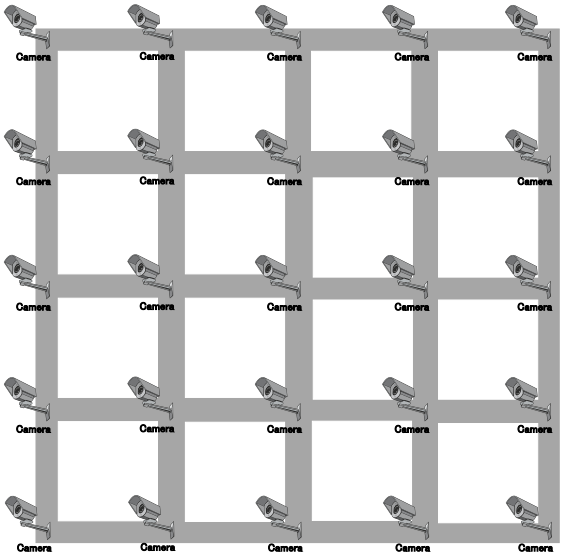


Figure 8: A map for the simulation
(25 camera devices)

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.

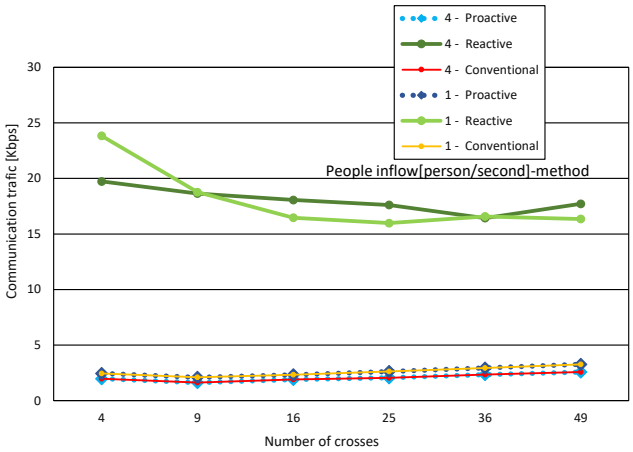


Figure 9: Communication traffic changing the number of crosses (number of camera devices) and the people inflow

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.

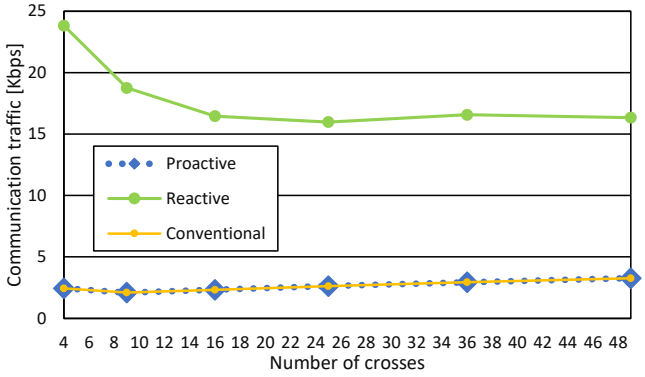


Figure 10: Communication traffic changing the number of crosses (number of camera devices)

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

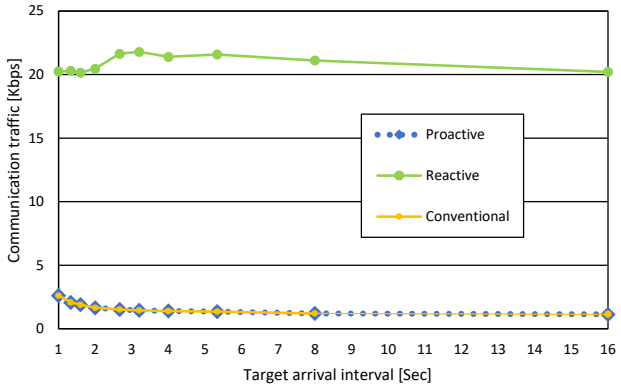


Figure11: Communication traffic of changing the target arrival interval

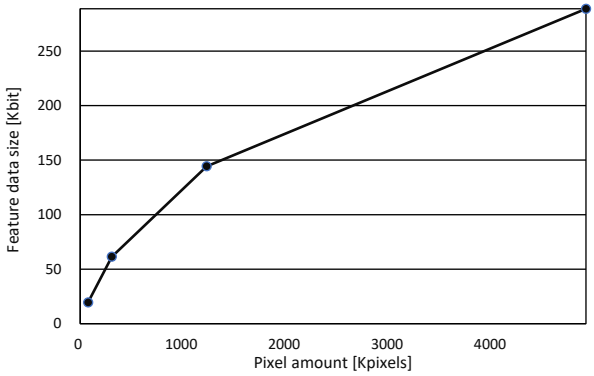


Figure12: Number of pixels and features

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

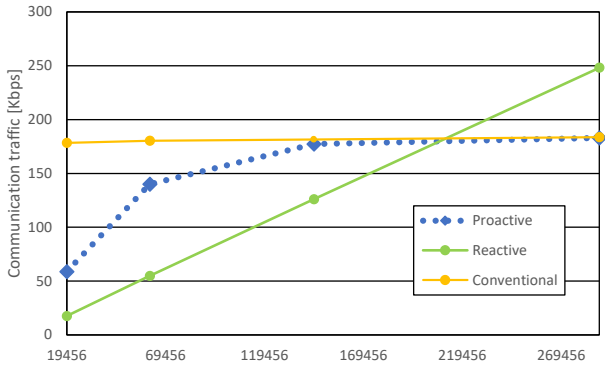


Figure13: Communication traffic for features by resolution

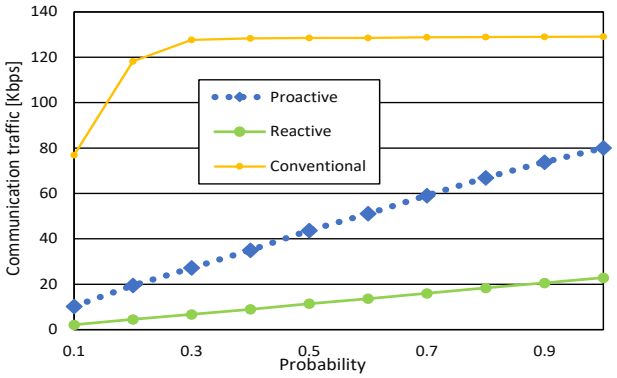


Figure 14: Comparison of communication traffic

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).

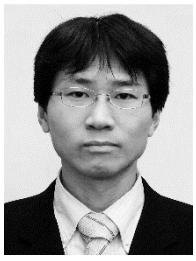
[2] S. Liao and L. Shao, “Graph sampling based deep metric learning for generalizable person re-identification”, arXiv:2104.01546, 11pages (2021).

[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.

Regular Paper

A Study on Effectiveness of SNS Data in Flood Estimation

Kei Hiroi†, Akihito Kohiga‡, and Yoichi Shinoda‡

†Disaster Prevention Research Institute, Kyoto University, Japan
‡Japan Advanced Institute of Science and Technology, Japan
hiroi@dimsis.dpri.kyoto-u.ac.jp

Abstract - Reports of damage posted to social networking services (SNSs) by residents of disaster-stricken areas at the time of a disaster are expected to be of great use. They may be a valuable source of information in areas where it is difficult to install, operate, and maintain observation devices or where devices are missing. However, their effective use for damage assessment has not yet been determined. Therefore, a study on the complementary use of SNS data for flood analysis using data assimilation to improve damage assessment is urgently needed. In this paper, we report the evaluation results of data assimilation assuming that SNS data can be collected stably, and we discuss how useful SNS data are for flood damage assessments.

Keywords: flood estimation, state-space model, temporal-spatial analysis, data assimilation

1 INTRODUCTION

There are concerns that the risk of floods will intensify on a global scale. The Fifth Assessment Report of the Intergovernmental Panel on Climate Change (IPCC) stated that global warming is gradually progressing, and it is likely that the frequency and intensity of rainfall will change accordingly [1]. There are many areas worldwide where the frequency and intensity of heavy rain and flooding are increasing [2], [3]. Among the measures against flood damage, observing rainfall, rivers, and flooding and understanding the changing situations of rainfall and rivers as well as their influences enable people to determine what actions should be taken and to take effective steps to prevent or mitigate damage. Many previous studies have attempted to estimate flood risks using area vulnerability. For example, in [4], the flood risk in the city was estimated with a detailed spatial resolution of approximately 2 meters. [5], [6] conducted research to estimate index-based flood risk using a theoretical hydraulic engineering model. Furthermore, a Chinese case in [7] examined recognition of risks during the 1997 Red River flood situation. Studies are actively conducted to correctly analyze risks by presenting risks to people in affected areas and raising awareness of individual flood risks, which can lead to mitigation behavior [8], [9].

