Special Issue: Measurement, control, and analysis of motion using ICT and AI

Preface to the Special Issue

Ryuichi Imai [1]

[1] Faculty of Engineering and Design, Hosei University, 2-33 Ichigaya-tamachi, Shinjuku-ku, Tokyo 162-0843, Japan

In recent years, the rapid advancement of ICT (Information and Communication Technology) and AI (Artificial Intelligence) has led to increased research activity in the measurement, control, and analysis of various moving entities, including humans and machines. These technologies are being applied across a wide range of fields, including construction, agriculture, manufacturing, healthcare, and sports, driving the development of innovative methodologies.

In the field of measurement technology, advancements in sensors and IoT (Internet of Things) have enabled real-time and highly precise data collection. For example, on construction sites, precise tracking of heavy machinery and workers contributes to improved safety and operational efficiency. In agriculture, drones and robots are being utilized to monitor crop growth, facilitating the advancement of precision farming.

In the domain of control technology, autonomous driving and robotics have made significant progress, enabling the independent operation of moving entities in diverse environments. In the manufacturing sector, AI-driven robotic control is enhancing productivity and reducing labor burdens. Similarly, in healthcare, the integration of AI into surgical support robots is enabling more precise medical procedures, marking a transformative shift in medical technology.

In the area of motion analysis, researchers are analyzing collected data to understand behavioral patterns and convert tacit knowledge into explicit knowledge. In sports science, motion analysis of athletes is aiding in performance enhancement, while in the medical field, research is being conducted to quantify rehabilitation effectiveness, facilitating more effective treatment strategies.

This special issue aims to gather the latest research findings on the measurement, control, and analysis of moving entities across various domains and to foster innovation through interdisciplinary and cross-sector collaboration. By facilitating knowledge exchange and cooperation among researchers and engineers from different fields, further technological advancements can be expected. We hope that this special issue will contribute to the continued evolution of both academic and technological frontiers.

Published: 26 March 2025
* Correspondence: imai@hosei.ac.jp
Publisher’s Note: JOURNAL OF DIGITAL LIFE. stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Digital Life

Special Issue: Measurement, control, and analysis of motion using ICT and AI
Article

Fundamental Study on Detection of Dangerous Objects on the Road Surface Leading to Motorcycle Accidents Using a 360-Degree Camera

Haruka Inoue [1], * and Yuma Nakasuji [1]

[1] Faculty of Information Technology and Social Sciences, Osaka University of Economics, 2-2-8 Higashiyodogawa-ku, Osaka 533-8533, Japan

Abstract

In recent years, the number of fatalities in traffic accidents involving motorcyclists has remained almost unchanged, with single-vehicle accidents accounting for 37.2% of all accidents by accident type in the past five years. In the development of overturn prevention devices for motorcycles, problems remain in post-mounting of the device as well as its downsizing. On the other hand, an existing study using deep learning has proposed a method for detecting dangerous objects on the road surface leading motorcycles to overturn, though this method still needs verification under different conditions. In this study, we apply a method for detecting dangerous objects on the road surface from video images using YOLO to two types of 360-degree cameras and verify that this method is versatile under different conditions.

Keywords: Motorcycle; Dangerous objects on the road surface; 360-degree camera; Deep learning.

1. Introduction

In recent years, the number of fatalities in traffic accidents involving motorcyclists has remained almost unchanged, and although the Metropolitan Police Department has been conducting motorcycle safety classes, the number of fatalities increased for all ages in 2023. Single-vehicle accidents accounted for 37.2% of all accidents by accident type in the past five years from 2018 to 2022 (Tokyo Metropolitan Police Department, 2024). The occurrence situations of traffic accidents resulting in injury or death in 2024 show that the number of fatalities from motorcycle accidents is about twice as high as that of automobiles accidents. Although ADAS (Nikkei xTECH, 2024), an advanced safety technology for motorcycles has been developed, its diffusion is slower than that for automobiles. Therefore, riders are required to follow the traffic rules and instantly predict danger. An existing study on the development of an overturn prevention device for motorcycles using the gyro effect suggests a need for downsizing the device (Senoo et al., 2017). A study on detecting dangerous objects as well as detection of dangerous objects that may cause motorcycles to overturn using deep learning (Inoue et al., 2023) as well as a study on detecting dangers leading to motorcycle accidents using 360-degree cameras (Inoue et al., 2024) show the difficult issue of verification under different conditions. In this study, we apply a method to detect dangerous objects on the road surface from video images using YOLO to two types of 360-degree cameras (hereinafter referred to as "Dangerous object detection method") and verify that this method is versatile. In this study, as with the existing studies, fallen leaves, gravel, manholes, bumps, and wet road surfaces are considered as dangerous objects on the road surface.

2. Method

Fig. 1 shows the process flow of the dangerous object detection method. The dangerous object detection method consists of a learning function and an estimation function. The input data for the learning function is the learning data, and the output data is the dangerous object detection model. The input data for the estimation function are video

Received: 22 November 2024, Accepted: 18 January 2025, Published: 26 March 2025

* Correspondence: h.inoue@osaka-ue.ac.jp

images taken by the 360-degree camera while riding a motorcycle, and the output data are the results of dangerous object detection.

The learning function builds up a learning model to detect dangerous objects on the road surface that may cause a motorcycle to overturn. Specifically, as shown in Table 1, the model to detect fallen leaves, gravel, manholes, bumps, and wet road surface as dangerous objects from video images (hereinafter referred to as "dangerous object detection model") by annotating dangerous objects on the road surface and learning them using YOLOv5.

The estimation function is used to detect dangerous objects on the road surface from video images captured by the 360-degree camera. In the image generation process, the THETA+ application is used to convert the display format to flat, and crop to 1.91:1, and cut out the video image at 3 fps. The dangerous object detection process is used to detect dangerous objects on the road surface using the dangerous object detection model built up by the learning function.

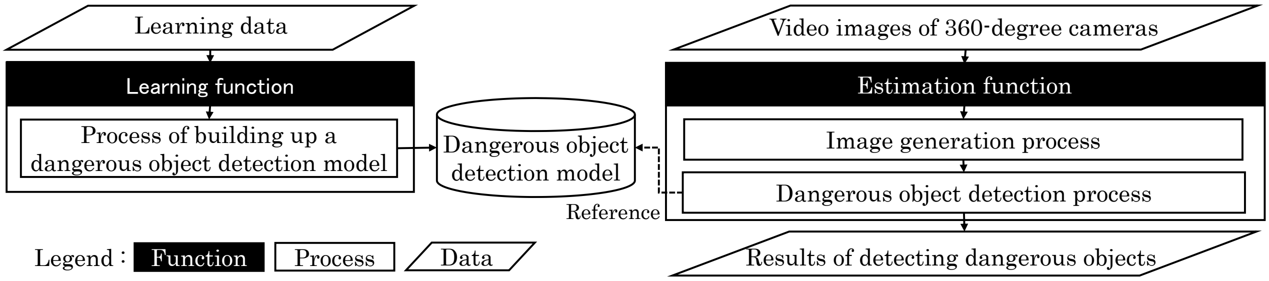
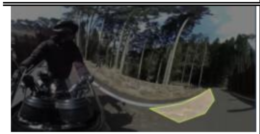
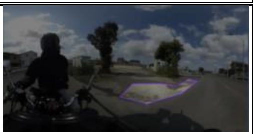

Learning data
Learning function
Process of building up a dangerous object detection model
Dangerous object detection model
Reference
Video images of 360-degree cameras
Estimation function
Image generation process
Dangerous object detection process
Results of detecting dangerous objects
Legend :
Function
Process
Data

Fig. 1. Process Flow

Table 1. Example of annotation

Fallen leave	Gravel	Manhole	Bump	Wet road surface
				

3. Demonstration Experiment

In this experiment, we confirm the versatility of the proposed method by applying it to the video images shot by using 360-degree cameras under different conditions regarding types of cameras, resolution, and the versions of YOLO. The experimental conditions for Experiments 1 through 3 are shown in Table 2.

3.1. Method of the experiment

First, this study targets two types of cameras: THETA SC and THETA V, both of which are products of RICHO. In this experiment, each of the 360-degree cameras are installed at the front part the motorcycle (Fig. 2). A male person in his 20s rides the motorcycle along the same section of road in the urban area of Wakayama Prefecture multiple times with the same speed as much as possible. Then, the results of detecting dangerous objects detected by applying the dangerous object detection method to the respective video images are compared with the manually generated correct-answer data, to make evaluation based on precision, recall, and F-measure. The learning model was built up using different images from the data used for the evaluation. For the learning data, the data shot by THETA SC on

Table 2. Experimental conditions

Experiment	Camera		Video resolution		YOLO	
	THETA SC	THETA V	Full HD	4K	v5	v8
1	○	○	○	-	○	-
2	-	○	○	○	○	-
3	○	-	○	-	○	○

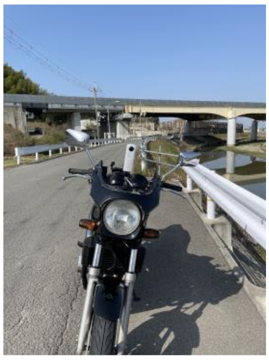


Fig. 2. Experimental view

March 20, 21, July 10, and 12, 2024 were used. 3,103 images were used for the dangerous object detection model. On the other hand, the evaluation data were those shot by THETA SC and THETA V on March 22 and July 11, 2024.

3.2. Experiment 1: Verification of versatility for different types of cameras

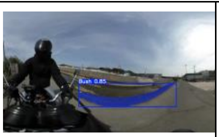
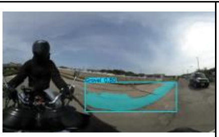
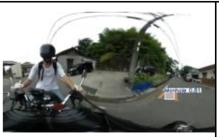
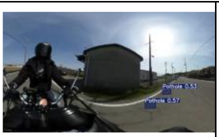
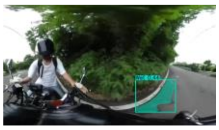
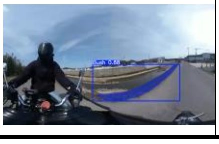
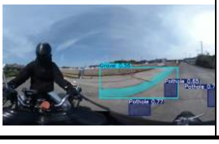
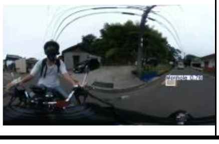
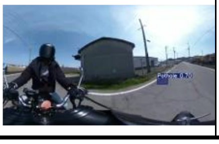
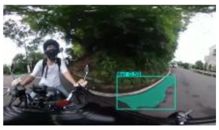
In Experiment 1, The dangerous object detection method is applied to the video images shot using THETA SC and THETA V to verify its versatility for different types of cameras. The ISO sensitivity differs between the two, ranging from 100 to 1,600 with THETA SC and from 64 to 6,400 with THETA V. The video resolution is 1,920 x 960 for both.

Table 3 shows the experimental results of Experiment 1, and Table 4 shows an example of the results of detecting dangerous objects. The overall F-measure is 0.917 for THETA SC and 0.839 for THETA V respectively, indicating that the dangerous object detection method is capable of detecting dangerous objects on the road surface correctly on the whole. Comparing the F-measure by camera type, the difference between THETA SC and THETA V was 0.078, which is not much difference. However, as the results of detecting bumps and wet road surface shown in Table 3, there were cases where only one of the 360-degree cameras was able to detect dangerous objects, even when the images were taken at the same point. Besides, comparing the F-measure by dangerous object type, the F-measure was lower for bumps and wet road surfaces than for fallen leaves, gravel, and manholes. Focusing on the result of detecting dangerous objects, there was a tendency of omission of detection for small bumps or wet road surfaces covered with shadows. We will increase the number of learning data under various environments and change the version of YOLO in order to advance the system.

Table 3. Experimental result of Experiment 1

Camera	Video resolution	YOLO	Dangerous objects	Fallen leave	Gravel	Manhole	Bump	Wet road surface	Total
THETA SC	Full HD	v5	Total number	20	23	11	47	11	112
			Number of determination cases	19	19	8	37	8	91
			Number of correct answers	19	19	8	37	8	91
			Precision	1.000	1.000	1.000	1.000	1.000	1.000
			Recall	0.950	0.826	0.727	0.787	0.727	0.813
			F-measure	0.974	0.905	0.842	0.881	0.842	0.897
THETA V			Total number	16	18	8	44	11	97
			Number of determination cases	15	13	7	33	7	75
			Number of correct answers	15	13	7	33	7	75
			Precision	1.000	1.000	1.000	1.000	1.000	1.000
			Recall	0.938	0.722	0.875	0.750	0.636	0.773
			F-measure	0.968	0.839	0.933	0.857	0.778	0.872

Table 4. Result of detecting dangerous objects with different cameras

Dangerous objects	Fallen leave	Gravel	Manhole	Bump	Wet road surface
THETA SC					
THETA V					

3.3. Experiment 2: Verification of versatility for different resolutions

In Experiment 2, the dangerous object detection method is applied to the video images shot using THETA V with resolutions of 1,920 x 960 and 2,840 x 1,920 to verify its versatility for different resolutions. It should be noted that as THETA SC only has a resolution of 1,920 x 960, it was excluded from the experiment.

Table 5 shows the experimental results of Experiment 2, and Table 6 shows an example of the results of detecting dangerous objects. The overall F-measure is 0.872 for full HD, and 0.857 for 4k, indicating that the dangerous object detection method is capable of detecting dangerous objects on the road surface correctly for the most part. Comparing the F-measure by the resolutions, the difference between full HD and 4K was 0.015, which was not a large difference. However, the result of detection shows the tendency that 4k is capable of detecting dangerous objects on the road surface located at a remote distance compared with full HD.

Comparison of F-measure by types of dangerous objects shows that it is high for fallen leaves and manholes in the case of 4K, just as the same with full HD. Focusing on the dangerous object for which the F-measure for 4K is lower than that for full HD, examples of success and failure of gravel, bumps, and wet road surfaces are shown in Table 7. The result of detection indicates the tendency of failing to detect light-colored gravel, bumps where it was difficult to visually check the unevenness of the road surface, and the road surface where the wetted area is small. In the future, we plan to advance the method by increasing the learning data under diverse environments and by changing the version of YOLO.

Table 5. Experimental result of Experiment 2

Camera	Video resolution	YOLO	Dangerous objects	Fallen leave	Gravel	Manhole	Bump	Wet road surface	Total
THETA V	4K	v5	Total number	14	22	10	41	9	96
			Number of determination cases	14	15	9	29	5	72
			Number of correct answers	14	15	9	29	5	72
			Precision	1.000	1.000	1.000	1.000	1.000	1.000
			Recall	1.000	0.682	0.900	0.707	0.556	0.750
			F-measure	1.000	0.811	0.947	0.829	0.714	0.857

Table 6. Result of detecting dangerous objects with different resolutions

Video Resolution	Fallen leave	Gravel	Manhole	Bump	Wet road surface
Full HD					
4K					

3.4. Experiment 3: Verification of versatility for different versions of YOLO

In Experiment 2, the dangerous object detection model is generated for two types of versions: YOLOv5 and YOLOv8 to verify the versatility of the method in the case of different versions of YOLO.

Table 8 shows the experimental result of Experiment 3, and Table 9 shows an example of the results of detecting dangerous objects. The overall F-measure is 0.897 for YOLOv5 and 0.874 for YOLOv8, which indicate that the dangerous object detection method is capable of dangerous objects on the road surface for the most part. The detection result indicates the tendency that YOLOv8 is capable of detect dangerous objects on the road surface located at a distance better than YOLOv5. However, YOLOv8 sometimes detected a wet road surface erroneously as a manhole. The detection result shown in Table 10 suggests that its cause can be considered that the pattern of the wet road surface is similar to the manhole. In addition, focusing on the gravel for which the F-measure with YOLOv8 is lower than that with YOLOv5, it is made clear that the recall ratio of the gravel is low with YOLOv8, and that there are more failures in detection than other dangerous objects. Just as in Experience 2, its cause can be considered the difference in color of the gravel. Since this occurs due to differences in weather conditions as shown in Table 11, we plan to advance the method by increase the learning data under diverse environments in the future.

The results of experiments 1 to 3 indicate that there is little difference among the overall F-measure when applying the dangerous object detection method to the video images shot with 360-degree cameras under different conditions as to the types of cameras, resolutions, and YOLO versions, which proves the versatility of the proposed dangerous object detection method. Furthermore, assuming its utilization in the actual sites on different dates or under different weather conditions, the fact that the detection accuracy was equal to or higher than 0.85 in different dates and under different weather conditions in this study indicates that this dangerous object detection method is useful.

Table 7. Examples of success and failure for the results of detecting dangerous objects

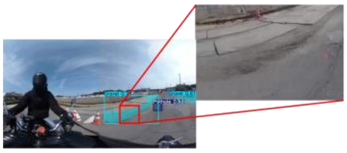
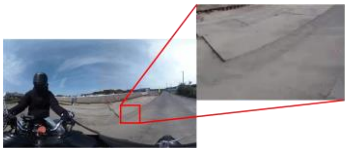
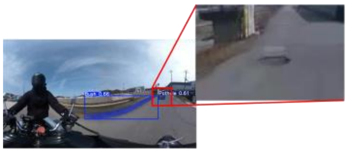
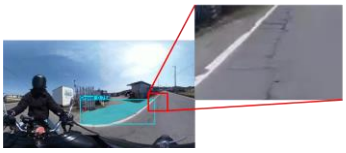
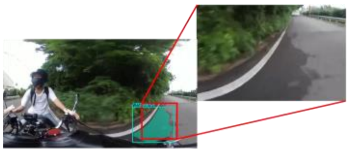
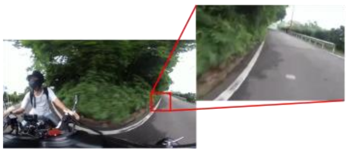
Dangerous objects	Success	Failure
Gravel		
Bump		
Wet road surface		

Table 8. Experimental result of Experiment 3

Camera	Video resolution	YOLO	Dangerous objects	Fallen leave	Gravel	Manhole	Bump	Wet road surface	Total
THETA SC	Full HD	v8	Total number	20	24	11	49	11	115
			Number of determination cases	19	17	10	37	8	91
			Number of correct answers	19	17	10	36	8	90
			Precision	1.000	1.000	1.000	0.973	1.000	0.989
			Recall	0.950	0.708	0.909	0.735	0.727	0.783
			F-measure	0.974	0.829	0.952	0.837	0.842	0.874

Table 9. Result of detecting dangerous objects with different versions of YOLO

YOLO	Fallen leave	Gravel	Manhole	Bump	Wet road surface
v5					
v8					

Table 10. Example of erroneous detection of the wet road surface

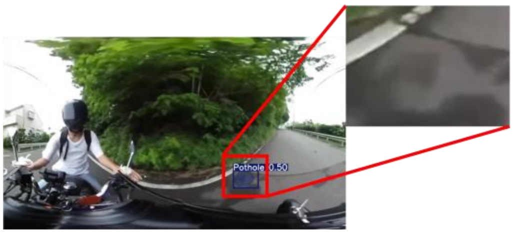
Correct	Error	Detection result with YOLOv8
Wet road surface	Manhole	

Table 11. Shot images of dangerous objects under different weather conditions

Weather	Fair Weather	Rain
Gravel		

4. Conclusion

In this study, we verified the versatility of the method to detect dangerous objects on the road surfaces including fallen leaves, gravel, manholes, bumps, and wet road surfaces. In the demonstration experiments, we applied the dangerous object detection method to the video images shot under respective conditions for the 360-degree camera (THETA SC and THETA V), resolutions (full HD and 4K), and YOLO versions (YOLOv5 and YOLOv8) to evaluate the precision ratio, recall ratio, and F-measure. As a result of demonstration experiments, it was found that there is little difference in the F-measure under different conditions such as types of cameras, resolution, and versions of YOLO, and consequently it is capable of detecting dangerous objects on the roads for the most part.

In the future, we plan to improve its accuracy by increasing the learning data under a variety of environments to deal with a problem of detection errors through repeated verification under different environments. We also aim to decrease the number of motorcycle accidents by detecting the factors leading to motorcycle accidents with additional information about the drivers.

Funding

This research received no external funding.

Conflicts of Interest

The authors declare no conflict of interest.

References

Inoue, H., et al. (2023). Research for detecting dangerous objects leading to overturning of motorcycles using deep learning. *Proceedings of the 85th National Convention of IPSJ*, *85*(1), 965–966.

Inoue, H., et al. (2024). Research for Detecting Dangerous Leading to Motorcycle Accident Using 360 Degree Camera. *Proceedings of the 86th National Convention of IPSJ*, *86*(1), 2-351–2-352.

Nikkei xTECH, Hitachi Astemos aims to commercialize ADAS for motorcycles by 2028. (2024). https://xtech.nikkei.com/atcl/nxt/column/18/02594/102800022/

Senoo, D., et al. (2017). Development of motor-and-bicycle anti roll-down system. *The Proceedings of the Transportation and Logistics Conference*, *26*. https://doi.org/10.1299/jsmetld.2017.26.1104

Tokyo Metropolitan Police Department. (2024). Motorcycle traffic fatality statistics (through 2023). https://www.keishicho.metro.tokyo.lg.jp/kotsu/jikoboshi/nirinsha/2rin_jiko.html

Digital Life

Journal of Digital Life 2025,5,S2

Special Issue: Measurement, control, and analysis of motion using ICT and AI

Article

Wildlife Approach Detection Using a Custom-Built Multimodal IoT Camera System with Environmental Sound Analysis

Ryo Tochimoto [1], Katsunori Oyama [2], Kazuki Nakamura [2]

[1] Graduate School of Computer Science, Nihon University, 1 Nakagawara, Tamuramachi Tokusada, Koriyama, Fukushima 963-8642, Japan

[2] Department of Computer Science, College of Engineering, Nihon University, 1 Nakagawara, Tamuramachi Tokusada, Koriyama, Fukushima 963-8642, Japan

Abstract

This paper presents a custom-built IoT camera system designed for recognizing wild animal approaches, where data transmission and power consumption are critical concerns in resource-constrained outdoor settings. The proposed method involves the spectral analysis on both infrared and environmental sound data before uploading images and videos to the remote server. Experiments, including battery endurance tests and wildlife monitoring, were conducted to validate the system. These results showed that the system minimized false positives caused by environmental factors such as wind or vegetation movement. Importantly, adding frequency features from audio waveforms that capture sounds including wind noise and footsteps led to an improvement in detection accuracy, which increased the AUC from 0.894 to 0.990 in Random Forest (RF) and from 0.900 with infrared sensor data alone to 0.987 in Logistic Regression (LR). These findings contribute to applications in wildlife conservation, agricultural protection, and ecosystem monitoring.

Keywords: Wildlife approach detection; Environmental sound analysis; Low-power IoT systems.

1. Introduction

Crop damage caused by wildlife remains a serious social issue, as it leads to certain vulnerable species becoming un-cultivable (Ministry of Agriculture, Forestry and Fisheries, 2023). Monitoring animal behavior is a critical first step in controlling wildlife pests; however, tracking free-roaming animals such as wild boars within a camera's field of view is inherently challenging. Fixed cameras equipped with human detection sensors in the real-world outdoor settings often experience false detections due to noise generated by swaying vegetation, resulting in an excessive number of unnecessary images being uploaded to a cloud server. Additionally, battery exhaustion is a persistent issue when installing cameras in remote, mountainous areas where securing a power source is impractical. Even when animals are captured within the camera’s field of view, it is often challenging to interpret situations involving an approaching animal based solely on images. Proper interpretation and appropriate actions require considering the behavioral and environmental contexts with those the detection results (Chang et al., 2009; Wu et al., 2023). Environmental contexts can include various audible events such as footsteps, wind, and vegetation movement. These environmental contexts have the potential to indirectly identify animal presence or movement.

Most wildlife monitoring systems adopt either bioacoustic monitoring or image processing techniques. Bioacoustic monitoring is effective for detecting animals through vocalizations, enabling the monitoring of species such as frogs and deer (McLoughlin et al., 2019; Lostanlen et al., 2019). However, this approach assumes that animals vocalize and that environmental noise is minimal. On the other hand, YOLO (You Only Look Once) models are widely used for their high accuracy and real-time performance in image processing (Li et al., 2023; Ma et al., 2024). Yet, both methods

Received: 22 November 2024, Revised: 18 January 2025, Accepted: 26 January 2025, Published: 26 March 2025

* Correspondence: oyama.katsunori@nihon-u.ac.jp

Publisher’s Note: JOURNAL OF DIGITAL LIFE. stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.

face challenges in terms of energy consumption and data transmission when implemented on resource-constrained devices such as the Raspberry Pi.

Existing wildlife monitoring solutions include commercial trail cameras from SPYPOINT and Moultrie. These products often employ AI-based filters to mitigate false triggers and can upload images via Wi-Fi or cellular networks. However, they primarily rely on PIR sensors and image-based analysis before uploading images, which may lead to increased false positives in windy or densely vegetated environments (SPYPOINT, n.d.; Moultrie Mobile, n.d.). In contrast, our approach integrates both infrared and audio frequency features (e.g., wind noise, footsteps) to reduce false positives more effectively, especially under challenging outdoor conditions.

Our study is among the first to indicate that combining infrared and audio data efficiently improves detection accuracy, which not only conserves energy and reduces data transmission but also extends operational longevity in remote wildlife monitoring applications. We have been developing custom-built IoT camera systems based on Raspberry Pi Zero 2 by integrating infrared and audio sensors for monitoring animal movements (Tochimoto et al., 2023). Unlike existing systems, our multimodal approach uniquely employs spectral analysis on both infrared and environmental sound data collected from the surroundings to minimize false positives caused by environmental factors such as wind or vegetation movement. This study primarily targets medium to large animals, such as wild boars, deer, monkeys, and raccoons, as these species are known to cause significant damage to crops.

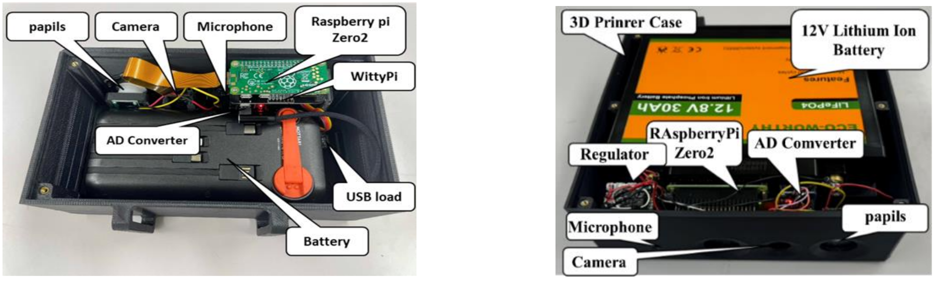
This paper presents the experimental evaluations of our multimodal IoT camera system, conducted at two distinct locations in Japan. In 2023, we tested the system in an open space near a residential area to assess its performance under moderate environmental conditions by focusing on the system's ability to reduce false detections in a relatively controlled setting. Subsequently, from May to July 2024, we deployed the system in a mountainous region of Katsurao Village, Fukushima, Japan, where the environment introduced challenges such as dense vegetation and variable weather conditions.

2. Multimodal IoT Camera System

2.1. Hardware Design and Implementation

The IoT camera system developed in this study is based on the Raspberry Pi Zero 2 to integrate various sensors such as an infrared sensor, infrared camera, Raspberry Pi camera, and audio microphone. Two types of the multimodal camera systems were designed for this research: The first model, **Version 1 (Ver 1)**, uses a small mobile battery paired with a solar panel, while the second model, **Version 2 (Ver 2)**, employs a 12 V lithium-ion battery to support extended continuous operation. In Ver 1, the system features a solar panel that recharges the battery using sunlight. Ver 2, on the other hand, has a larger battery capacity than Ver 1 and is also equipped with an infrared camera to enable nighttime detection. Both models are housed in custom-designed cases created with a 3D printer, which encase all sensors and batteries. The front of the case includes an infrared sensor, a camera, and an audio microphone. Each case is designed to be highly waterproof, with functionality verified through high-pressure shower tests, as shown in Figure 1.

The infrared sensor used in this study is a pyroelectric infrared sensor, PaPIRs (manufactured by Panasonic, long-distance detection type. This sensor is a long-distance detection type with a 12-m range, providing an analog output that enables the collection of time-series data from the surrounding environment for advanced time-series analysis. Our preliminary tests indicate that beyond 12 m, false alarms rise significantly due to environmental factors, which informed our decision to limit the range for improved accuracy and power conservation (Tochimoto et al., 2023). Detection ranges of commercial trail cameras can exceed 15 m, potentially increasing the risk of false positives in dense vegetation. In contrast, we selected a 12 m effective detection radius to balance sensitivity and battery efficiency.



(a) (b)

Figure 1: Two Types of Multimodal IoT Camera Systems: (a) the Version 1 (Ver 1) with a Mobile Battery and (b) the Version 2 (Ver 2) with an Extended Battery for Longer Operation.

In addition, an omnidirectional audio microphone, as illustrated in Figure 2, captures environmental audio waveforms, including sounds from rain, wind, and animal vocalizations. Processing this time-series audio data allows the system to classify diverse environmental sounds, during interpreting motion detected by the infrared sensor.