However, these previous studies are not temporal estimation methods; rather, they are static estimation approaches used to calculate maximum water level. A static estimation result is a risk estimation in which the risk value might change due to rainfall fluctuations. Considering evacuation behavior, dynamic risk estimation is required because flood situations change very rapidly with flooding phenomena over streets due to water overflowing from small rivers and waterways spreading throughout the city in a complicated manner and due to rainwater that cannot be completely drained. Therefore, it is necessary to calculate the high temporal-spatial flood level, which fluctuates according to the rainfall situation, to understand risk with a high temporal resolution for guiding evacuation behavior. Our research goal is to detect flooding as time-series data with only a limited number of observation devices.

This paper investigates whether SNS data can be used to assess flooding. SNS data are effective for determining flood levels even in places where it is difficult to install, operate, and manage observation devices. Although there have been many studies on flood damage detection using SNSs, their effectiveness has not been clarified, and the amount and content of data collected are not fixed depending on the flood damage case. This paper validates the SNS data using the following procedure based on our state-space model (SSM), which was used in our previous research. The purpose of SNS data validation is to investigate whether SNS data can contribute to the accuracy of flood assessment and under what conditions SNS data can improve accuracy.

Then, we also suggest a system for an appropriate SNS use case that could further improve the accuracy of flooding assessment by adding SNS data as well as observation data and calculating flooding conditions for the entire affected area. To realize this system, it is necessary to validate the effectiveness of the SNS data.

We generate quantified SNS data from flood simulations in this paper. To generate the SNS data, we use time-series data collected from observation devices at multiple locations. The results of flood analysis simulation were assimilated using the time-series data to simulate the SNS locations, the timing of postings, and numerical flood levels. Then, we determine the errors in the simulated SNS data. Afterward, we regenerate a flood level at the observation location based on the simulated SNS data with errors and examine the error accuracy against the data assimilation accuracy.

2 RELATED WORKS

2.1 Flood Monitoring

Traditional river sensors [10], [11] have succeeded in detecting disaster signs in large-scale rivers, which have the ad-

ISSN1883-4566 © 2023 - Informatics Society and the authors. All rights reserved.

vantage of stable monitoring and the disadvantage of installation limitation (i.e., very large equipment, high installation cost of several million dollars, and complicated preconfiguration). Improvements in installation limitations enable the possibility of a large number of sensor installations and reliable detection to improve monitoring sensors with higher resolution. Flood prediction, hydrological techniques [12] or artificial neural networks [13], [14] are proposed as high prediction methods. Predicting rising river levels has resulted in highly precise river inflow in the view of large-scale river analyses. However, these previous methods cannot predict the flooding of smaller rivers and waterways. This is because complex water flow prediction requires analyzing complicated relationships among a plurality of confluent rivers and factoring in the impact of rainfall dynamics.

2.2 Risk Estimation with Higher Spatial Resolution

Various studies have already attempted to generate information about places that are dangerous. In case studies, such as [15], [16], their research aimed to present risks on maps. Sinnakaudan et al. [15] developed an ArcView GIS extension as an efficient and interactive spatial decision support tool for flood risk analysis. Their extension is capable of analyzing computed water surface profiles and producing a related flood map for the Pari River in ArcView GIS. In another GIS-based flood risk assessment, Lyu et al. [16] studied the Guangzhou metro system's vulnerability. Their results showed the vulnerability of several metro stations using the flood event that occurred in Guangzhou on May 10, 2016.

Some studies have proposed modeling methods that collect data strictly as input data [4], [17], [18]. Ernst et al. [4] presented a microscale flood risk analysis procedure as a 2-meter grid, relying on detailed 2D inundation modeling and on a high-resolution topographic and land-use database. However, detailed risk estimation requires detailed data measurements, such as laser altimeter data, and it is not realistic to measure these data in all areas.

2.3 Flood Detection through Social Networking Services

Another way to learn about flooded areas is through social networking services (SNSs). Kim et al. [19] stated that social networking is the fourth most popular information source for accessing emergency information. Then they applied social network analysis to convert emergency social network data into knowledge for the 2016 flood in Louisiana. Their objective was to support emergency agencies in developing their social media operation strategies for a disaster mitigation plan. This study explored patterns of interaction between online users and disaster responses.

Sufi et al. [20] designed a disaster monitoring system on social media feeds related to disasters through AI- and NLP-based sentiment analysis. Their system has a mean accuracy of 0.05. They report that their system shows potential disaster locations with an average accuracy of 0.93. Teodorescu [21] designed a method to analyze SNSs for forecasting and relief and mitigation measures. His method analyzes SNS-related time series with the aim of establishing correlations between the disaster characteristics and the SNS response. Although studies using SNS have been applied in many flood damage cases, SNS data are not always posted as expected, and the accuracy may not be achieved as reported in these studies.

2.4 Issues and Approaches

To find safe evacuation routes, it is important to determine the situation regarding the roads in urban areas. Currently, flood damage assessment is based on two methods: numerical simulation (e.g., flood analysis) and monitoring using low-resolution ground observation data (precipitation and river water levels).

Numerical simulations are based on differential equations for flood flow in urban areas for a given amount of precipitation, and the maximum flood level in a detailed given area (e.g., a 10-m grid) is calculated. Based on the calculated results, areas that are anticipated to be hazardous during heavy rainfall are published. However, the analysis uses an ideal model that assumes fixed parameters, such as the amount of precipitation, its runoff coefficient, and the outflow conditions of drainage channels. Therefore, in urban areas with complex rainfall distributions and land uses, the analytical results and actual flood levels will differ. As a result, flooding of roads occurs prior to the announcement of warnings and evacuation information, leading to damage.

On the other hand, monitoring establishes thresholds for dangerous water levels at specific locations where there is concern about road underpasses and river breaches. This method involves situation monitoring to detect the occurrence of flooding based on observation data. This method easily assesses the actual damage but has limited observation points.

SNS data are expected to solve these monitoring limitations. As indicated in the previous section, the importance of SNSs in flood damage detection has long been known and has been applied in many flood damage cases. However, there is a fundamental problem with water damage detection using SNS. That is, SNS data are not necessarily posted in every case. While it may work effectively in floods with a high number of postings, it is highly likely that it will not be as accurate as reported in floods with a low number of postings. In particular, it may be difficult to post while ensuring safety in heavily damaged areas, and communication problems may prevent posting. We are convinced that these problems are obstacles to the effective use of SNSs for flood damage detection. Therefore, in this paper, we investigate how much SNS data regarding the number of postings, their contents, and the timing of postings would be effective for damage assessment.

We intend to develop a system for improving the accuracy of flood level estimation through data assimilation using heterogeneous data for investigation in this paper. We have previously proposed a method for estimating the expansion process of flooding by applying data assimilation using heterogeneous observation time-series data to simulate flood analyses. Our estimation method showed a significant improvement, with an error of less than 9 cm. We are planning to add SNS data to this estimation method to further improve its accuracy and to

develop a system to present the flood disaster situation in the entire affected area. However, although there have been many studies on flood damage detection using SNSs, the amount and content of the data collected are not well defined for each flood damage case. The effectiveness of using SNS for flood level estimation has not yet been clarified. Therefore, with the aim of developing a system to accurately estimate and present flooding situations for entire affected areas, this paper investigates the relationship between SNS data and the accuracy of flooding estimation through simulations.

3 PROPOSAL OF A DATA ASSIMILATION METHOD FOR IMPROVING THE ACCURACY OF FLOOD ESTIMATION

3.1 Overview

This research proposes a system for estimating floods through data assimilation to identify safe evacuation routes and timing in the event of an urban flood. We propose a flood estimation system based on data assimilation. This system aims to improve the accuracy of flood estimation by assimilating data using observation time-series data and SNS data. We intend to further improve accuracy by encouraging system users to submit messages that compensate for the lack of observation locations. The system will leverage the advantages of both observations, which can provide accurate time-series data, and SNS data, which can easily provide a large amount of data based on the number of posted messages.

The procedure in our system consists of three phases(Fig. 1). In Phase 1: SNS Data Validation, where we must validate the effectiveness of SNS data for improving the accuracy of flooding assessments. Although some previous studies have shown that SNS data are effective for disaster damage assessment, they are empirical validations based on data collected during that particular disaster, and we cannot eliminate the possibility that the highly effective data were simply collected coincidentally. There have been few verifications in terms of what kind of SNS data are effective or ineffective among the SNS data posted during disasters. Consequently, we need a scientific validation of the relationship between SNS data and analysis.