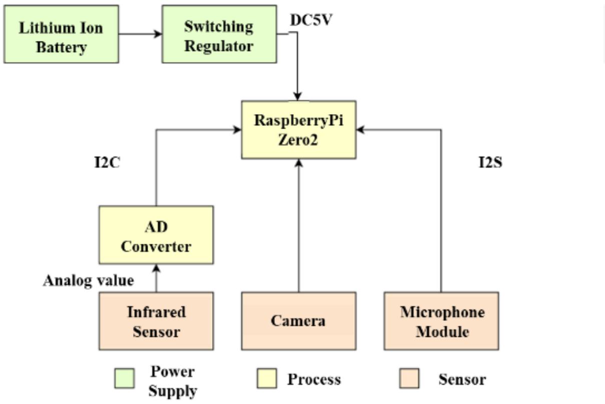


Figure 2: Block Diagram of Ver 2 IoT Camera System

2.2. Software Design and Implementation

This system requires continuous acquisition of infrared data while simultaneously operating the camera and microphone. Multithreading is employed for enabling the concurrent operation of various sensors. Every 0.1 seconds, the system retrieves values from the infrared sensor. As shown in Figure 3 of the system's sequence diagram, if a reaction is detected, it initiates both the photo capture and audio recording processes as subprocesses to collect data concurrently, while infrared data collection remains uninterrupted. Additionally, creating a cron job on the Raspberry Pi Zero 2, which is a resident program for executing scripts automatically, enables the device to start sensing immediately upon power-on.

The transmission of image data is the heaviest load on server communication. Therefore, to minimize unnecessary image transmissions and improve efficiency in resource-limited environments, such as mountainous areas, the system uses machine learning models locally to assess the likelihood of animal detection before transmission. Only when there is a high probability of detecting an animal does the system transmit relevant data to the remote server.

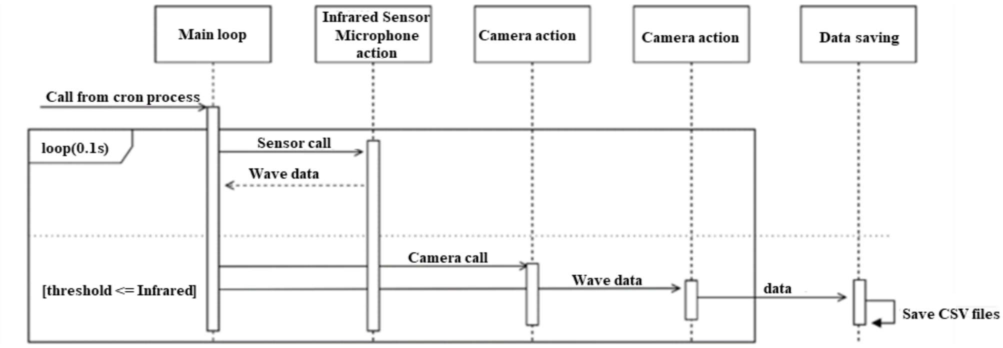


Figure 3: System Sequence for Multithreading Process

3. Methods

3.1. Operation Testing

The multimodal IoT camera system continuously monitors infrared waveforms. The system records images, videos, and audio data to log detected events when the detection threshold is exceeded. The collected infrared waveform data is stored in CSV format hourly, and data retrieval is performed remotely via a 4th Generation Mobile Communication System (4G) connection. Figure 4 illustrates the test setup. A power supply with a solar panel and a 4G router is placed at the center, with the Camera 1 and the Camera 2 representing the IoT camera systems developed and installed for this study. Each IoT camera system is connected via Wi-Fi and can be placed anywhere within a Wi-Fi range of approximately 50 m.

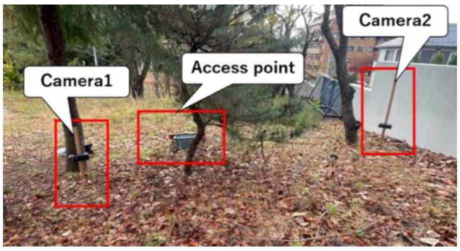
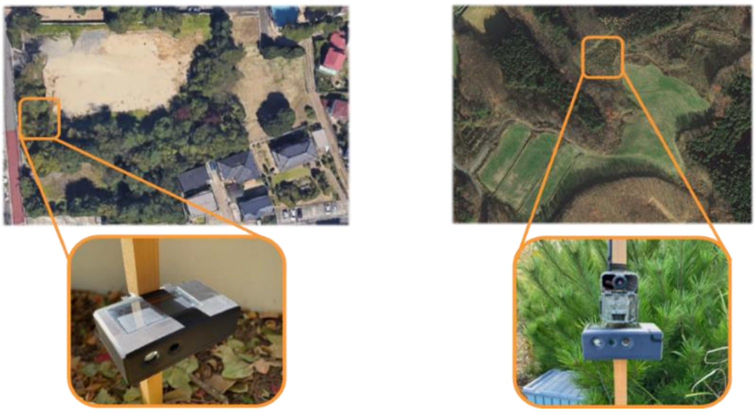


Figure 4: Operation Testing



(a) Open space near a residential area (b) Mountainous area

Figure 5: Test Sites

3.2. Test Sites and Measured Data

Data collection for this study was conducted at two test sites: (a) an open space near a residential area in April 2023 and (b) a mountainous area in Katsurao Village, Fukushima, Japan, from May to July 2024. Two IoT cameras were installed at each site, with the sensing areas marked as squares in the figures. At both locations, the ground was covered with leaves and grass, and vegetation was within the cameras' field of view. Conducting experiments at two different locations not only provided diverse measured data for model training and testing but also strengthened the reliability of the findings by validating the system under the practical environmental conditions.

Figure 6 shows an example of the infrared and audio waveform data collected per window time. To prepare the data for analysis, the following processing steps were performed. Noise artifacts may occur depending on the recording's start time (as shown in Figure 6), and these can affect the Fourier Transform (FFT) results. Therefore, as part of the audio data preprocessing, the first 0.3 seconds from the start of the recording were removed. Additionally, as preprocessing for the infrared data, a 10-second window was extracted from the point of detection. Next, FFT was applied to the time-series data of the infrared and audio waveforms. By converting these waveforms to the frequency domain for each time window, characteristic features were extracted. Subsequently, the sum and variance of the infrared and audio waveform FFT results were calculated, and labels for successful detection ("Approaching") or failed detection ("No Approaching") were added based on the footage captured during detection, creating the dataset for analysis.

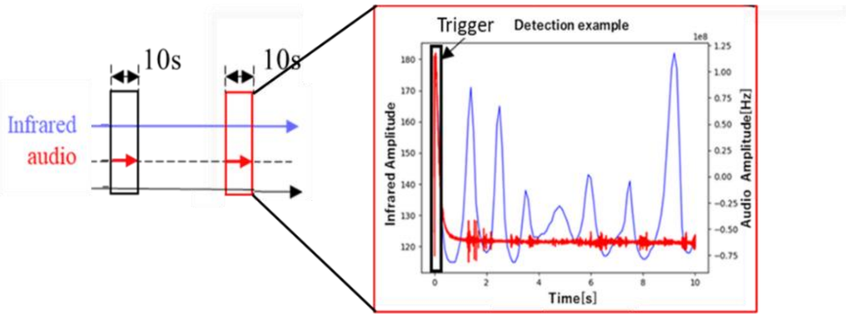
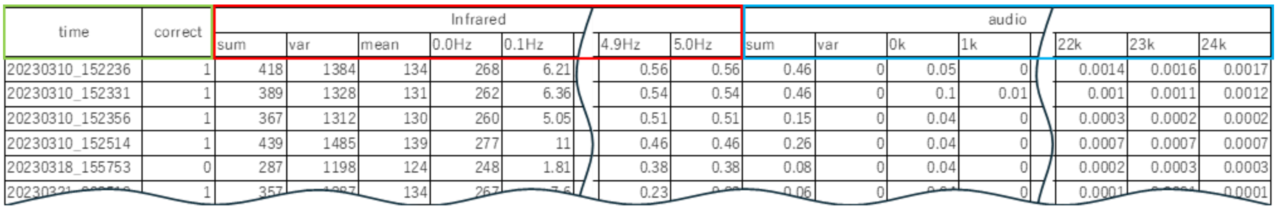


Figure 6: Examples of infrared and audio waveform data collected within a single time window

As shown in Table 1, the first set of columns in the dataset includes the filename, which contains the timestamp and the ground truth label. The subsequent columns (highlighted in red and blue) represent frequency features extracted from the infrared and audio waveforms. The column labeled "sum" indicates the sum of the components within the active frequency bands, "var" represents the variance of those components within the 10-second window, and "mean" represents their mean value. The "0.0 Hz" column indicates the spectral power of the frequency band between 0 Hz and 0.1 Hz.

Table 1: Example Dataset with Frequency Features from Infrared and Audio Waveforms



time	correct	Infrared								audio							
		sum	var	mean	0.0Hz	0.1Hz		4.9Hz	5.0Hz	sum	var	0k	1k		22k	23k	24k
20230310_152236	1	418	1384	134	268	6.21		0.56	0.56	0.46	0	0.05	0		0.0014	0.0016	0.0017
20230310_152331	1	389	1328	131	262	6.36		0.54	0.54	0.46	0	0.1	0.01		0.001	0.0011	0.0012
20230310_152356	1	367	1312	130	260	5.05		0.51	0.51	0.15	0	0.04	0		0.0003	0.0002	0.0002
20230310_152514	1	439	1485	139	277	11		0.46	0.46	0.26	0	0.04	0		0.0007	0.0007	0.0007
20230318_155753	0	287	1198	124	248	1.81		0.38	0.38	0.08	0	0.04	0		0.0002	0.0003	0.0003
[illegible]	1	357	[illegible]	134	267	[illegible]		0.23	[illegible]	0.06	0	[illegible]	0		0.0001	[illegible]	0.0001

3.3. PCA Method

To extract spectral features in the frequency domain, a dataset was generated where the spectral power for each frequency band served as a feature. The infrared waveform was segmented into units of either 0.1 Hz or 1 Hz, while the audio waveform was divided into units of 1 kHz, 2 kHz, and 4 kHz to assess their impact on detection accuracy. This segmentation, however, resulted in high-dimensional data. As illustrated in Table 2, six datasets (Dataset1 through Dataset6) were created based on these configurations. To manage the high dimensionality, Principal Component Analysis (PCA) was employed for dimensionality reduction, and detection accuracy was evaluated both with and without PCA.

Table 2: Column Specifications for Each Dataset Based on Infrared and Audio Waveforms

Dataset	Infrared Waveform	Audio Waveform	Infrared Columns	Audio Columns
Dataset1	0.1 Hz	1 kHz	54	27
Dataset2	0.1 Hz	2 kHz	54	14
Dataset3	0.1 Hz	4 kHz	54	8
Dataset4	1 Hz	1 kHz	10	27
Dataset5	1 Hz	2 kHz	10	14
Dataset6	1 Hz	4 kHz	10	8

3.4. Machine Learning Method

Two machine learning models, Random Forest (RF) and Logistic Regression (LR), were evaluated to compare the performance of the system using infrared and audio waveform data. Additionally, two approaches were tested to maximize model accuracy: one applied dimensionality reduction via Principal Component Analysis (PCA) to extract key features, while the other used all features without PCA.

In the PCA-applied approach, the reduced-dimension data were input into the model. In contrast, in the non-PCA approach, all features were used directly without dimensionality reduction. This comparison aimed to evaluate improvements in training efficiency achieved through dimensionality reduction and to analyze performance differences between models using all features and those using reduced features.

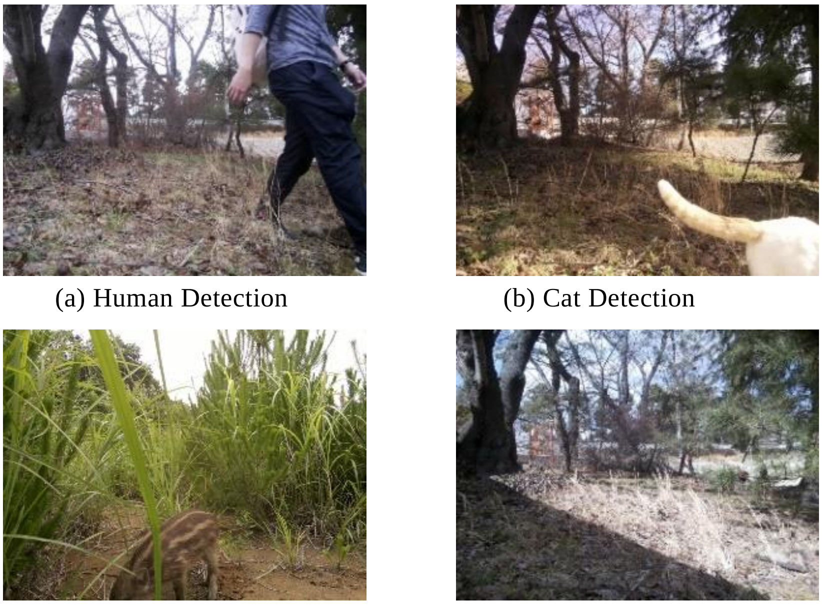
To address class imbalances in the training data, the Synthetic Minority Over-sampling Technique (SMOTE) was applied, and a stratified 5-fold cross-validation was performed. This approach enables a balanced training dataset by reducing the effects of class imbalance during training. The test data were used without additional processing, and model performance was assessed using the Area Under the Curve (AUC) metric.

4. Results

4.1. Operation Verification Results

Table 3 shows the predicted and actual operating days of the system. In this experiment, two types of systems were tested: The Ver 1 as the mobile battery model and the Ver 2 as the lithium-ion battery model, with the operating time of each system measured.

In the Ver 1, power consumption was approximately 1 W, with a battery capacity of 144 Wh. Although the predicted operating time was 144 hours, the result showed the system operated continuously for 168 hours, which is likely extended by the solar panel recharging the battery. In the Ver 2, with a battery capacity of 360 Wh (12 V) and a power consumption of 1 W, the estimated operating time was 360 hours. The actual operating time was 336 hours, approximately two weeks, and was close to the expected result. Both models demonstrated that power consumption and environmental conditions influenced operating time, with the Ver 2 proving more suitable for stable, long-term operation.