Phase 2: Flood Assessment Promotion Requirement Validation, which is a more detailed survey compared with the validation conducted in Phase 1. We investigate the flooding conditions of various past flood events and the social networking data posted at the time of the event. Simultaneously, we simulate SNS data based on the relationship between the Past SNS data and the physical measurement of events from learning process. Furthermore, SNS Generator simulates numerical data of flooding under the scenario of disaster occurrence and generates SNS data based on the measured data. Through phase 2, the effectiveness of SNS data in flood assessment will be validated for as many flood disasters as possible. Moreover, the system creates dissemination requirements to determine what kind of SNS data would improve the accuracy of the assessment.

Phase 3: Flood assessment, as shown on the right side of Fig. 1. In Phase 2, we will store the SNS data knowledge that

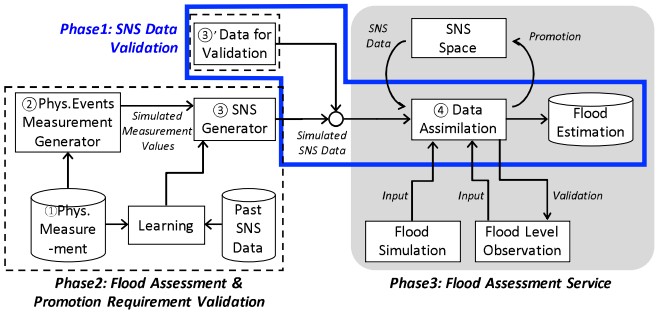


Figure 1: Proposed System for Flood Assessment Approach

is useful for flood assessment. Furthermore, our system will use this knowledge to disseminate the desired data to the SNS space. Phase 3 involves performing a highly accurate flood assessment using the collected data. This assessment would involve an estimation of flood level using data assimilation. In data assimilation, the estimated water level analyzed by the flood simulation is corrected with observed data and SNS data using a state-space model. The results of data assimilation are compared with the observed data and SNS data. As a result of the comparison, the SNS data are obtained through commercial services and other means in areas where flooding has been detected with low accuracy. The results can also be used to improve the data assimilation process. We plan to develop a dissemination function that encourages SNS space to post SNS data in areas with insufficient data. By providing SNS data from SNS space, the data assimilation results can be updated to improve accuracy.

3.2 Flood Assessment Approach

Our proposed simulation is based on the use of observations and SNS data. Previous studies [22] have shown that observation data contribute significantly to the accuracy of flood water level correction, but there is an upper limit to the number of observation devices because they need to be installed. Therefore, the effective use of SNS data when devising a system is considered to be critical to improving accuracy. As mentioned in 2.4, there are many studies on flood damage detection using SNS data, most of which are analyses of flood damage, where a large amount of data is available. However, whether or not we can collect a large amount of SNS data depends on the characteristics of the flooded area. Thus, it is not always possible to collect SNS data that can be used for flood analysis in all flood disasters. In addition, SNS data collected at the time of actual flooding also vary in the expression of information and the timing of postings. Furthermore, the accuracy of many of the data cannot be verified. Therefore, to sustainably utilize SNS data for disaster management, it is necessary to verify its effectiveness in various cases.

We investigate the conditions under which SNS can contribute to flood estimation and how SNS data can improve the accuracy of flood analysis. Phase 2 in Fig. 1 shows our expected use of SNSs. The SNS data collected from past floods are compared with actual flood damage (①physical measurement), and the results are used by a ②physical events measurement generator to simulate the damage that occurred at

the time of the flood as simulated measurement values. Based on simulated measurement values, ③an SNS generator generates simulated SNS data. Using simulated SNS data collected in this phase, we perform ④data assimilation that combines flood analysis simulations and observed data, and calculate under what conditions the SNS data should be collected to improve flood estimation accuracy. The process of generating simulated SNS data here involves data analysis in a cyber-physical system. By feeding the analytical results back to the data assimilation process, that is, the physical space, we expect to improve the accuracy of data assimilation and disseminate contributions by the SNS Promotor. The purpose of this research is to verify the accuracy of flood estimation using simulated SNS data and to determine the accuracy, precision, and timing of postings on SNSs that can be used in times of disaster.

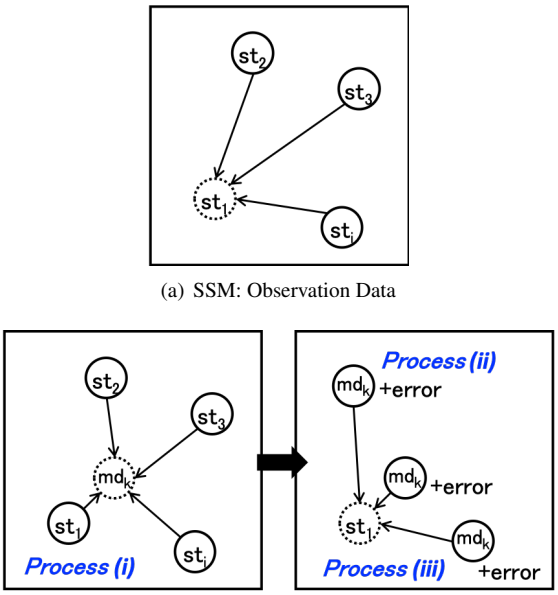
3.3 Main Objective of This Paper

Of the three phases required to develop our system, this paper focuses on Phase 1: SNS Data Validation. This phase is shown in the blue box in Fig. 1. The objective is to investigate whether SNS data containing various errors (sometimes the errors are considered to have a significant impact) are effective in improving the accuracy of data assimilation and flood estimation. The ③′validation data are simulated as expected data that would be posted based on the characteristics of the SNS data. We use Twitter as the SNS data source. These SNS data contain various errors in location, timing of postings, and flood water levels. However, these data are generated by a simple simulation and are not fully replicated in the SNS data.

When using SNS data for analysis, we need a process to convert text- and image-based posts into numerical values. The quantification of SNS data does not ensure an accurate calculation of flood levels, since the data representation of text-based SNSs is ambiguous; thus, it affects the quantification of flood levels. SNS data reporting water levels during flooding could include measurement of water levels (e.g., 30 cm) or be expressed in comparison to a body (e.g., up to his or her knee). Not only water levels but also location and timing contain errors in quantification, depending on the type of data representation. We investigate the SNS data impact considering this error on statistical flood analysis, such as data assimilation.

3.4 SNS Data Validation for Data Assimilation

The basic idea of Phase 1 is described as follows: SNS Data Validation for Data Assimilation in this paper is illustrated in Fig. 2(a), state-space model (SSM) using SNS Data. The purpose of SNS data validation is to investigate whether SNS data can contribute to the precise estimation of flood assessment and under what conditions SNS data can improve accuracy. Since we are unable to determine the error rate contained in the SNS data collected at the time of flooding, we generate alternative quantified SNS data from simulations. To generate the SNS data, we use time-series data collected from observation devices at multiple locations. The results of flood



(a) SSM: Observation Data

(b) SSM: SNS Data

Figure 2: SNS Data Validation for Data Assimilation ($st_{1,\ldots,i}$:observation location, md_k:SNS data posting location)

analysis simulation are assimilated with the time-series data to simulate the SNS locations, the timing of postings, and numerical flood levels (Fig. 2(b)(i)). For this data assimilation, we used the spatial-temporal SSM proposed in our previous study [22](Fig. 2(a) SSM using Observation Data). Note that, a spatial-temporal state-space model in this paper, is applied without using the waterway and sewer data used in the previous study [22], because these data are generally limited in availability.

Then, we determined the errors in the simulated SNS data (Fig. 2(b)(ii)). This process assumes the errors in location, time, and water level value that are present in the textual data of the actual SNS data. Afterward, we regenerate a flood level on the observation location based on the simulated SNS data with errors and examine the accuracy of the errors on the data assimilation accuracy (Fig. 2(b)(iii)). This regeneration uses a state-space model that applies the state-space model of the previous study [22] toward the spatial direction. Here, if there is a small difference between the time-series data and the data assimilation results at the observation location, the SNS data are applicable to flood assessment by data assimilation. In contrast, if the data assimilation accuracy is low despite the small error appended to the simulated SNS data, then there are problems using the SNS data.

3.4.1 Process(i) Simulated SNS data for Flood Analysis Simulation

The process flow is shown in Fig. 3. In process(i), the results of the flood analysis simulation are assimilated with time-series data collected from observation locations to simulate the locations, timing of posting, and flood water level values. This section outlines the spatial-temporal state-space model used in our data assimilation. The basic flood analysis is

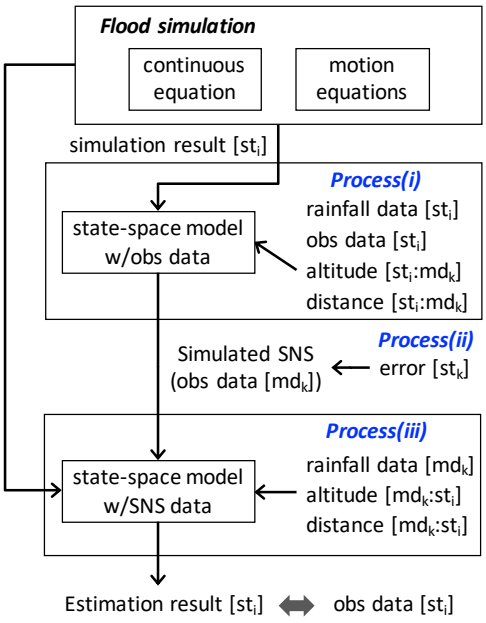


Figure 3: SNS Data Process Flow ($st_{1,\ldots,i}$:observation location, md_k:SNS data posting location)

based on a conventional simulation that uses a surface flooding model. This method calculates the amount of runoff at each grid location by expressing the flooding flow as a continuous equation and motion equations.