(a) Human Detection

(b) Cat Detection

(c) Wild Boar Detection

(d) Swaying Vegetation

Figure 7: Images Captured During Detection Events

Table 3: Operational Records and Detection Counts

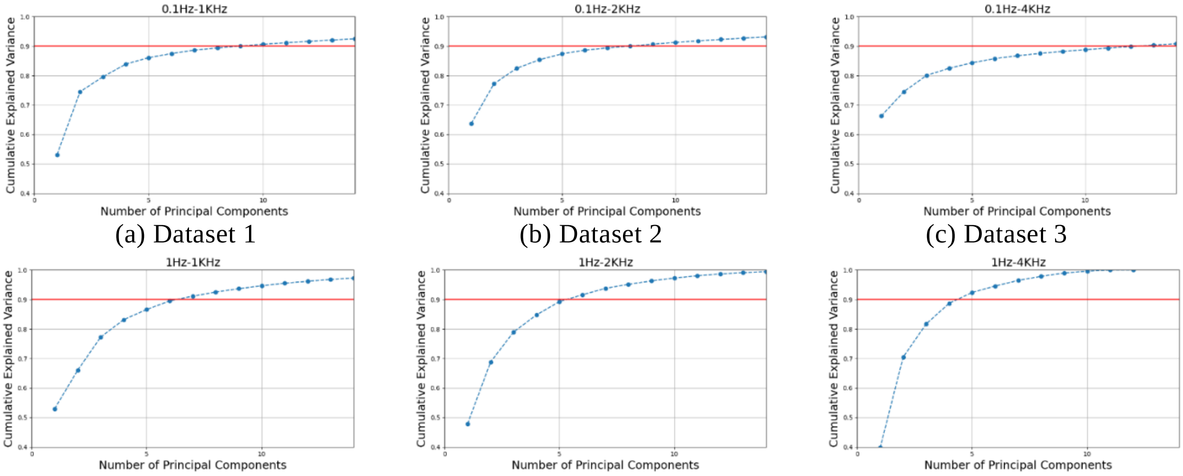
Version	estimated operation time	Number of working hour	Number of detections
Ver 1	168	168	59
Ver 2	360	336	66

4.2. Data Collection Results

At the first test site, 267 data instances were collected, including 13 instances with animals and 254 without. At the second test site in Katsurao Village, 2,087 data instances were collected, with 109 containing animals and 1,978 without. The images shown in Figure 7 below provide examples of actual camera captures. At the first site, data included images of humans and cats, while at the second test site, images included humans, dogs, raccoons, and wild boars. When no animals were present, the cameras occasionally recorded empty scenes triggered mistakenly by swaying vegetation due to wind.

4.3. PCA Results

To improve the performance of the machine learning model, feature selection was conducted to identify the most relevant columns (features). Given the high dimensionality of the current dataset, a Principal Component Analysis (PCA) was applied to reduce dimensionality and extract the most important features. For each dataset, principal components were selected until the cumulative contribution rate reached 90%, and the contribution of each feature to



(a) Dataset 1

(b) Dataset 2

(c) Dataset 3

(d) Dataset 4

(e) Dataset 5

(f) Dataset 6

Figure 8: Cumulative Contribution Rate

these components was evaluated. Figure 8 shows the cumulative contribution rates of PCA for each dataset, while Table 4 lists the top 10 contributing features in Principal Components 1 (PC1) to 5 (PC5) for each dataset. Based on the analysis of the contributing features and the video data, the PC1 likely corresponds to swaying vegetation, and the PC2 to environmental noise such as wind.

Table 4: Contribution Rate (Example from Dataset1)

PC1 (Swaying Vegetation)		PC2 (Environmental Noise, e.g., Wind)		PC3 (Animal Vocal Characteristics)		PC4 (Slow Movements of Animals)		PC5 Environmental Noise, e.g., Wind	
2.7 Hz	0.149	10 kHz	0.228	2 kHz	0.175	0.4Hz	0.267	14 kHz	0.208
2.5 Hz	0.148	9 kHz	0.227	1 kHz	0.174	0.6Hz	0.242	13 kHz	0.186
2.1 Hz	0.148	11 kHz	0.224	3 kHz	0.174	0.5Hz	0.235	15 kHz	0.175
2.2 Hz	0.148	7 kHz	0.223	0.4 Hz	0.170	0.9Hz	0.193	9 kHz	0.092
2.8 Hz	0.148	8 kHz	0.223	0.6 Hz	0.159	0.7Hz	0.187	11 kHz	0.081
3.0 Hz	0.148	6 kHz	0.221	4 kHz	0.148	0.1Hz	0.178	10 kHz	0.078
2.3 Hz	0.148	12 kHz	0.216	0.5 Hz	0.147	0.3Hz	0.169	12 kHz	0.074
2.4 Hz	0.148	5 kHz	0.216	0.3 Hz	0.145	23 kHz	0.168	16 kHz	0.063
3.1 Hz	0.148	16 kHz	0.213	6 kHz	0.142	20 kHz	0.162	4.7 Hz	0.059
3.2 Hz	0.147	13 kHz	0.213	7 kHz	0.138	24 kHz	0.160	4.6 Hz	0.057

For the infrared waveform, low-frequency components such as 2–3 Hz and 0.1–1 Hz made substantial contributions across multiple datasets, as they appear to reflect responses to animal approaches and environmental fluctuations. In the audio waveform, high and mid-frequency components, specifically in the 10–11 kHz and 1–3 kHz ranges, were also significant, and may correspond to noise and animal-related sounds in the audio signal. The variance of the audio waveform ("var" in the audio category, as shown in Table 1) contributed strongly across multiple datasets as a key indicator of variations in environmental and animal sounds.

4.4. Machine Learning Results

Table 5 shows the 5-fold stratified cross-validated AUC scores and the independent test AUC scores of the RF model with and without PCA. These cross-validated AUC scores are averaged over a stratified 5-fold procedure with SMOTE applied to the training split (80 %), while the test AUC scores represent performance on a separate hold-out set (20 %) without further oversampling. From the results, the combination of infrared 0.1 Hz and audio 4 kHz achieved the highest AUC score of 0.990. When PCA was applied, the combination of infrared 0.1 Hz and audio 1 kHz recorded a high AUC score of 0.986. These results indicate that for lower-dimensional data, high performance can be achieved even without using PCA.

Table 6 presents the AUC results of the LR model with and without PCA. When PCA was applied, the combination of infrared 0.1 Hz and audio 1 kHz achieved an AUC of 0.987 with PCA, which nearly matched the 0.985 AUC without PCA. This finding indicates a slight improvement in model accuracy when PCA is applied to higher-dimensional data.

The Random Forest (RF) model maintained high performance regardless of whether PCA was applied or not. It is noteworthy that the combination of infrared 0.1 Hz and audio 4 kHz achieved the highest AUC score of 0.990 without PCA. Furthermore, even with PCA applied, high AUC scores of 0.986 were recorded for the combinations of infrared 0.1 Hz and audio 1 kHz, as well as infrared 0.1 Hz and audio 2 kHz. These results indicate that the RF model can achieve sufficient performance without PCA, while PCA proves to be effective for high-dimensional data.

On the other hand, applying PCA to low-dimensional data was found to reduce performance. For instance, in the Logistic Regression (LR) model, the combination of infrared 1 Hz and audio 4 kHz achieved an AUC score of 0.950 without PCA, which dropped to 0.877 when PCA was applied. This demonstrates that PCA is not always effective in every scenario.

Table 5: Random Forest AUC with and without PCA

Infrared	Audio	PCA	Cross-Validated AUC (Mean:0.98)	Test AUC (Mean:0.96)
0.1 Hz	4 kHz	No	0.952	0.990
1 Hz	1 kHz	No	0.984	0.989
0.1 Hz	1 kHz	Yes	0.990	0.986
0.1 Hz	2 kHz	Yes	0.992	0.986
1 Hz	4 kHz	No	0.972	0.984
1 Hz	2 kHz	No	0.982	0.982
0.1 Hz	1 kHz	No	0.973	0.980
0.1 Hz	2 kHz	No	0.984	0.979
1 Hz	2 kHz	Yes	0.983	0.915
1 Hz	1 kHz	Yes	0.975	0.906
1 Hz	4 kHz	Yes	0.981	0.895
0.1 Hz	4 kHz	Yes	0.991	0.874

Table 6: Logistic Regression AUC with and without PCA

Infrared	Audio	PCA	Cross-Validated AUC (Mean:0.93)	Test AUC (Mean:0.94)
0.1Hz	1KHz	Yes	0.961	0.987
1Hz	1KHz	No	0.941	0.987
1Hz	2KHz	No	0.955	0.985
0.1Hz	2KHz	Yes	0.961	0.980
0.1Hz	1KHz	No	0.931	0.972
0.1Hz	2KHz	No	0.925	0.965
1Hz	4KHz	No	0.893	0.950
0.1Hz	4KHz	No	0.892	0.922
1Hz	1KHz	Yes	0.946	0.881
1Hz	2KHz	Yes	0.918	0.879
1Hz	4KHz	Yes	0.884	0.877
0.1Hz	4KHz	Yes	0.922	0.847

4.5. Variable Importance in Machine Learning Models

The important frequency bands were identified using the Random Forest (RF) and Logistic Regression (LR). As shown in Tables 7 and 8, the analysis focused on the data with the highest AUC values among those without PCA, specifically the combination of infrared 0.1 Hz and audio 1 kHz, to identify key frequency bands. Table 7 summarizes the feature importance from the RF, while Table 8 shows the coefficients from the LR.

In the RF results, the 5 kHz frequency band showed the highest importance (0.079), followed by 11 kHz, 13 kHz, and 10 kHz. These results indicate that high-frequency audio data plays a significant role in animal detection. In contrast, the importance of infrared data was relatively low, and high-frequency audio data contributes substantially to the model's predictions.

The RF model prioritized high-frequency audio bands (e.g., 5 kHz, 11 kHz, 13 kHz), and it suggests that audio data plays an important role in detecting animal vocalizations and environmental noise. The LR model, however, placed greater emphasis on low-frequency infrared data, such as 0.7 Hz and 1.7 Hz. This suggests that infrared data is more effective for identifying slow animal movements and environmental fluctuations.

Table 7: Random Forest Feature Importance

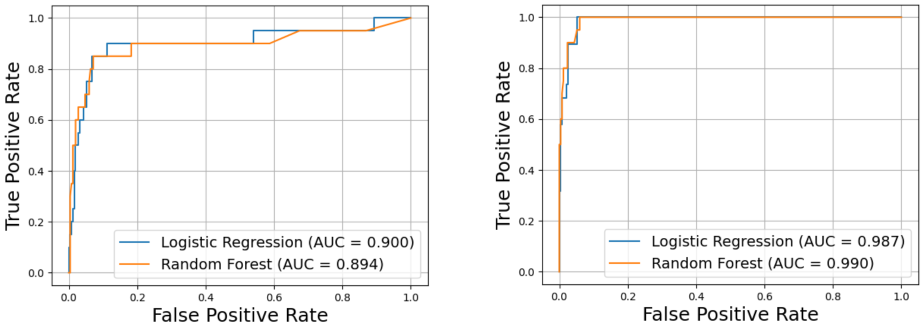
Feature	Importance	Data Type
5 kHz	0.079	Audio
11 kHz	0.047	Audio
13 kHz	0.045	Audio
10 kHz	0.042	Audio
8 kHz	0.040	Audio
9 kHz	0.040	Audio
12 kHz	0.039	Audio
7 kHz	0.038	Audio
infrared_mean	0.036	Infrared
infrared_var	0.033	Infrared
6 kHz	0.033	Audio
0.5 Hz	0.032	Infrared
audio_var	0.032	Infrared
0.4 Hz	0.031	Infrared
2 kHz	0.029	Audio
4 kHz	0.029	Audio
0.0 Hz	0.029	Infrared
audio_add	0.021	Audio
3 kHz	0.018	Audio
1 kHz	0.018	Audio

Table8: Logistic Regression Coefficients

Feature	Importance	Data Type
2 kHz	1.83	Audio
0.7 Hz	1.79	Infrared
1.7 Hz	1.60	Infrared
15 kHz	1.44	Audio
6 kHz	1.27	Audio
5.0 Hz	1.27	Infrared
13 kHz	1.18	Audio
0.4 Hz	1.10	Infrared
0.6 Hz	1.09	Infrared
4.2 Hz	1.08	Infrared
4.5 Hz	1.08	Infrared
0.5 Hz	1.03	Infrared
5 kHz	1.03	Audio
14 kHz	0.99	Audio
2.9 Hz	0.96	Infrared
10 kHz	0.80	Audio
3.5 Hz	0.80	Infrared
12 kHz	0.78	Audio
1.5 Hz	0.69	Infrared
2.4 Hz	0.67	Infrared

4.6. Influence of Audio Waveform

This study assessed the effect of integrating audio waveforms with infrared waveforms on the accuracy of animal approach detection. Models trained with only infrared waveforms were compared to those using both infrared and audio waveforms. As shown in Figure 9, AUC scores for the RF and the LR achieved 0.894 and 0.900, respectively, when using only infrared waveforms. These models using frequency features extracted from infrared waveforms alone achieved a reasonable level of accuracy, although the risk of false positives and missed detections remains. The inclusion of audio frequency features improved the AUC to 0.990 for the RF and 0.987 for the LR. These results are consistent with the cross-validated and test AUC values shown in Tables 5 and 6. From these findings, we confirm that integrating audio waveforms plays an essential role.

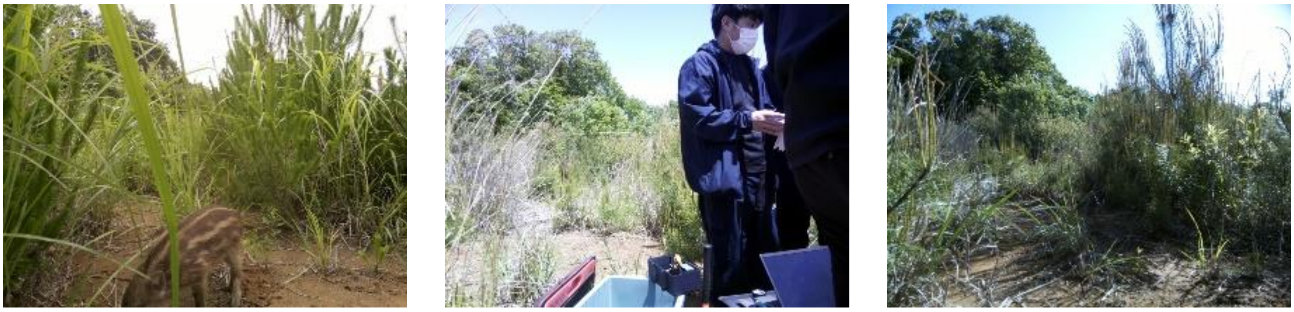


(a) Infrared Only

(b) Infrared + Audio

Figure 9: ROC Curves and AUC Comparison Between Infrared Only and Infrared with Audio

Table 9 summarizes selected training results for each model, while Figure 10 illustrates examples of detection under different conditions: **Case 1** (Wild Boar) represents a straightforward scenario where accurate detection was achieved with both infrared-only and infrared-plus-audio data. **Case 2** (Human) is the case of a false negative in the infrared-only approach, where minimal movement between frames caused the model to misclassify the instance as “no approaching.” The addition of frequency features extracted from audio waveforms allowed the detection of human voices, resulting in a correct classification of “approaching.” Lastly, **Case 3** (No Detection) is the case of a false positive with the infrared-only approach, where swaying vegetation was misclassified as “approaching.” The addition of frequency features from audio waveforms helped identify wind noise for the correct classification of “no approaching.”