A continuous equation is defined as follows.

$$\frac{\partial h}{\partial t} + \frac{\partial M}{\partial x} + \frac{\partial N}{\partial y} = 0 \tag{1}$$

The motion equations are given as follows:

$$\frac{\partial M}{\partial t} + \frac{\partial UM}{\partial x} + \frac{\partial VM}{\partial y} + gh\frac{\partial H}{\partial x} + \frac{1}{\rho}\tau_x(b) = 0 \tag{2}$$

$$\frac{\partial N}{\partial t} + \frac{\partial UN}{\partial x} + \frac{\partial VN}{\partial y} + gh\frac{\partial H}{\partial y} + \frac{1}{\rho}\tau_y(b) = 0 \tag{3}$$

Each parameter is defined as t: time, H: water level, h: flood level, U: flow velocity (X direction), V: flow velocity (Y direction), g: gravity acceleration, ρ: water density, M: flux (X direction), and N: flux (Y direction) ($M = uh$, $N = vh$).

Here, the shear force in the x direction $\tau_x(b)$ and the shear force in the y direction $\tau_y(b)$ are defined as follows.

$$\tau_x(b) = \frac{\rho g n^2 \overline{U}\sqrt{U^2+V^2}}{h^{\frac{1}{3}}} \tag{4}$$

$$\tau_y(b) = \frac{\rho g n^2 \overline{V}\sqrt{U^2+V^2}}{h^{\frac{1}{3}}} \tag{5}$$

The roughness coefficient n (the resistance value of river water to touch obstacles) can be expressed as follows, considering the influence of a building.

$$n^2 = n_0^2 + 0.020 \times \frac{\theta}{100-\theta} \times h^{\frac{4}{3}} \tag{6}$$

(n:bottom roughness coefficient, n_o:composition equivalent roughness coefficient,and θ:building occupancy rate)

Equation (1)-(3) calculates flood level h for each grid, accounting for the runoff from the inside of the sewer line to the ground surface and the flooding to the ground surface due to rainwater. For the above equations, the inflow into each grid represents the flux into each grid from adjacent grids and the effect of buildings on the inflow in each grid.

We define D as the two-dimensional space corresponding to the region of interest and divide D into m grids of d meters each. Let $s_i \in D$ denote the location coordinates of each grid (s_i is denoted by i). Using equation (1)-(3), we calculate $h_t(i)$ for the flood level of each grid at time t.

Then, using the state-space model, we estimate the flood level of grid s_k from the observations $y_t^{(i)}$ collected at the observation location at time t. Grid $s_k(k = 1, 2, 3...., m)$ is the location indicated by the SNS data. The state-space model is represented by two types of observation equations; the flood analysis simulation result $h_t(i)$ at grid s_i, and the difference between the flood analysis simulation and the observed value at the observation location. This state-space model is defined by the equations (7)(8)(9).

$$\boldsymbol{y_t^{(i)}} = S_t \boldsymbol{r_t^{(i)}} + \boldsymbol{G_t^{(i)}} \boldsymbol{x_t^{(i)}} + \boldsymbol{e_t^{(i)}} \tag{7}$$

$$\boldsymbol{r_t^{(i)}} = \boldsymbol{r_{t-1}^{(i)}} + \boldsymbol{v_t^{(i)}} \tag{8}$$

$$\boldsymbol{x_t^{(i)}} = \boldsymbol{x_{t-1}^{(i)}} + \boldsymbol{u_t^{(i)}} \tag{9}$$

The $\boldsymbol{r_t^{(i)}}$ denotes the state at time t and $\boldsymbol{v_t^{(i)}}$ denotes noise. The term $\boldsymbol{G_t^{(i)}} \boldsymbol{x_t^{(i)}}$ represents the total inflow/outflow, and $\boldsymbol{x_t^{(i)}}$ is the difference between the flood analysis simulation results and the observed values. The $\boldsymbol{u_t^{(i)}}$ denotes the noise at time t. $\boldsymbol{G_t^{(i)}}$ is the adjacency matrix indicating the spatial component.

3.4.2 Process(ii) Appending Errors to Simulated SNS Data

Now, in process(ii), we append the error component to the simulated SNS data. Actual SNS data show a variety of representations of water levels. For example, "It is flooded up to my knees," "The car is flooded," or pictures are posted with comments such as "It is raining so hard." This paper considers SNS data that express water levels in words or show flood condition. The flood level $h_t(k)$ indicated by the SNS data is assumed to contain an error component e. The represen-tation type of the SNS data is considered to be a factor that causes errors due to the quantification of the SNS data (Table 1). We assume that the type of data representation occurs for each of the posted location, time, and water level values. For water level values, SNS data can be expressed in the form of measurement, comparison with an object, or description of the situation. In the case of measurement, it is considered to be measured by a guess, which results in a difference from the actual water level.

When compared with objects, water levels are explained based on objects such as knee height or up to the ankles, but

the sizes of these objects vary from each user, so even if quantified, they differ from the actual water level. Although this is only an assumption, the average length below the knee for Japanese people is 46.7 cm for males and 42.9 cm for females, a difference of approximately 4 cm even in the average value. In the describing the situation, the data mostly describe the flooding aspect, with little mention of water levels; consequently, it is expected that quantification itself is often difficult. In cases of pictures showing flood conditions, errors could be contained during the estimation process of converting the images to numerical data.

For information representation of location, the following information can be considered: GPS, address, road/river, landmark, and city/town name. When GPS data are attached to SNS data, the exact location at the time of posting can be reflected in the quantified SNS data. However, if the location indicated by the posted message differs from the location at the time of posting, there is an error compared with the posted location. The same error occurs for other types of data representations. In some cases, addresses of flooded areas are posted for rescue in flooding situations. Although there may be an error of a few meters, the location information would be approximately correct. If a road/river is described as a location, it is considered difficult to determine the exact location from the text content itself. In the case of landmarks, the data may indicate the location in front of the landmark, whereas it is also possible that the data indicate the location where the landmark is visible, in which case a large error of approximately 100 meters or more would occur. For city/town names, we consider a significant error of several kilometers when identifying the location due to the wide range of areas indicated by the data.

For the time information, the following three forms are considered: timestamp, comparison, and date/time range. The timestamp shows the exact date and time in the simulated SNS data. For comparison, it is considered to be a popular form of time; nevertheless, representations, such as "just now" are likely to include an error of several tens of minutes, as the sense of time differs among individuals. Additionally, it is assumed that there are many cases describing a range of dates/times, such as "approximately 21:00" or "this evening". While errors are expected to be small for numerical time representation, in the case of "night" and other representations, errors are likely to be on the order of several hours. Furthermore, as with location, there are cases in which the time of posting also differs from the time of flooding. This difference may result in a significant error in the time representation.

We define the error components at location s_k, resulting from the quantification, as the error in the representation type $e_{\Delta v,\Delta l,\Delta t}(k)$, the error relative to the posting location/time $\zeta_{\Delta l'}(k)$, and the error from the time of posting $\zeta_{\Delta t'}(k)$.

3.4.3 Process(iii) Flood Level Estimation and its Validation

Process(iii) regenerates the time-series data for the observation location to investigate the error impact on the data assimilation accuracy based on the simulated SNS data with error $h_t(k) + e_{\Delta v,\Delta l,\Delta t}(k) + \zeta_{\Delta l',\Delta t'}(k)$. Here, we employee a state-space model that utilizes the model detailed in 3.4.1 in the spatial direction. For equation (7)(8)(9), at time t, the simulated SNS data $h_t(k) + e_{\Delta v,\Delta l,\Delta t}(k) + \zeta_{\Delta l',\Delta t'}(k)$ is substituted into $y_t^{(i)}$ to estimate the flood level $h'(k')$ of the target location $s_{k'}$. Simulated SNS data on a particular location is not continuous time-series data. Thus, there is only one t in the simulated SNS data, and the state-space model in process(iii) is applied only applied spatially.

The difference between the restored water level $h'(k')$ and the actual water level $h(k')$ is shown as the effect of the error component $e_{\Delta v,\Delta l,\Delta t}(k) + \zeta_{\Delta l',\Delta t'}(k)$ on the data assimilation method. This paper validates the SNS data effectiveness by comparing flood level $h'(i)$ regenerated from the simulated SNS data at location k with the actual observed water level $h(i)$.