(a) Case1: Wild Boar

(b) Case2: Human

(c) Case3: No Detection

Figure 10: Comparison of Detection Cases Under Various Conditions

Table 9: Comparison of Detection Results: Infrared Only vs. Infrared with Audio

Case	Ground Truth	LR_IR Only	RF_IR Only	LR_IR+Audio	RF_IR+Audio
Case1 (Wild Boar)	P	P	P	P	P
Case2 (Human)	P	N	N	P	P
Case3 (No Detection)	N	P	P	N	N

LR: Random Forest RF: Random Forest IR: Infrared

5. Discussion

This study proposed a multimodal detection system that integrates frequency features from infrared and audio waveforms. The analysis confirmed that frequency bands identified through FFT analysis are important for identifying animal movements and mitigating false detections. Infrared waveforms provided low-frequency components, such as 2–3 Hz (swaying vegetation) and 0.1–1 Hz (slow movements of animals or humans), which were particularly significant, as highlighted in Section 4.5.

Similarly, audio waveforms contributed mid-to-high frequency bands, such as 1–3 kHz (human voices or animal vocalizations) and 10–11 kHz (environmental noise like wind). These features, as demonstrated in Section 4.6, significantly improved the AUC for both the RF and LR models. The integration of audio data effectively reduced errors caused by environmental factors, such as vegetation sway.

Experiments in this study were conducted at two distinct test sites: a residential area and a mountainous region. These environments invited unique challenges such as varying vegetation density and weather conditions for validating the robustness and adaptability of the proposed system. By effectively reducing false detections through the integration of infrared and audio data, the proposed system shows considerable promise for real-world applications.

In this paper, we evaluated both Random Forest (RF)—a non-linear ensemble method—and Logistic Regression (LR)—a linear model—to compare their performance in our system. Based on the analysis in Section 4.4, RF consistently delivers robust, high performance on multimodal (infrared + audio) and higher-dimensional datasets across diverse frequency-segmentation settings. Meanwhile, LR can perform nearly as well, particularly when paired with effective PCA for high-dimensional data, though it is more sensitive to feature engineering. Accordingly, we recommend RF as an immediate, stable solution, whereas LR (and other linear models) may be preferable in scenarios where interpretability is a priority, especially when identifying which features contribute most to the detection results.

Nevertheless, this study has certain limitations. The system relies on camera-based data collection for ground truth labels, limiting detection to the camera’s field of view and introducing a risk of missed detections outside this range. Additionally, as noted in Section 3.1, the Raspberry Pi hardware brought challenges in power consumption, constraining operational time even with a larger battery. Overcoming these challenges will be essential for improving the system’s practicality and enabling its deployment in diverse real-world scenarios.

6. Conclusion and Future Challenges

This study proposed and evaluated a multimodal detection system that integrates frequency features from infrared and audio waveforms to improve the accuracy of animal approach detection. By applying FFT to these waveform types, the system identified critical frequency bands that contribute to the detection accuracy. Infrared waveforms captured low-frequency components, such as 0.1–1 Hz and 2–3 Hz, linked to animal movements and environmental fluctuations. Audio waveforms provided mid-to-high frequency features, such as 1–3 kHz and 10–11 kHz, capturing animal vocalizations, footsteps, and environmental noise. This integration proved highly effective by improving the AUC for RF and LR models to 0.990 and 0.987, respectively. From these results, we confirm that combining infrared and audio data is the key strategy for practical application of the multimodal IoT camera systems.

Future efforts should focus on improvement of robustness and reliability by adapting the system to diverse environmental conditions. Expanding the dataset to include more balanced samples representing various animal species is necessary for refining detection performance. Additionally, optimizing machine learning models for real-time processing and addressing energy efficiency will enable long-term deployment in resource-constrained outdoor environments. By addressing these challenges, the proposed system has the potential to significantly advance wildlife management, mitigate crop damage, and contribute to broader environmental conservation efforts.

Author Contributions

Conceptualization, K.O., and K.N.; methodology, K.O.; software, R.T.; validation, R.T., K.O., and K.O.; formal analysis, R.T.; investigation, R.T. and K.O.; data curation, R.T.; writing—original draft preparation, R.T.; writing—review and editing, K.O.; visualization, R.T.; supervision, K.O.; project administration, K.O. and K.N.

Conflicts of Interest

The authors declare no conflict of interest.

Acknowledgements

We gratefully acknowledge the Fukushima Innovation Coast Framework Promotion Project, which provided a strong foundation for collaboration and innovation in developing the IoT camera systems used in this research.

References:

Chang, C. K., Jiang, H., Ming, H., & Oyama, K. (2009). Situ: A situation-theoretic approach to context-aware service evolution. *IEEE Transactions on Services Computing, 2*(3), 261–275. https://doi.org/10.1109/TSC.2009.21

Li, S., Zhang, H., & Xu, F. (2023). Intelligent detection method for wildlife based on deep learning. *Sensors, 23*(19), 9669. https://doi.org/10.3390/s23249669

Lostanlen, V., Salamon, J., Farnsworth, A., Kelling, S., & Bello, J. P. (2019). Robust sound event detection in bioacoustic sensor networks. *PLOS ONE, 14*(10), e0214168. https://doi.org/10.1371/journal.pone.0214168

Ma, Z., Dong, Y., Xia, Y., Xu, D., Xu, F., & Chen, F. (2024). Wildlife real-time detection in complex forest scenes based on YOLOv5s deep learning network. *Remote Sensing, 16*(8), 1350. https://doi.org/10.3390/rs16081350

McLoughlin, M. P., Stewart, R., & McElligott, A. G. (2019). Automated bioacoustics: Methods in ecology and conservation and their potential for animal welfare monitoring. *Journal of the Royal Society Interface, 16*(150), 20190225. https://doi.org/10.1098/rsif.2019.0225

Ministry of Agriculture, Forestry and Fisheries. Summary of the Annual Report on Food, Agriculture and Rural Areas in Japan (FY2023). https://www.maff.go.jp/e/data/publish/Annual_Report/AnnualReportonFoodAgricultureandRuralAreas_FY2023.pdf

Moultrie Mobile. (n.d.). Why am I getting so many pictures or pictures with nothing in them? https://support.moultriemobile.com/hc/en-us/articles/1500006416442-Why-am-I-getting-so-many-pictures-or-pictures-with-nothing-in-them

SPYPOINT. (n.d.). Best practices for resolving false triggers. https://spypoint.onsitesupport.io/knowledge-base/article/best-practices-for-resolving-false-triggers

Tochimoto, R., Oyama, K., & Ming, H. (2023). Development of an IoT camera system for situation recognition of approaching animals. In *Proceedings of the IEEE International Conference on Software Services Engineering (SSE)* (pp. 1–6). https://doi.org/10.1109/SSE60056.2023.00047

Wu, J., Feng, Y., & Chang, C. K. (2023). Sound of daily living identification based on hierarchical situation audition. *Sensors, 23*(7), 3726. https://doi.org/10.3390/s23073726

Digital Life

Journal of Digital Life 2025,5,S3

Special Issue: Measurement, Control, and Analysis of Motion Using ICT and AI
Article

Research on Indoor Self-Location Estimation Technique Using Similar Image Retrieval Considering Environmental Changes

Masaya Nakahara [1], Yoshinori Tsukada [2], Yoshimasa Umehara [3] and Shota Yamashita [4]

[1] Faculty of Information Science and Arts, Osaka Electro-Communication University, 1130-70 Kiyotaki, Shijonawate-shi, Osaka 575-0063, Japan
[2] Faculty of Engineering, Reitaku University, 2-2-1 Hikarigaoka, Kashiwa-shi, Chiba 277-8686, Japan
[3] Faculty of Business Administration, Setsunan University, 17-8 Ikedanakamachi, Neyagawa-shi, Osaka 572-8508, Japan
[4] Graduate School of Information Science and Arts, Osaka Electro-Communication University, 1130-70 Kiyotaki, Shijonawate-shi, Osaka 575-0063, Japan

Abstract
In Japan, the shortage of human resources due to the declining birthrate and aging population is becoming a social problem. Particularly in the security industry, the irregular working hours and associated risks are making it increasingly challenging to secure workers. This has led to a rise in use of security systems that utilize security cameras and drones. However, in factories and other buildings with a lot of equipment and intricate structures, there is the problem of blind spots caused by occlusion. This situation necessitates the use of automated drone patrols, and a problem arises when self-position estimation fails in areas where acquiring feature points is difficult, such as corridors. To solve these problems, in a previous study, we devised a technique for position estimation using a method that can calculate similarity based on changes in the distribution of color information across the entire image. In this study, we propose a method that can cope with environmental changes caused by object movement while combining feature point-based methods.

Keywords: *automated patrol, drone, position estimation, image search*

1. Introduction

In Japan, the shortage of human resources has become a social problem due to the declining birthrate and aging population. This has had a serious effect on the security industry because of the irregular hours of work, the danger involved in responding to suspicious persons, and the large number of personnel required to patrol large facilities (Ministry of Health, Labour and Welfare, 2024). These factors necessitate the development of security systems that utilize security cameras and drones as a solution to the shortage of human resources. However, security systems that utilize security cameras face several challenges. For example, when monitoring areas with many pieces of equipment and intricate locations, such as factories, there are concerns that the number of cameras installed will increase and blind spots will occur due to the effects of occlusion. Therefore, it is expected that drones and robots that can move autonomously can mount and move cameras to patrol and monitor these areas, thereby reducing the number of personnel required for security.

For example, Skydio 2+ is a drone that can fly autonomously using camera images. It uses Visual SLAM to estimate its own position with high accuracy even in non-GNSS space, based on the images from multiple cameras installed on the drone. This system enables safe navigation in narrow, complex structures with steel or concrete frames under bridges and in wide-area shooting. As methods for estimating self-position using camera images in fields other than drones, “A method for estimating self-position by feature point matching” (Okamoto et al., 2012; Yamazaki et al.,

Received: 21 November 2024, Revised: 20 January 2025, Accepted: 15 February 2025, Published: 26 March 2025
* Correspondence: nakahara@oecu.jp

2019) and “A method for feature point matching on a 3D map generated from RGBD camera shots” (Matsumoto et al., 2024; Yang et al., 2020) have been proposed. The former calculates features from a set of previously captured images and compares them with features obtained from the captured images to estimate the location of the captured images. In “A method for feature point matching on a 3D map generated from RGBD camera shots,” a 3D map is created using an RGBD camera to obtain the positional relationships of characteristic structures and objects in a building. Then, based on the created 3D map, the positions where the features match the input image are searched for using deep learning and other methods.

All existing methods estimate self-location under the assumption of many common features in the video, regardless of the time of year. Therefore, when targeting narrow indoor areas with many plain walls, such as those patrolled by security drones, there is “the problem of failing to estimate self-position due to the small number of features” and “the problem of failing to estimate self-position when the feature object itself is moving” in places, such as factories. For example, in the case of Skydio 2+, when flying over a complex structure or a wide area, one of the cameras will always reflect a feature object or landmark location, enabling highly accurate self-position estimation. However, in the case of an indoor hallway between plain walls, the distance from the camera to the plain walls on both sides of the subject is short, and it may be difficult for each camera to always have sufficient features for self-position estimation during flight. Conversely, in “A method for estimating self-position by feature point matching” and “A method for feature point matching on a 3D map generated from RGBD camera shots,” position estimation is based on previously obtained features. Therefore, when applied to locations such as factories, where objects such as tools and instruments are easily moved, the number of commonly obtained features decreases, and a completely different position may be misestimated as the self-position.

In our previous study (Yamashita et al., 2024), we proposed a method using not only a feature point-based approach but also dHash (Figure 1), an algorithm that searches for overlapping images based on changes in the distribution of color information across the entire image. The dHash method calculates a hash value based on the distribution of color changes in the entire image using the luminance gradient of each segmented image area in relation to adjacent areas. Using this algorithm, hash values similar to those in daylight can be calculated based on small differences in luminance even in environments with insufficient light. Therefore, it is highly possible to generate hash values that approximate the nearest pre-captured image even at nighttime, if the features within the shooting range are visible. This method can be used to capture color changes common to images with location information that have been previously captured and images used for self-location estimation, even with few obtained features. In addition, because dHash utilizes information from the entire image, it effectively suppresses the effects of changes in local features caused by object movement better than feature point-based methods. For example, in the case of a hallway, opening and closing doors may cause environmental changes. Therefore, the method can address existing methods problems, such as those of “failing to estimate self-position due to the small number of features” and of “failing to estimate self-position when the feature object itself is moving.” However, demonstration experiments showed that the estimation results are prone to errors on straight sections. However, while the drone used for automatic patrol does not need to change the direction of travel in the straight sections, it needs to change the direction of travel significantly in the straight sections near the curve points. Therefore, the number of images taken in advance must be denser when the drone is close to a curve, and more accurate position estimation is required than in existing studies. In the existing method, no feature change occurs between the images taken before and after the straight section except for the distance from the wall at the end of the curve point, and it is said that there is little difference in the similarity in the straight section near the curve point (Figure 2).

In this study, we propose a method that selects multiple candidate images similar to the input image using scale-invariant feature transform (SIFT) features and then estimates similar images using dHash among them. This method is expected to improve the estimation results for straight sections by considering the distribution of local features influenced by columns, windows, and other factors.

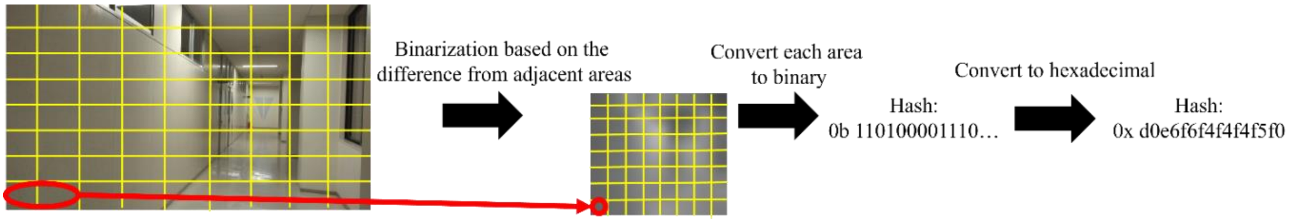


Figure 1. Processing steps of dHash

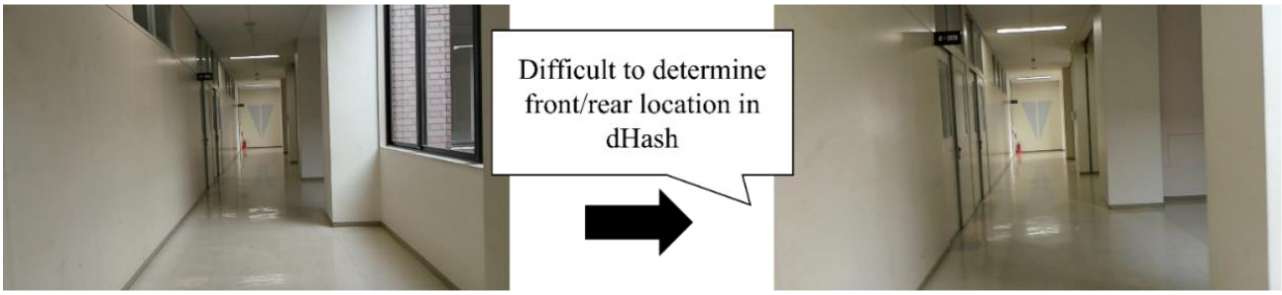


Figure 2. Examples of straight sections that are prone to estimation failures

2. Methods

2.1 Overview of Methodology

Based on the issues discussed in Section 1, we propose a self-positioning estimation technique that considers the similarity of images taken in straight sections, such as corridors, indoors, and in factories, where security drones target many plain wall surfaces. Figure 3 shows the process flow of the proposed method, which consists of the “Candidate Image Selection Function,” “Similar Image Retrieval Function,” and “Location Estimation Function.” The input data of the proposed method consists of “camera images for location estimation,” “images and location information on the patrol route taken in advance,” and “a map of the patrol route composed of point cloud data.” The output data is the “coordinates of the estimated location,” which can be displayed on a map. In this case, the images on the travel route in the input data are stored with the coordinate values on the map of the travel route corresponding to the shooting position of each image in advance (Figure 4). In this method, images on the patrol route are collected at regular intervals in straight sections, while images are collected at denser intervals in curve sections. This is because the estimated position acquired by this method is used to provide movement instructions to the drone, so it is necessary to collect images at high density in the curve section.

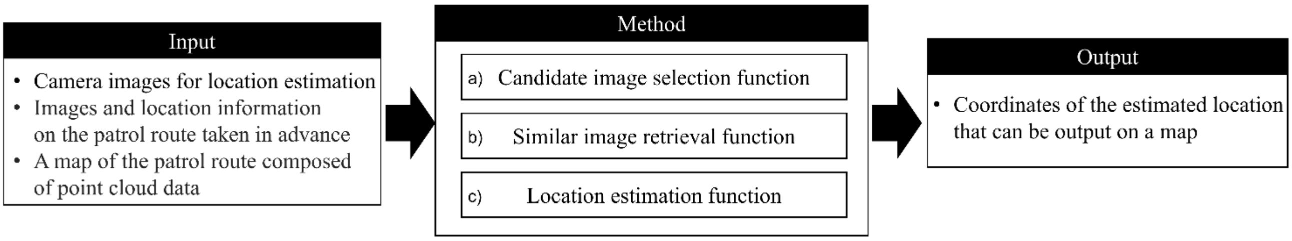

Input
• Camera images for location estimation
• Images and location information on the patrol route taken in advance
• A map of the patrol route composed of point cloud data
Method
a) Candidate image selection function
b) Similar image retrieval function
c) Location estimation function
Output
• Coordinates of the estimated location that can be output on a map

Figure 3. Flow of the proposed method

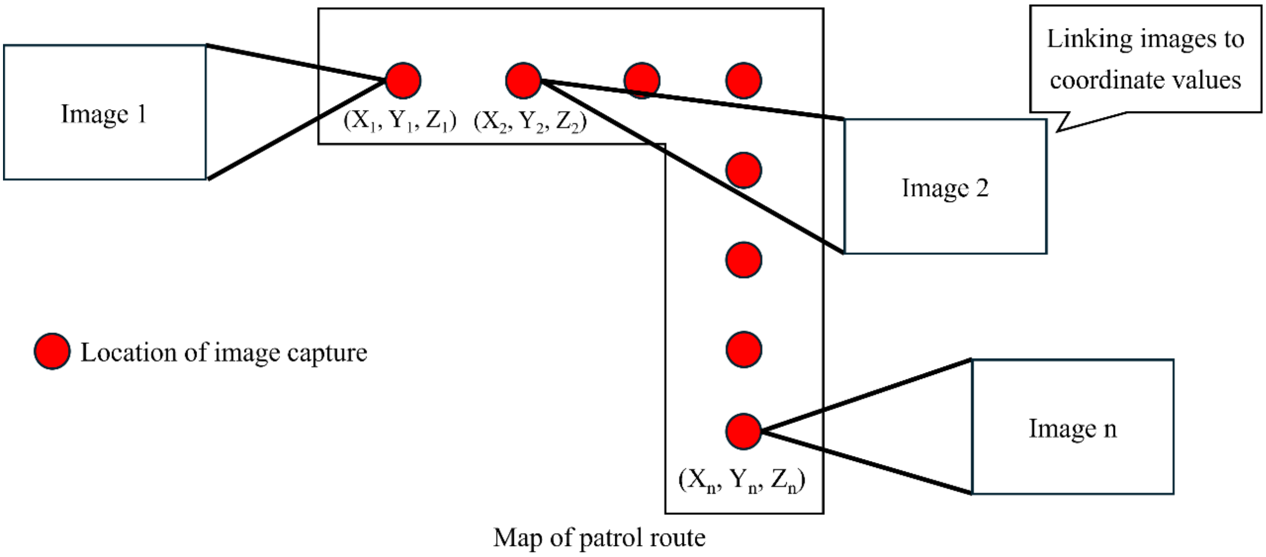


Figure 4. Image diagram of input data

2.2 Candidate Image Selection Function

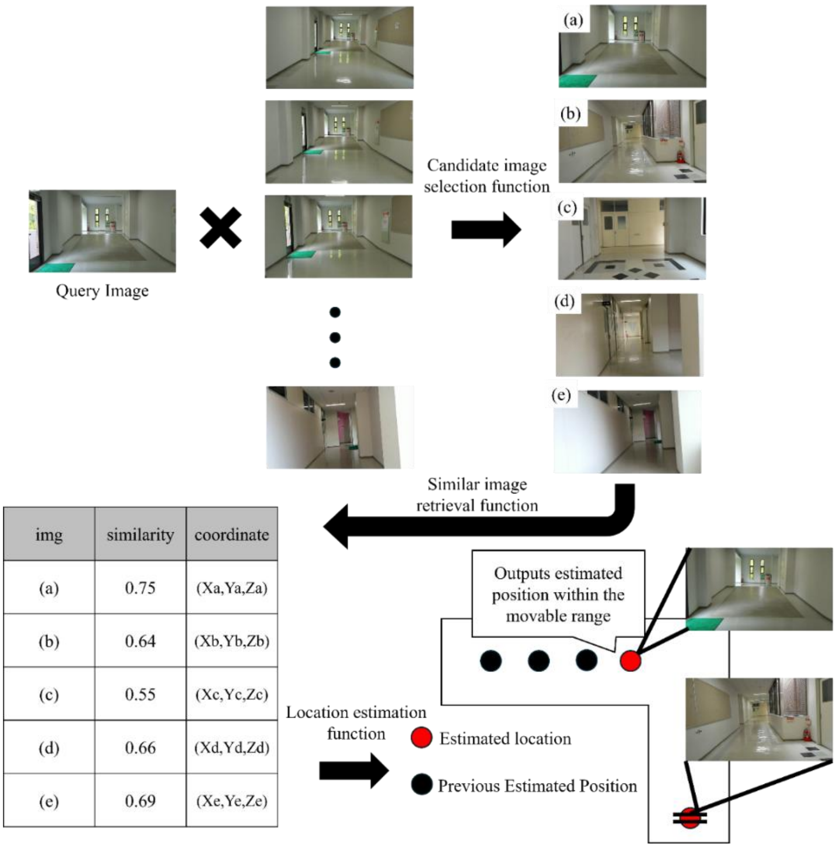
In the "Candidate Image Selection Function," the distance to the feature point obtained by SIFT is used as the basis for calculating the similarity between each image and the target images to be searched for. First, local features are calculated using SIFT for images on the traversing path and images used for self-position estimation. However, if local features of the images on the traversing path have already been calculated, they are calculated only for the image used for self-position estimation. The Hamming distance to the corresponding feature is then calculated using Brute-Force Matcher, and the average of all Hamming distances calculated for each image is obtained. This selects a group of images with a certain number of matching features.

2.3 Similar Image Retrieval Function

The "Similar Image Retrieval Function" uses dHash to estimate and output images with a threshold level of similarity or higher with respect to the images selected in the "Candidate Image Selection Function." First, a hash value is obtained from each image using dHash. Specifically, the grayscale image is divided into regions of a certain size, and the difference in luminance between adjacent regions is calculated. Then, based on the calculated results, the lightness and darkness of the left and right areas are expressed as a string of 01 and output as a hash value. Next, the hash values of the images on the traversing path are compared with the hash values from the images used for self-position estimation, and the Hamming distance is calculated from the XOR operation results. The calculated Hamming distance is normalized in the range of 0–1, and the value is used as the similarity. Furthermore, images on the traversing path with similarity above a threshold value are output as similar images.

2.4 Location Estimation Function

The "Location Estimation Function" estimates the most appropriate location on the map of the traversing route from the images with high similarity estimated by the "Similar Image Retrieval Function." First, the system obtains the coordinates associated with the images on the traversing path that have a similarity greater than a threshold. Then, using the acquired coordinate values and the estimation result of the previous shooting position, the system outputs the coordinate values associated with the image with the highest similarity within the range where the drone can move from the previous position to the current shooting position(Figure 5). However, in the absence of a previous estimated position, the system outputs the position of the image with the highest similarity as the current shooting position.



img	similarity	coordinate
(a)	0.75	(Xa,Ya,Za)
(b)	0.64	(Xb,Yb,Zb)
(c)	0.55	(Xc,Yc,Zc)
(d)	0.66	(Xd,Yd,Zd)
(e)	0.69	(Xe,Ye,Ze)

Query Image
Candidate image selection function
(a)
(b)
(c)
(d)
(e)
Similar image retrieval function
Outputs estimated position within the movable range
Location estimation function
● Estimated location
● Previous Estimated Position

Figure 5. Diagram of each function

3. Results

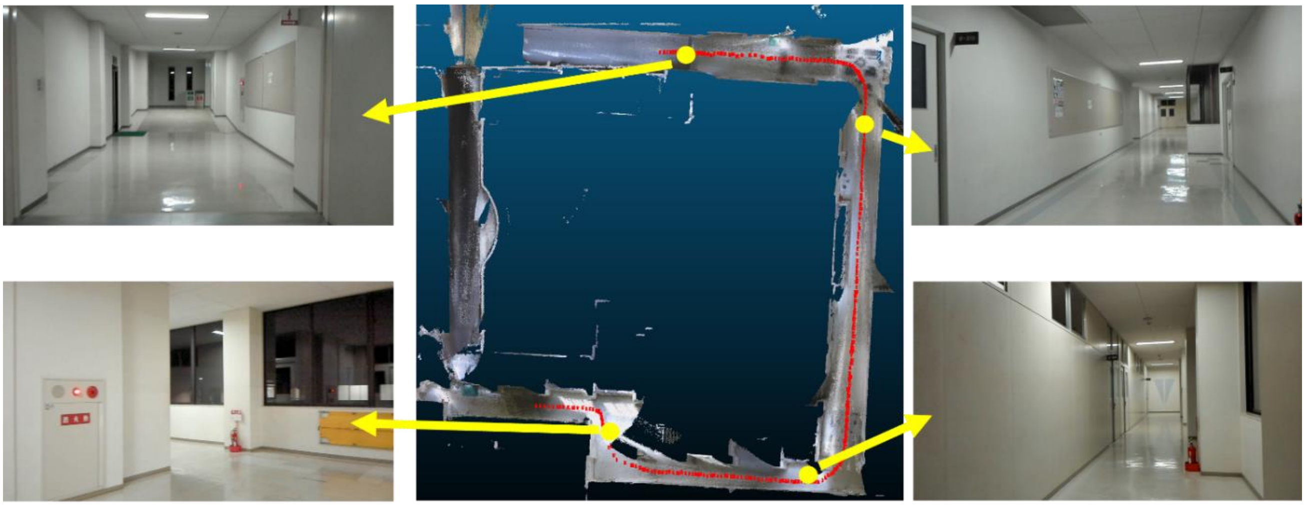
3.1 Verification Experiments

We applied the proposed method to an indoor corridor at night, assuming two types of situations: one in which the situation is the same as when the image was taken beforehand, and another where the situation has changed. Then, we verified the applicability of the proposed method to self-position estimation for automatic patrols by security drones. The experiment location was an L-shaped corridor on a university campus, which has many plain walls and straight sections that are difficult to estimate using the proposed method and previous research methods. For the input data, a 3D map consisting of point cloud data was constructed using a unit that can measure point cloud data by SLAM, as used in previous work (Kajitani et al., 2024). The measurement unit recorded images of the patrol route in advance and linked the coordinate values to the map (Figure 6).

In the present experiment, to compare with the previous study and to verify the effects of changes in local features, we also verified the case in which the objects that are features, such as door opening and closing and installation locations, which are likely to be features in self-position estimation, are varied in each case of the previous study (Yamashita et al., 2024) and the proposed method. In addition, each input image must match the shooting conditions of a small drone that can fly indoors. To simulate flight, we raised a hand-held web camera capable of capturing RGB images to the same height as the small drone's flight altitude. During the evaluation, we compared the self-positions estimated by each method with the actual shooting positions. We then compared the percentage of correctly estimated positions to the shooting positions at all locations, thereby confirming the usefulness of the proposed method. Furthermore, we verified the applicability of the proposed method from the viewpoint of applying it to automatic patrols by security drones.