Table 1: SNS Data Representation Types

Data	Types	Example
Level	measurement	30 cm
	comparison	knee height
	description	looks like river
Location	GPS	34.9104, 135.8002
	address	1-1 Gokasho,Uji-city,
	road/river	Route 24
	landmark	In fromt of Kyoto Station
	city/town name	Uji city
Time	timestamp	2022/7/8/21:00
	comparison	just now
	range	approximately 21:00

3.5 Evaluation and Discussion

3.5.1 Evaluation Purpose and Flood Case

This evaluation investigates the data representation effect that text-based SNSs have on flood level quantification. SNS data reporting water levels during floods often include the measurement of water levels or are expressed in comparison to a body of water. In addition to water levels, location and time also contain errors in quantification, depending on the type of representation. We investigate the SNS data impact with these errors on statistical flood analysis, including data assimilation. Based on the results, we will discuss what SNS data form would contribute to flood assessments.

Our evaluation is based on SNS data simulated by a state-space model with observed flood data. We append errors to the simulated SNS data and observe the effects of the errors. Then, we determine errors that could occur due to the quantification of the SNS data, as shown in Fig. 4. This paper uses flood observations collected in Tsushima city, Aichi Prefecture, Japan, from October 22 to 23, 2017, due to rainfall caused by Typhoon 23. A rainfall amount of 32 mm/h was observed at 23:00 at the precipitation gauge nearest to our target area (Aisai Observatory, Aichi Prefecture). Using these rainfall data as input values, we calculated a flood analysis simulation with the flood analysis simulation NILIM 2.0. With the results of the simulated flood analysis, we apply the flood estimation method using the state-space model. To generate

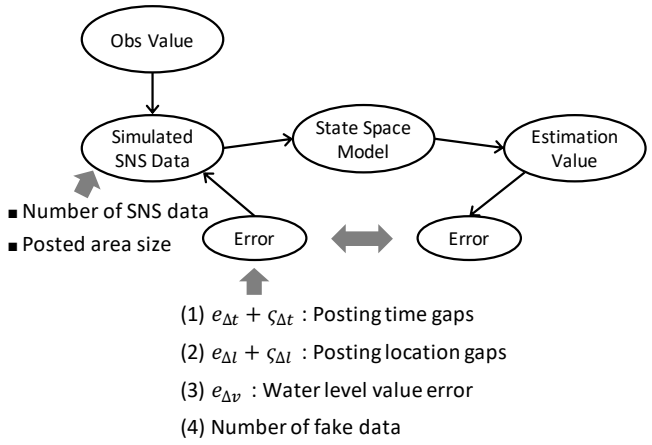


Figure 4: Appending Errors to Simulated SNS Data

simulated SNS data, the state-spatial model with observation data uses water level observation data at waterways collected every 5 minutes from pressure-type sensors installed at four locations in the target area.

Detailed information about the observation locations is presented as follows. Observation locations 1 and 2, and 3 and 4 are on the same waterway. The distance between observation locations 1 and 2 is approximately 500 meters, and the distance between locations 3 and 4 is approximately 600 meters. Observation location 4 is connected to the sewer. Two waterways are approximately 500 meters apart. There are no floodgates between the observation locations. The difference in elevation in this area is within 0.30 meters, and the elevation values (elevation model by the Geospatial Information Authority of Japan) are equal at all four observation locations. The heights from the bottom of the waterway to the road are 1.01, 1.14, 0.72, and 1.28 meters, and the usual water levels are 0.06, 0.07, 0.16, and 0.31 meters. On the day of the flood, the installed sensor devices showed, water overflowing the waterways and flood levels of up to 0.26, 0.25, 0.63, and 0.48 meters above the road.

3.5.2 Evaluation Procedure

The evaluation randomly extracts simulated SNS data according to the number of SNS data from the posted area size in Table 2. Errors appending to the simulated SNS data are parameterized to indicate fluctuations. The parameters that indicate the fluctuation of each error are shown in Table 2. We begin by using the number of SNS data posted and the area size where SNS data were posted as two common parameters. Subsequently, we adopt four different parameters to generate the errors as in (1)-(4). We repeat the above procedure five times and compute the mean value of the estimation. A total number of 489,637 simulated SNSs appended with the following errors are applied to the state-space model to calculate the estimated water levels for the four observation sites with time-series data.

Evaluation (1) There are two types of time information errors originating from the representation of time information: fluctuations in the estimation of time information from posted SNS data, and fluctuations due to the gap between the posted time and the flooding conditions indicated by the posting messages. We append these two fluctuations of time information, $e_{\Delta t}(k)$ and $\zeta_{\Delta t'}(k)$, to the simulated SNS data as posting time lags behind flood conditions. The value of the parameter: posting time lags in Table 2 adds a delay to the time of the simulated SNS data.

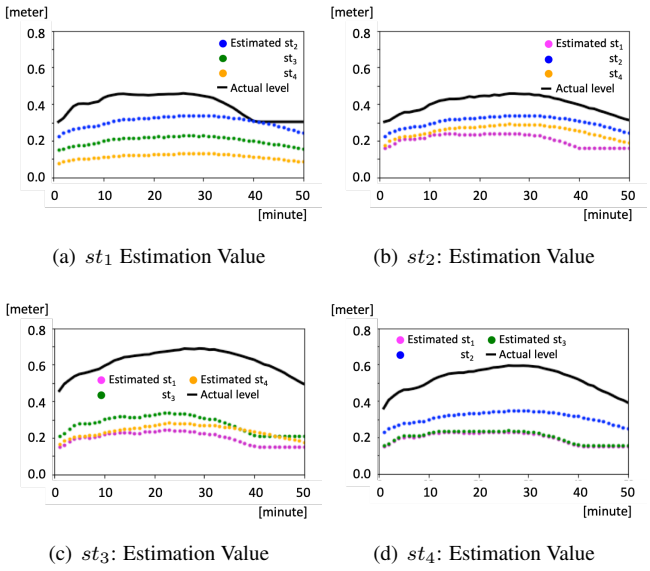
Table 2: Parameters related to Error Fluctuations

Parameters	Fluctuations
Number of SNS data	8,24,48,80,120,435
Posted area size (Radius)	10,20,30,40,50, 100[meters]
(1)Posting time lags	10,20,30,40[minites]
(2)Posting location (Radius)	10,20,30,40,50, 100[meters]
(3)Water level value error	0,10,20,30[cm]
(4)Number of fake data (Ratio)	10,20,30,40,50, 60,70,80,90[%]

Evaluation (2) Location errors originating from the representation of location information can be considered as fluctuations in location information when location information is estimated from posted SNS data, and fluctuations due to the gap between the location indicated by the posted messages and posting location. These two location fluctuations $e_{\Delta l}(k)$ and $\zeta_{\Delta l'}(k)$ are appended to the simulated SNS data as posting location gaps. We randomly swap the location information of the simulated SNS data within the radius area indicated by the parameter: posting location gaps in Table 2.

Evaluation (3) The error, resulting from the information representation of the water level value, could be the fluctuation of the value when the water level is estimated from the posted SNS data. Although this fluctuation can have a range of different values, this paper assumes a fixed parameter for the error in order to verify the effect of the SNS data. We add the value indicated by the parameter: water level value error in Table 2 to the water level in the simulated SNS data.

Evaluation (4) Furthermore, SNS data can be considered to include the posting of fake data. To evaluate the effect of the fake data on the assessment of flooding, we substitute the simulated SNS data with the fake data. We prepare two types of fake data; fake data that posts under flooding conditions "no flooding is occurring (water level: 0 m)", and fake data that post regardless of the current water level "flooding is occurring to the extent of the first floor of a building (water level: 1.5 m)". We parameterize the ratio of fake data among the simulated SNS data to calculate the estimated water level. Then we substitute 0 or 1.5 meters of simulated SNS data at the rate indicated by the parameter number of fake data in Table 2.



(a) st_1 Estimation Value (b) st_2: Estimation Value

(c) st_3: Estimation Value (d) st_4: Estimation Value

Figure 5: Preliminary result: SSM with Observation Data

Table 3: RMSE: SSM with Observation Data

Location	mean	minimum	maximun
st_1	0.19	0.12	0.24
st_2	0.20	0.15	0.23
st_3	0.38	0.28	0.43
st_4	0.28	0.18	0.33

3.5.3 Results

Preliminary result: SSM with Observation Data To compare accuracies, this section shows the estimated flood level $l^{(i)}_{t_T,k}$ using the state-space model with observation data (Fig. 2 SSM using observation data). Of the four observation locations, we apply the observation data from three locations (from time $t = 0$ to $t = 50$) to "SSM using Observation Data" to estimate the flood water level at the remaining one location (the water level at st_1 is estimated using the water level time-series data at st_2, st_3, and st_4). The estimation results are shown in Fig. 5. The black lines indicate the actual observed flood water levels, and the magenta, green, blue, and orange dots indicate the estimated water levels using the state-space model. Root Mean Squared Error: RMSE between the mean of the estimation results and the actual observed values is shown in Table 3.

In Fig. 5 (a), the estimated water level at st_1 was the most accurate using the data from st_3, with an average RMSE of 0.19 meters. The estimated water level for st_2 in Fig. 5 (b) was nearly the most accurate, with an average RMSE of 0.20 meters, using data from st_3. Both the estimation for st_3 in Fig. 5 (c) and st_4 in and Fig. 5 (d) using data from other observation locations were less accurate, with average RMSEs of 0.38 meters and 0.28 meters, respectively. The results of either estimation resulted in a large difference from the water level at the flood peak. These results are due to the failure of the "SSM using Observation Data" to follow the rising and falling water levels. In our previous study [22], the maximum error was 9 cm because we included data of waterways and sewers. However, this paper does not use those data, in order to apply our system in areas where it is difficult to collect the data. For all observation locations, we found that estimation using data from distant observation locations resulted in large errors.

Table 4: RMSE: Result(1) : Posting Time Lags

Time Lags [minutes]		10	20	30	40
st_1	mean	0.09	0.10	0.12	0.14
	minimun	0.00	0.00	0.00	0.06
	maximun	0.42	0.45	0.29	0.24
st_2	mean	0.06	0.06	0.07	0.09
	minimun	0.00	0.00	0.00	0.00
	maximun	0.51	0.42	0.26	0.23
st_3	mean	0.16	0.16	0.16	0.12
	minimun	0.00	0.00	0.00	0.00
	maximun	0.31	0.31	0.30	0.24
st_4	mean	0.09	0.09	0.08	0.05
	minimun	0.00	0.00	0.00	0.00
	maximun	0.52	0.46	0.22	0.31

Result (1) : Posting Time Lags This section describes the results of the estimation when time lags are appended to the simulated SNS data. The values of the time lags are generated as delays of 10, 20, 30, and 40 minutes. The RMSEs between estimated flood levels and actual observations applied to the state-space model are shown in Table 4. For st_1 and st_2, the larger the time delay is, the larger the mean value of the RMSE is. However, for st_3 and st_4, the larger the time lags are, the smaller the RMSE is. When the time delay was 10 minutes, the estimated water level difference was 0.09 meters, an improvement by 10 cm from the estimated value in 3.5.3. The minimum value of RMSE was 0.00 meters for all st_i, equal to the observed value. On the other hand, the maximum value of RMSE was 0.52 meters, resulting in a large error.

We found that the large time lag did not result in a very significant effect on the estimated water level. One reason for this may be that the actual flood levels were not large, with a maximum of 0.6 meters, and there was no sudden water level rise at any of the observation locations. However, it was found that the time delay in the SNS data was allowable for a flood of this scale.

Result (2) : Posting Location Gaps As in the Posting Location Gaps we rewrote the location information of the simulated SNS data within a specific area. We randomly changed the values of the appended location gaps to different location information from the true location of the area with a radius of 10, 20, 30, 40, 50, and 100 meters. Table 5 shows the RMSEs of the flood estimation results using simulated SNS data with location gaps. The minimum value was 0.00 meters, equal to the actual observed value in all observation locations. For st_1 and st_2, the RMSE increases with the location gap. The mean RMSE for st_3 and st_4 is approximately the same regardless of the location gap, whereas the maximum RMSE

Table 5: RMSE: Result(2)：Posting Location Gaps

Location Gaps [meter]		10	20	30	40	50	100
st_1	mean	0.08	0.08	0.08	0.08	0.09	0.09
	minimun	0.00	0.00	0.00	0.00	0.00	0.00
	maximun	0.23	0.25	0.24	0.25	0.26	0.29
st_2	mean	0.06	0.06	0.06	0.06	0.06	0.07
	minimun	0.00	0.00	0.00	0.00	0.00	0.00
	maximun	0.19	0.19	0.20	0.19	0.19	0.33
st_3	mean	0.15	0.15	0.14	0.14	0.14	0.16
	minimun	0.00	0.00	0.00	0.00	0.00	0.00
	maximun	0.29	0.30	0.29	0.29	0.29	0.30
st_4	mean	0.09	0.09	0.08	0.08	0.08	0.08
	minimun	0.00	0.00	0.00	0.00	0.00	0.00
	maximun	0.22	0.22	0.34	0.30	0.37	0.58

results in a larger RMSE as the location gap increases. Comparing only the mean values, the RMSE values are almost the same for any st_i. This finding can be explained by the fact that our location gap area radius is at most 100 meters, which is a small area. Thus, the estimation results indicate the possibility of estimating the flooding situation for flooding of this scale, even if there is a gap in the posting location within these area sizes.

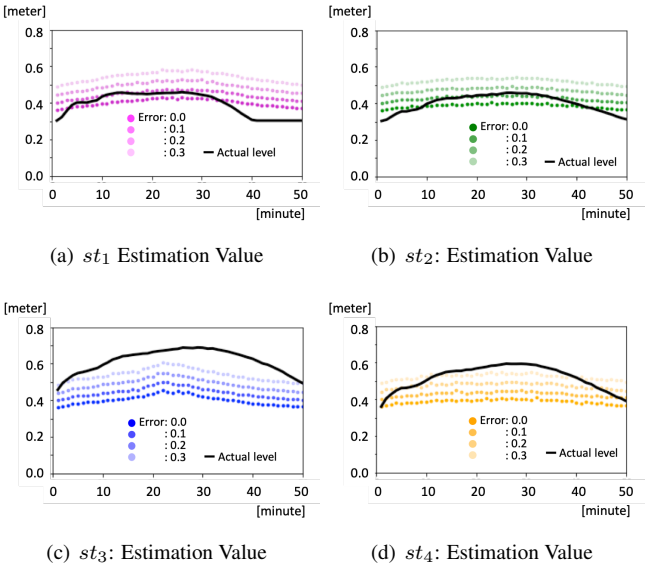
Result (3)：Water Level Value Error This section shows the results of applying the state-space model with error values appended to the simulated SNS data as Water Level Value Error in Table 6. When no error values were added, the mean values of st_1 and st_2 showed the smallest RMSE. The larger error value resulted in a larger RMSE. On the other hand, st_3 and st_4 showed a large RMSE for mean value.

Figure 6 compares the time-series data of the actual observations with the mean value of the estimated values. For st_1 and st_2, as the water level changes, the water level estimated from the SNS data also changes and shows little difference from the actual observation. In addition, larger error values tend to provide larger estimates of the results. Moreover, ST_3 and ST_4 result in a difference by approximately 0.20 meters between the estimated result (error value: 0) and the actual water level at the peak of the flooding. We consider that the estimated values were closer to the actual observed value when the error value was increased since there was such a large difference at the error value of 0. One possible reason for this is that the values of st_3 and st_4 were estimated to be lower due to the use of data from other observation locations when generating the simulated SNS data.

Table 6 shows that the minimum value is almost 0.00 meters, even when large values are added as errors. However, the maximum value results in an extremely large value. In st_1 and st_2, the mean values of RMSE do not change significantly after adding the error value, contrary to this the error value: 0.30 meters in st_4 shows an RMSE of 0.52 meters, which is a large error. Accordingly, this would require a data collection method and analysis process that reduces the error in values, and a process to validate the reliability when large water levels are estimated would be required.

Table 6: RMSE: Result(3)：Water Level Value Error

Value Error [meter]		0.00	0.10	0.20	0.30
st_1	mean	0.04	0.05	0.10	0.15
	minimun	0.00	0.00	0.01	0.06
	maximun	0.13	0.17	0.27	0.30
st_2	mean	0.04	0.03	0.07	0.12
	minimun	0.00	0.00	0.00	0.03
	maximun	0.26	0.15	0.25	0.39
st_3	mean	0.21	0.17	0.13	0.08
	minimun	0.00	0.03	0.00	0.00
	maximun	0.30	0.28	0.25	0.22
st_4	mean	0.13	0.09	0.06	0.05
	minimun	0.00	0.00	0.00	0.00
	maximun	0.22	0.18	0.37	0.52



(a) st_1 Estimation Value (b) st_2: Estimation Value

(c) st_3: Estimation Value (d) st_4: Estimation Value

Figure 6: Result(3)：Water Level Value Error (Mean Value)

Result (4)：Number of Fake Data We describe the estimation results of substituting simulated SNS data for fake data. The fake data were either 0 or 1.5 meters values at 10, 20, 30, 40, 50, 60, 70, 80, and 90% of the simulated SNS data used in the state-space model. This evaluation did not add error values to the water level values.