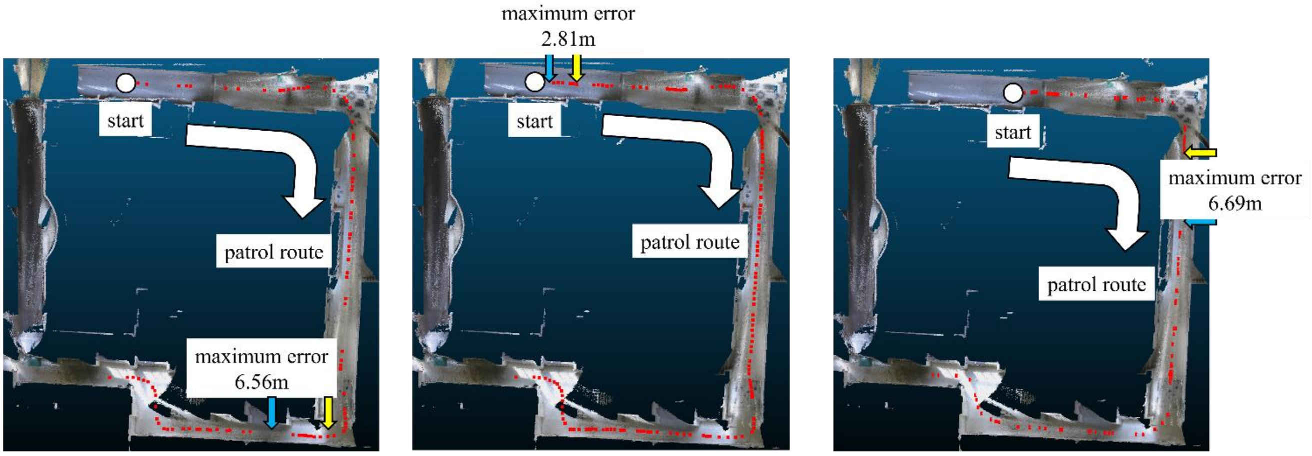
3.2 Experiments Results

Figure 7(a) shows the visualization results of position estimation using only dHash without changing local features; Figure 7(b) shows the visualization results of position estimation using the proposed method; and Figure 7(c) shows the visualization results of position estimation using the proposed method with changing local features. Table 1 shows the percentage of correct responses, the average position error, and the maximum error for similar images in each result. Notably, the location estimation process, which solely relied on dHash and required input images with local feature changes, experienced significant failure. Consequently, we didn't verify the visualization results or compute the error amount.



Red Point： Location of images on the tied patrol route

Figure 6. Map visualization of patrol routes



(a) dHash・No change in local features (previous study)

(b) dHash+SIFT ・No change in local features (proposed method)

(c) dHash+SIFT ・changes in local features (proposed method)

Red Point： Estimated shooting position

Blue Arrow： Actual shooting position   Yellow Arrow： Wrongly estimated location

Figure 7. Location Estimation Visualization Results

Table 1. Percentage of correct answers in each result

Local feature	Method	Number of input images	Number of correct images	Correct responses	Average position error	Maximum error
No change	dHash	216	153	70.8%	2.54m	6.56m
	dHash+SIFT (Our Method)	216	209	96.3%	2.14m	2.81m
Change	dHash	129	36	27.9%	error	error
	dHash+SIFT (Our Method)	129	106	82.8%	3.37m	6.69m

4. Discussion

As shown by visualization results of position estimation in Figure 7(b) and (c), the position of the drone on its patrol path is generally estimated from the webcam image, even when SIFT is combined with SIFT. The comparison of the percentage of correct responses in Table 1 confirms that the combination of SIFT produces more accurate location estimation results in both cases, with and without changes in local features. This suggests that the method improves two issues in the existing approach. Comparing the respective results with no change in local features, the maximum values of the percentage of correct answers and position errors confirm the improvement in the accuracy of position estimation. Specifically, the calculation of more than 90% of the correct answers validates the feasibility of location estimation across all sections. Therefore, we confirm that combining a method such as dHash, which generously captures overall features, and SIFT, which is a feature point-based method, is useful even with few features, such as in a corridor or in a straight section, which has been an issue in previous studies.

The combined dHash and SIFT method was generally successful in estimating the input images when local features were changing. However, when the results were compared with and without local features, the percentage of correct responses decreased and the location error worsened. In fact, when we checked the location where the maximum error occurred in Figure 7(c), we found several locations where local features changed (Figure 8). Thus, it is unlikely that feature point-based methods alone can further improve estimation accuracy in locations with many changes in local features. It is expected to achieve higher accuracy by comparing the results of similarity calculations between the feature point-based method and dHash and using the more reliable estimation result as a reference for location estimation. However, this experiment assumed that the actual patrolling guards might be in a dark place with no lighting, despite taking the input images with the lighting on. Therefore, we need to enhance the method's sensitivity to light intensity changes using gamma correction or deep learning to produce images that mimic a quasi-bright environment.

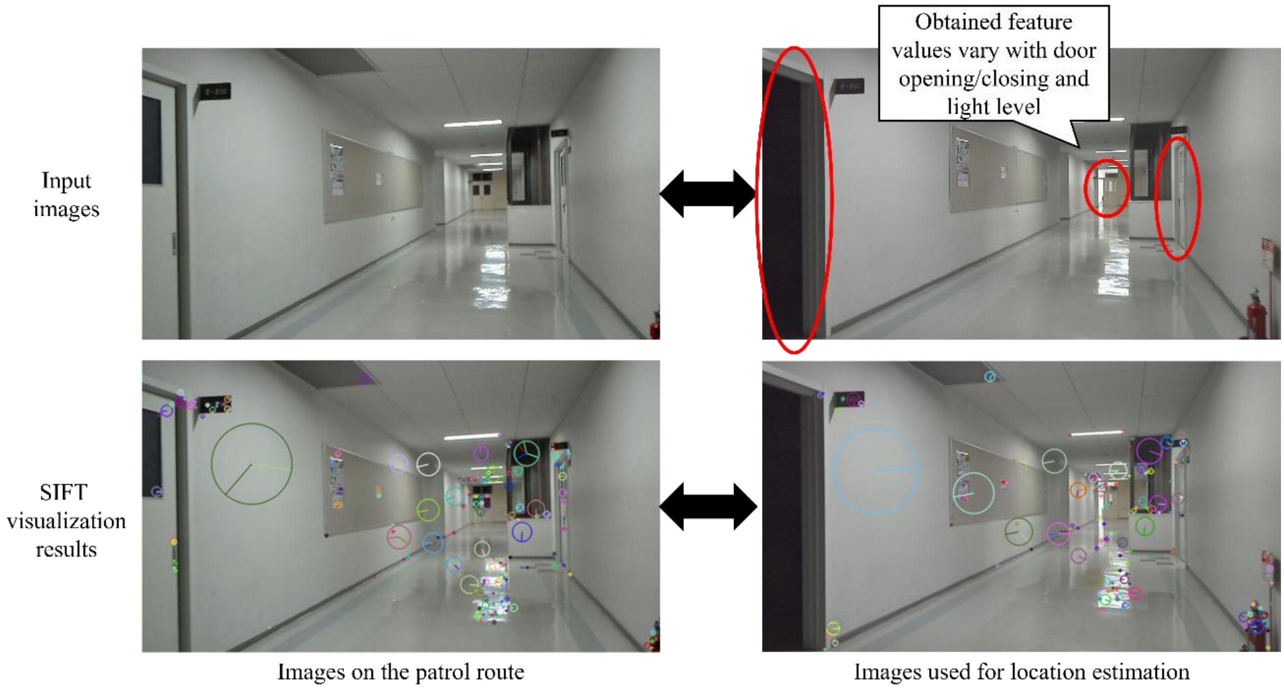

Obtained feature values vary with door opening/closing and light level
Input images
SIFT visualization results
Images on the patrol route
Images used for location estimation

Figure 8. Locations of maximum error in Figure 7(c) results

5. Conclusions

In this paper, we propose a method for estimating self-location by combining SIFT and dHash to search similar images for indoor areas with few features. Empirical experiments confirm that the overall accuracy can be improved, even for straight sections that previous studies found difficult to estimate. In the future, we aim to make this method robust to changes in light intensity and develop a self-position estimation technique that can be applied even at night.

Author Contributions

Conceptualization, M.N. and S.Y.; methodology, M.N., Y.T., Y.U. and S.Y.; software, S.Y.; validation, S.Y.; formal analysis, S.Y.; investigation, S.Y.; resources, M.N. and S.Y; data curation, S.Y.; writing—original draft preparation, S.Y.; writing—review and editing, M.N., Y.T. and Y.U; visualization, S.Y.; supervision, M.N., Y.T. and Y.U.; project administration, M.N.

Funding

This research received no external funding.

Informed Consent Statement

Not applicable.

Conflicts of Interest

The authors declare no conflict of interest.

References:

Kajitani, M., Nakahara, M., Tsukada, Y., Umehara, Y., Nishida, Y., Shimizu, N. & Tanaka, S. (2024). Research on correction of point cloud data by handheld laser scanner with SLAM. *Proceedings of the 86th National Convention of IPSJ*, *86*(4), 805-806.

Matsumoto, Y., Nakano, G. & Ogura, K. (2024). Indoor visual localization using point and line correspondences in dense colored point cloud. *2024 IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*, 3604-3613.

Ministry of Health, Labour and Welfare. Employment referrals for general workers. (2024). https://www.mhlw.go.jp/toukei/list/114-1.html

Okamoto, K., Kazama, H. & Kawamoto, K. (2012). A fuzzy codebook based image search method for visual localization in libraries. *28th Fuzzy System Symposium*, 28, 444-447.

Skydio. Skydio2+. (2024). https://www.skydio.com/skydio-2-plus-enterprise

Yamashita, S., Nakahara, M., Tsukada, Y. & Umehara, Y. (2024). Research on estimation technique of self-location indoors using similar image retrieval technique. *Proceedings of the symposium on civil engineering informatics*, 49, 205-208.

Yamazaki, K., Shishido, H., Kitahara, I. & Kameda, Y. (2019). Evaluation for harmonic location estimation system of image retrieval and SLAM. *International Workshop on Advanced Imaging Technologies 2020*

Yang, Y., Toth, C. & Brzezinska, D. (2020). A 3D map aided deep learning based indoor localization system for smart devices, *The International Archives of the Photogrammetry Remote Sensing and Spatial Information Sciences*., XLIII-B4-2020, 391–397.

Journal of Digital Life 2025,5,S4

Special Issue: Measurement, control, and analysis of motion using ICT and AI
Article

A Study on the Development of a Traffic Volume Counting Method by Vehicle Type and Direction Using Deep Learning

Yuhei Yamamoto [1], Masaya Nakahara [2], Ryo Sumiyoshi [3], Wenyuan Jiang [4], Daisuke Kamiya [5] and Ryuichi Imai [6]*

[1]Faculty of Environmental and Urban Engineering, Kansai University, 3-3-35 Yamate-cho, Suita-shi, Osaka 564-8680, Japan
[2]Faculty of Information Science and Arts, Osaka Electro-Communication University, 1130-70 Kiyotaki, Shijonawate-shi, Osaka 575-0063, Japan
[3]Doctoral Course Graduate School of Engineering and Design, Hosei University, 2-33 Ichigaya-tamachi, Shinjuku-ku, Tokyo 162-0843, Japan
[4]Faculty of Engineering, Osaka Sangyo University, 3-1-1 Nakagakiuchi, Daito-shi, Osaka 574-8530, Japan
[5]Faculty of Engineering, University of the Ryukyus, 1 Senbaru, Nishihara-cho, Nakagami-gun, Okinawa 903-0213, Japan
[6]Faculty of Engineering and Design, Hosei University, 2-33 Ichigaya-tamachi, Shinjuku-ku, Tokyo 162-0843, Japan

Abstract
The turning movement count is investigated to understand the traffic conditions at intersections and identify bottleneck locations. In recent years, methods utilizing probe data and AI-based analysis of video images have been developed to streamline the survey process. Existing methods can count vehicles as they pass but struggle to classify vehicle types. Therefore, the objective of this study is to develop a method for counting turning movement count by vehicle type using deep learning. In this method, YOLOv8 is used to detect cars, buses, and trucks in video images, and BoT-SORT is used for tracking. When a vehicle being tracked crosses the cross-sectional lines and auxiliary lines at the intersection captured in the video images, it is counted by class. In this case, the entry direction of vehicles that cannot be determined upon entering the intersection is estimated based on accurately counted vehicles. Additionally, the entry direction is inferred from a series of vector information within the detection bounding boxes. The results of the verification experiment showed that the proposed method can count the directional traffic volume with an accuracy of over 95.0% and classify the three vehicle classes—car, bus, and truck—with an accuracy of over 90.0%.

Keywords: *Turning Movement Counts, Vehicle, Deep Learning, Image Processing, Classification*

1. Introduction

In many countries, turning movement counts surveys are conducted to understand the usage of roads (Japan International Cooperation Agency, 2018 and Streetlight Data, 2024). In Japan, turning movement counts are counted by turning movements (right-turn, left-turn, and straight) and vehicle type to understand traffic conditions at intersections and identify bottleneck locations. This survey requires at least four surveyors per intersection, leading to increased survey costs as the number of survey locations increases. For example, in Tokyo, a large-scale survey was conducted as part of the Major Intersection Traffic Volume Survey, covering 125 intersections and requiring more than 500 surveyors (Metropolitan Police Department, 2023). Against this backdrop, in recent years, the Ministry of Land, Infrastructure, Transport and Tourism has been exploring survey methods that utilize probe data to streamline the process, as well as methods that analyze recorded video images using AI (Ministry of Land, Infrastructure, Transport and Tourism, 2019). A survey method using probe data has demonstrated the potential to count turning movement counts by combining ETC 2.0 probe data with data collected from vehicle detectors (Shiomi, 2022).

Received: 21 November 2024, Revised: 29 January 2025, Accepted: 27 February 2025, Published: 26 March 2025

* Correspondence: imai@hosei.ac.jp

However, when the penetration rate of vehicles equipped with ETC 2.0 probe data is low, the accuracy of traffic volume counting decreases, presenting a significant challenge. Although the penetration rate is expected to increase as more vehicles are equipped with ETC 2.0 onboard units, the installation incurs additional costs. Therefore, it is challenging to rapidly promote the widespread adoption of onboard units. In response, we focused on a survey method that analyzes video images using AI. As survey methods using AI, there are two primary approaches: one involves counting based on vehicle trajectories (Horii et al., 2022), and the other sets cross-sectional lines on roads visible in video images and counts vehicles passing through these lines (Watanabe et al., 2023). However, the former method faces a challenge in that vehicle trajectories differ for each intersection, requiring parameter adjustments for counting every time the target intersection changes. This challenge could potentially be resolved by predefining the camera angles during filming, which may reduce variations in vehicle trajectories specific to each intersection. However, this approach cannot be applied to intersections that do not fit the predefined camera angles, leading to a reduction in versatility. The latter issue, as shown in Fig.1, arises from the occlusion that occurs when vehicles overlap near the cross-sectional line, causing vehicles farther from the camera to be obscured, which leads to counting omissions. To address this issue, we have developed a method for counting turning movement counts that sets auxiliary lines in addition to cross-sectional lines as a countermeasure against occlusion (Sumiyoshi et al., 2024). However, this method cannot count turning movement counts categorized by vehicle type. Therefore, the purpose of this study was to develop a method for counting turning movement counts by vehicle type using deep learning applied to video images of intersections. In Section 2, the proposed method is explained in detail, and the experimental conditions for verifying its effectiveness are described. Section 3 evaluates and discusses the results of the demonstration experiments. Section 4 provides a summary of this study.