Table 7 shows an abstract of the RMSE results. Compared to Table 6, the RMSE was larger and less accurate for the results with the addition of the fake data. The RMSE becomes larger as the ratio of fake data increases. When the ratio of fake data 0 is 90%, st_3 has an error of 0.69 meters. When the fake data were set to 1.5 meters, a larger error resulted, with a maximum error of 1.20 meters. In st_2, with 0 fake data, the maximum error was 1.5 meters. The process of averaging offsets this error value, as the amount of fake data is as small as 10%, resulting in a mean value of 0.15 meters.

Table 7 indicates that if the ratio of fake data is as small as 10 or 20%, the error would not be significant. Moreover, st_3, where the ratio of fake data is even 20%, shows a large error in the mean RMSE, This arises from low estimation accuracy even without error value st_3. This evaluation did not add error

Table 7: RMSE: Result(4)：Number of Fake Data

Ratio [%]			10	30	50	70	90
st_1	0.0	mean	0.07	0.13	0.21	0.29	0.36
		minimun	0.00	0.00	0.00	0.00	0.00
		maximun	0.46	0.46	0.46	0.46	0.46
	1.5	mean	0.15	0.36	0.57	0.81	1.02
		minimun	0.00	0.00	0.00	0.00	0.00
		maximun	1.20	1.20	1.20	1.20	1.20
st_2	0.0	mean	0.07	0.14	0.21	0.30	0.37
		minimun	0.00	0.00	0.00	0.00	0.00
		maximun	1.50	0.46	0.46	0.46	0.46
	1.5	mean	0.14	0.34	0.56	0.80	1.01
		minimun	0.00	0.00	0.00	0.00	0.00
		maximun	1.50	1.20	1.20	1.20	1.20
st_3	0.0	mean	0.26	0.34	0.42	0.51	0.58
		minimun	0.01	0.01	0.01	0.07	0.07
		maximun	0.69	0.69	0.69	0.69	0.69
	1.5	mean	0.22	0.20	0.44	0.63	0.81
		minimun	0.00	0.00	0.00	0.00	0.00
		maximun	1.01	1.04	1.04	1.04	1.04
st_4	0.0	mean	0.17	0.25	0.33	0.41	0.49
		minimun	0.00	0.00	0.00	0.00	0.00
		maximun	0.59	0.59	0.59	0.59	0.59
	1.5	mean	0.18	0.32	0.50	0.71	0.90
		minimun	0.00	0.00	0.00	0.00	0.00
		maximun	1.14	1.14	1.14	1.14	1.27

values to the simulated SNS data other than the fake data. We also consider that adding error value, time lags, and location gaps to the SNS data will further increase the error.

3.5.4 Discussion

The evaluation applied simulated SNS data with error data appended to the data as fluctuations in four aspects (1)-(4), to compare the estimated results to actual water levels. (1) time lag and (2) location gap showed that a narrow range of fewer than 100 meters does not significantly affect the RMSEs. For (3) water level error and (4) fake data, we found that as long as the error is small and the ratio of fake data is small, the estimation accuracy does not reduce significantly. All evaluations showed significant improvements compared to the state-space model with observations. This finding shows that the observation locations are approximately 500 meters away from the estimated locations, whereas the simulated SNS data are within 100 meters, allowing for more accurate estimation. Unlike time-series data, SNS data are sparse in the time direction, although it is effective for estimation if collected at locations that are near the estimated location.

In this evaluation, we also treated the number of SNS data and posted area size as parameters. These two parameters did not significantly affect the estimation results. Hence, we found that even if the number of SNS data is small, locations near the estimation location can be estimated with sufficient accuracy. For st_1 and st_2, the smaller the parameters (1)-(4) are, the smaller the error is. These results indicate that time lag and location gap are allowable within the range of values of this evaluation and that a small number of errors in water level values and fake data prevents a large error. For st_3 and st_4, the errors are large even without adding the water level error, requiring investigation as to the cause. Since this paper covers only four observation locations, we assume that this is due to the effect of low water levels observed at the other locations. The process in 3.4.1 calculated a low-accuracy estimation of the observed location. As a result, the simulated SNS data based on the estimation are also to be lower than the actual water level. Therefore, we need to improve the state-space model itself. One idea is to apply observations and SNS data together to the state-space model and implement a Kalman filter for locations where time-series data are available.

The purpose of our research is to investigate the conditions under which SNSs can contribute to flood estimation: how SNS data, if available, can improve the accuracy of flood analysis. To achieve the system in Fig. 1 Phase 1 validated the effectiveness of SNS data in improving the accuracy of flooding assessments as SNS data validation. The results of the assessment using simulated SNS data showed that a certain degree of error was allowable and that the accuracy was better than the estimation using observation data collected over a longer distance. We conclude that further processing improvements are needed, such as removing larger errors by estimation using a combination of observed and SNS data.

4 CONCLUSION

This study investigated whether SNS data can be used to assess flooding. We believe that SNS data can be effective in determining flood levels even in places where it is difficult to install, operate, and manage observation devices. Although there have been many studies on flood damage detection using SNSs, their effectiveness has not been clarified, and the amount and content of data collected are not fixed depending on the case of flood damage. This estimation method could further improve the accuracy of flooding assessment by adding SNS data as well as observation data and calculating flooding conditions for the entire affected area. Thus, we considered it necessary to validate the effectiveness of the SNS data. For the estimation, we utilized the method we have developed for estimating the expansion process of flooding using observation time-series data. This is a state-space model that uses observation data to compensate for flooding analysis simulations.

This paper validates the SNS data using the following procedure based on our state-space model. The purpose of SNS data validation is to investigate whether SNS data can contribute to the accuracy of flood assessment and under what conditions SNS data can improve the accuracy. Since we were unable to determine the error rate contained in the SNS data collected at the time of flooding, we generate quantified SNS data from simulations in this paper. To generate the SNS data, we used time-series data collected from observation devices at multiple locations. The results of flood analysis simulation were assimilated using the time-series data to simulate the SNS locations, the timing of postings, and numerical flood levels (Fig. 2(b)(i)). Then, we appended errors to the simu-

lated SNS data (Fig. 2(b)(ii)). Afterward, we regenerated a flood level at the observation location based on the simulated SNS data with errors and examined the accuracy of the errors on the data assimilation accuracy (Fig. 2(b)(iii)).

The estimation results show that even if the posted SNS data have a time lag and location gap, the error is small, averaging approximately 0.10 meters for a small area within 100 meters, except for a certain observation location. In cases where water levels contained errors and fake data were posted, we found that the errors were small and that if the ratio of fake data was small, the estimation accuracy would not be significantly reduced. All evaluations showed significant improvements compared to the state-space model with observations. Unlike time-series data, SNS data are sparse in the time direction, although it is effective for estimation if collected at locations that are near the estimated location. However, some of the sensors show large errors in the process of calculating simulated SNS data. These errors appeared because waterway and sewer data were not used in the state-space model. Since waterway and sewer data are difficult to collect, we plan to resolve this issue statistically using a Kalman filter. In addition, while this paper evaluates a flood case from a waterway, it is necessary to improve the accuracy by integrating observed time series data provided by water level observation devices so that the method can adapt to rapid water level rising, such as floods caused by large outflows from rivers. After a further improvement in accuracy, we will develop and implement Phase 2: Flood Assessment Promotion Requirement and Phase 3: Flood Assessment.

ACKNOWLEDGEMENT

This work was supported by JST, PRESTO Grant Number JPMJPR2036, and the commissioned research(No. 05401) by National Institute of Information and Communications Technology (NICT), Japan.

REFERENCES

[1] "Intergovernmental Panel on Climate Change Fifth Assessment Report (AR5)", Retrieved April 17, 2018, from https://www.ipcc.ch/report/ar5/.

[2] P. Milly, D. Christopher, T. R. Wetherald, A. K. Dunne, T. L. Delworth, "Increasing Risk of Great Floods in a Changing Climate, Nature, Nature Publishing Group", Vol.415, No.6871, page514, (2002).

[3] Y. Hirabayashi, R. Mahendran, S. Koirala, L. Konoshima, D. Yamazaki, S. Watanabe, H. Kim, S. Kanae, "Global Flood Risk under Climate Change", Nature Climate Change, Nature Publishing Group, Vol.3, No.9, page816, (2013).

[4] J. Ernst, J. B. Dewals, S. Detrembleur, P. Archambeau, S. Erpicum, M. Pirotton, "Micro-scale Flood Risk Analysis based on Detailed 2D Hydraulic Modelling and High Resolution Geographic Data", Natural Hazards, Springer, Vol.55, No.2, pp.181–209, (2010).

[5] A. S. M. Saudi, I. S. D. Ridzuan, A. Balakrishnan, A. Azid, D. M. A. Shukor, Z. I. Rizman, "New Flood Risk Index in Tropical Area Generated by using SPC Technique", Journal of Fundamental and Applied Sciences, Vol.9, No.4S, pp.828–850, (2017).

[6] S. F. Silva, M. Martinho, R. Capitão, T. Reis, C. J. Fortes, J. C. Ferreira, "An Index-based Method for Coastal-flood Risk Assessment in Low-lying Areas (Costa de Caparica, Portugal)", Ocean & Coastal Management, Elsevier, Vol.144, pp.90–104, (2017).

[7] D. H. Burn, "Perceptions of Flood Risk: A Case Study of the Red River Flood of 1997", Water Resources Research, Wiley Online Library, Vol.35, No.11, pp.3451–3458, (1999).

[8] J. W. Hall, I. C. Meadowcroft, P. B. Sayers, M. E. Bramley, "Integrated Flood Risk Management in England and Wales", Natural Hazards Review, American Society of Civil Engineers, Vol.4, No.3, pp.126–135, (2003).

[9] R. Miceli, I. Sotgiu, M. Settanni, "Disaster Preparedness and Perception of Flood Risk: A Study in an Alpine Valley in Italy", Journal of Environmental Psychology, Elsevier, Vol.28, No.2, pp.164–173, (2008).

[10] L. Pulvirentia, M. Chinib, N. Pierdiccaa, L. Guerrieroc, P. Ferrazzolic, "Flood Monitoring using Multi-temporal COSMO-SkyMed Data: Image Segmentation and Signature Interpretation", Remote Sensing of Environment, Vol.115, No.4, pp.990-1002, (2011).

[11] E. A. Basha, S. Ravela, D. Rus, "Model-based Monitoring for Early Warning Flood Detection", In Proceedings of the 6th ACM conference on Embedded network sensor systems (SenSys), pp.295-308, (2008).

[12] A. Elshorbagy, G. Corzo, S. Srinivasulu, D. Solomatine, "Experimental Investigation of the Predictive Capabilities of Data Driven Modeling Techniques in Hydrology —part 2: Application", Hydrology and Earth System Sciences, Vol.14, pp.1943-1961, (2010).

[13] A. Rafieeinasab, A. Norouzi, S. Kim, H. Habibi, B. Nazari, D. Seo, H. Lee, B. Cosgrove, Z. Cui, "Toward High-resolution Flash Flood Prediction in Large Urban Areas – Analysis of Sensitivity to Spatiotemporal Resolution of Rainfall Input and Hydrologic Modeling", Journal of Hydrology, Vol.531, part 2, pp.370-388, (2015).

[14] F. A. Ruslan, A. M. Samad, Z. M. Zain, R. Adnan, "Flood Prediction using NARX Neural Network and EKF Prediction Technique: A Comparative Study", In Proceeding of the IEEE 3rd International Conference on System Engineering and Technology (ICSET), pp.203-208, (2013).

[15] S. K. Sinnakaudan, A. A. Ghani, M. S. S. Ahmad, N. A. Zakaria, "Flood Risk Mapping for Pari River Incorporating Sediment Transport", Environmental Modelling & Software, Elsevier, Vol.18, No.2, pp.119–130, (2003).

[16] H. M. Lyu, W. J. Sun, S. L. Shen, A. Arulrajah, "Flood Risk Assessment in Metro Systems of Mega-cities using a GIS-based Modeling Approach", Science of the Total Environment, Elsevier, Vol.626, pp.1012–1025, (2018).

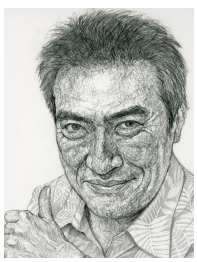
[17] N. M. Hunter, P. D. Bates, M. S. Horritt, M. D. Wilson, "Simple Spatially-distributed Models for Predicting Flood Inundation: A Review", Geomorphology, Elsevier, Vol.90, No.3-4, pp.208–225, (2007).

[18] M. S. Horritt, P. D. Bates, “Evaluation of 1D and 2D Numerical Models for Predicting River Flood Inundation”, Journal of Hydrology, Elsevier, Vol.268, No.1-4, pp.87–99, (2002).
[19] J. Kim, M. Hastak, “Social Network Analysis: Characteristics of Online Social Networks after a Disaster”, International Journal of Information Management, Vol.38, No.1, pp.86–96, (2018).
[20] F. K. Sufi, I. Khalil, “Automated Disaster Monitoring From Social Media Posts Using AI-Based Location Intelligence and Sentiment Analysis”, IEEE Transactions on Computational Social Systems, pp.1–11, (2022).
[21] H. N. Teodorescu, “Emergency-Related, Social Network Time Series: Description and Analysis”, Time Series Analysis and Forecasting, Springer International Publishing, pp.205–215, (2016).
[22] K. Hiroi, D. Murakami, K. Kurata, T. Tashiro, Y. Shinoda, “A Proposal of Data Assimilation Approach for Flood Level Estimation and Evaluation with Urban Flood Disasters”, Journal of Information Processing: Consumer Devices and Systems, Vol.10, No.3, pp.55–64, (2020) (In Japanese).

(Received: January 16, 2023)
(Accepted: August 5, 2023)

Kei Hiroi received her Master of Media Design and Ph.D. in Media Design in 2011 and 2014, respectively from Keio University. She has been an assistant professor in the department of Information and Communication Engineering, Graduate School of Engineering, Nagoya University. She is currently an associate professor in Disaster Prevention Research Institute, Kyoto University. Her research interests include disaster simulation, and crisis computing.

Akihito Kohiga is a Project Researcher of Information Sciences at Japan Advanced Institute of Science and Technology. He received Ph.D (Information Science) in 2020. His research interests lie in cloud computing, massive distributed simulation, modeling and architecture, especially in flood and evacuation fields.



Yoichi Shinoda is currently a professor of Japan Advanced Institute of Science and Technology. His research interests include Impact of digital technologies on human activities, Parallel and Distributed Systems, Networking Protocols and Systems, and Information Handlinf Systems.

Submission Guidance

About IJIS

International Journal of Informatics Society (ISSN 1883-4566) is published in one volume of three issues a year. One should be a member of Informatics Society for the submission of the article at least. Each submission article is reviewed at least two reviewers. The online version of the journal is available at the following site: http://www.infsoc.org.

Aims and Scope of Informatics Society

The evolution of informatics heralds a new information society. It provides more convenience to our life. Informatics and technologies have been integrated by various fields. For example, mathematics, linguistics, logics, engineering, and new fields will join it. Especially, we are continuing to maintain an awareness of informatics and communication convergence. Informatics Society is the organization that tries to develop informatics and technologies with this convergence. International Journal of Informatics Society (IJIS) is the journal of Informatics Society.

Areas of interest include, but are not limited to:

- Internet of Things (IoT)
- Smart Cities, Communities, and Spaces
- Big Data, Artificial Intelligence, and Data Science
- Network Systems and Protocols
- Computer Supported Cooperative Work and Groupware
- Security and Privacy in Information Systems
- Intelligent Transportation System
- Distributed Computing
- Multi-media communication
- Information systems
- Mobile computing
- Ubiquitous computing

Instruction to Authors

For detailed instructions please refer to the Authors Corner on our Web site, http://www.infsoc.org/.

Submission of manuscripts: There is no limitation of page count as full papers, each of which will be subject to a full review process. An electronic, PDF-based submission of papers is mandatory. Download and use the LaTeX2e or Microsoft Word sample IJIS formats.

http://www.infsoc.org/IJIS-Format.pdf

LaTeX2e

LaTeX2e files (ZIP) http://www.infsoc.org/template_IJIS.zip

Microsoft Word™

Sample document http://www.infsoc.org/sample_IJIS.doc

Please send the PDF file of your paper to secretariat@infsoc.org with the following information:

Title, Author: Name (Affiliation), Name (Affiliation), Corresponding Author. Address, Tel, Fax, E-mail:

Copyright

For all copying, reprint, or republication permission, write to: Copyrights and Permissions Department, Informatics Society, secretariat@infsoc.org.

Publisher

Address: Informatics Laboratory, 3-41 Tsujimachi, Kitaku, Nagoya 462-0032, Japan

E-mail: secretariat@infsoc.org

CONTENTS

Guest Editor's Message 113
Tetsuya Yokotani

Regular Paper 115
Proposal of a Broadcaster Support Method using MR Stamp in 360-degree Internet Live Broadcasting
Yoshia Saito and Kei Sato

Regular Paper 123
Feature Data Distribution Methods for Person Re-identification using Multiple Cameras
Satoru Matsumoto, Tomoki Yoshihisa, Tomoya Kawakami, and Yuuichi Teranishi

Regular Paper 133
A Study on Effectiveness of SNS Data in Flood Estimation
Kei Hiroi, Akihito Kohiga and Yoichi Shinoda