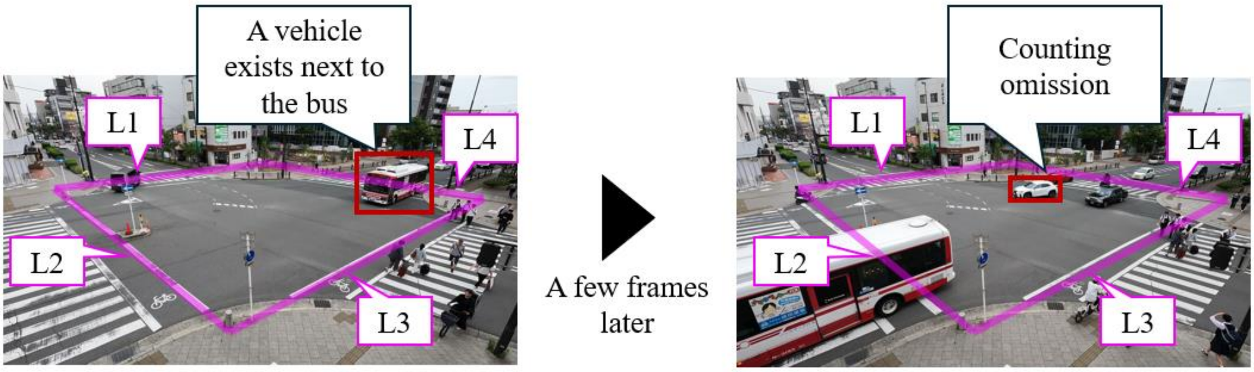


Fig.1. Scenarios where counting omissions occur in existing methods

2. Methods

In this section, we summarize the challenges identified in existing research and outline the development strategy of the method devised in this study. Next, we provide a detailed explanation of the proposed method. Then, we describe the conditions of the empirical experiments conducted using this method.

2.1 Development Approach for Counting Turning Movement Counts by Vehicle Type

This section organizes the challenges identified in existing studies and outlines the development approach for the method proposed in this study (Watanabe et al., 2023). This method has been reported to result in counting omissions when occlusion occurs within the region enclosed by the cross-sectional line, causing the vehicle ID to switch (see Fig.1). Hamamura et al. utilized YOLOv7, an object detection method, fine-tuning it with images of passenger cars, light trucks, buses, and motorcycles to count cross-sectional traffic volumes by vehicle type (Hamamura et al., 2023). Additionally, Okubo et al. employed the SSD object detection method, training it with images of light trucks, buses, passenger cars, small trucks, pedestrians, motorcycles, and bicycles to count cross-sectional traffic volumes by the trained classes (Okubo et al., 2020). These methods aim to count cross-sectional traffic volumes, leading to the placement of cross-sectional lines closer to the camera. However, to count turning movement counts, it is necessary to place cross-sectional lines at positions farther from the camera. Therefore, near the cross-sectional lines farther from the camera, vehicles appear smaller in the footage, potentially leading to a decrease in vehicle classification accuracy.

Based on the above, this study proposes a method to address undercounting caused by occlusion. The method involves analyzing traffic conditions using vehicles that are correctly counted and estimating the inflow directions of vehicles for which only outflow directions can be determined. Additionally, to improve the classification accuracy of small vehicles appearing in the footage, this study employs the YOLOv8x model, which can detect smaller objects with higher precision than YOLOv7 by replacing the Detection Head with one based on NAS-FPN (Varghese, R and Sambath, M, 2024). Furthermore, to enhance vehicle classification accuracy, this study incorporates not only the classification results of vehicles on the cross-sectional line but also employs a majority voting approach based on the classification results of vehicles within the area enclosed by the cross-sectional lines.

2.2 Proposal of a Method for Counting Turning Movement Counts by Vehicle Type

The processing flow of the proposed method is illustrated in Fig.2. This process consists of cross-sectional and auxiliary line setting, detection, counting, interpolation, and vehicle classification.

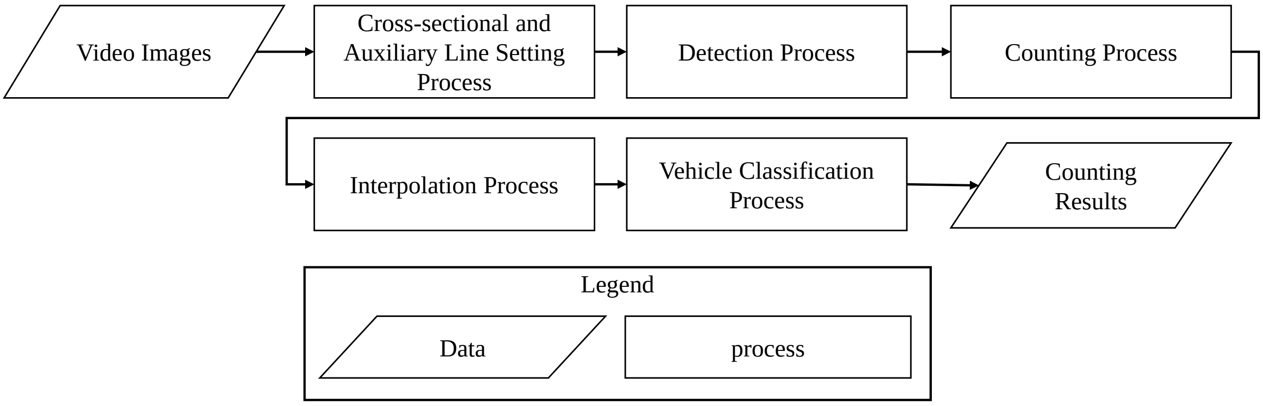


Fig.2. Processing Flow of the Proposed Method

In the cross-sectional and auxiliary line setting process, cross-sectional and auxiliary lines are established to determine the inflow and outflow directions. The flow of the process is illustrated in Fig.3a. First, eight points (points 1 through 8) are manually selected to enclose the intersection. Next, lines are drawn connecting points 1 and 2, points 3 and 4, points 5 and 6, and points 7 and 8. These lines are extended until adjacent lines intersect. Then, auxiliary lines are established by connecting the midpoints of opposing cross-sectional lines. This approach reduces the impact of occlusion within the area enclosed by the cross-sectional lines.

In the detection process, vehicles appearing in the video images are detected and tracked to count turning movement counts. In this process, the YOLOv8x model is used to detect three classes: car, bus, and truck. Next, the detected vehicles are tracked using BoT-SORT (Aharon et al., 2022). During this process, an ID is assigned to each tracked vehicle to prevent excessive counting at the cross-sectional lines.

In the counting process, vehicles crossing the cross-sectional and auxiliary lines are counted. First, when the midpoint of the bottom edge of a bounding box for a detected vehicle passes over a cross-sectional or auxiliary line, the ID of that vehicle is recorded. This allows for determining the direction from which the vehicle entered. Subsequently, when the midpoint of the bottom edge of the bounding box for the detected vehicle crosses another cross-sectional line, the vehicle's ID is recorded again. This process determines the outflow direction and ensures the vehicle is counted as a single unit.

Here, an example is explained for the case where a vehicle enters from L1 direction and exits from L3 direction. In determining the entry direction, it is assessed whether the vehicle has passed the cross-sectional lines and auxiliary lines by verifying if it satisfies equation (1). In this case, x1, y1 represents the intersection of L1 and L4, x2, y2 represents the intersection of L1 and L2, and xp, yp represents the midpoint of the lower edge of the bounding box in the previous frame. Next, the intersection between the line segment connecting the midpoints of the lower edges of the rectangle and L1 is determined using equations (2), (3), and (4). The CCW (Counter Clockwise) function defined in (2) is used to evaluate the geometric configuration of three points. When the result of this function is positive, the points are arranged in a counterclockwise order; when negative, they are arranged in a clockwise order; and when zero, the points are collinear. Furthermore, when both equations (3) and (4) are satisfied, it can be determined that the two line segments intersect, indicating that there is an enters from the direction of L1. In this case, A represents the midpoint of the lower edge of the bounding box in the previous frame, B represents the midpoint of the lower edge of the bounding box in the current frame, C represents the intersection of L1 and L4, and D represents the intersection of L1 and L2. In determining the exit direction, equation (5) is used to assess whether the midpoint of the lower edge of the bounding box in the previous frame is located within the region enclosed by the cross-sectional lines. In this case, x1, y1 represent the intersection of L3 and L4, x2, y2 represent the intersection of L3 and L2, and xp, yp denote the midpoint of the lower edge of the bounding box in the previous frame. Furthermore, similar to the entry direction determination, equations (2), (3), and (4) are used to verify whether the line segment connecting the midpoints of the lower edges of the bounding boxes intersects with L3. If an intersection is confirmed, the vehicle is determined to have exited in the direction of L3.

$$(x_2 - x_1) \cdot (y_p - y_1) - (y_2 - y_1) \cdot (x_p - x_1) \geq 0 \tag{1}$$

$$\mathrm{CCW(A, B, C)} = (y_C - y_A) \cdot (x_B - x_A) - (y_B - y_A) \cdot (x_C - x_A) \tag{2}$$

$$(\mathrm{CCW(A, B, C)}) \cdot (\mathrm{CCW(A, B, D)}) \leq 0 \tag{3}$$

$$(\mathrm{CCW(C, D, A)}) \cdot (\mathrm{CCW(C, D, B)}) \leq 0 \tag{4}$$

$$(x_2 - x_1) \cdot (y_p - y_1) - (y_2 - y_1) \cdot (x_p - x_1) < 0 \tag{5}$$

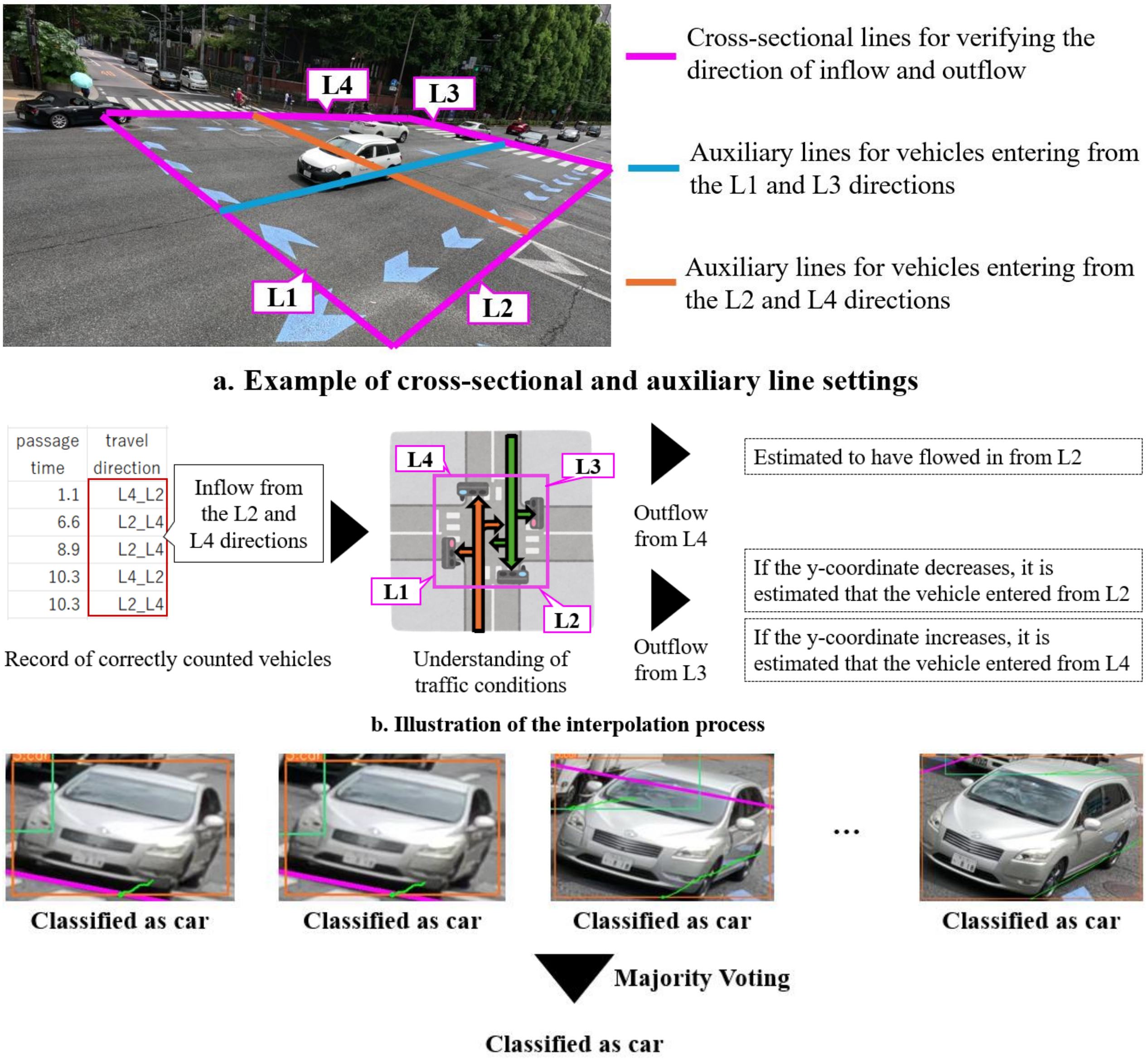
In the interpolation process, vehicles for which the inflow direction cannot be determined are supplemented by estimating their inflow direction, thereby addressing undercounting issues. The flow of this process is illustrated in Fig.3b. First, the inflow direction candidates are estimated for each time point based on the time when the counted vehicles crossed the second cross-sectional line and their travel direction. Next, for vehicles whose inflow direction cannot be determined due to occlusion, the inflow direction is uniquely estimated based on the vector information, including the sequence of timestamps and positions as the vehicle crosses the outflow cross-sectional line.

An example is provided here for the case where a vehicle enters from L2 direction and exits from L3 direction. First, the possible entry direction candidates are estimated based on the time when the target vehicle passed L3 and the time when the correctly counted vehicles passed the second section line. In this example, as shown in Fig. 3b, we assume that vehicles are entering from L2 and L4 directions during this time period. Then, when equation (6) is satisfied, the vehicle is estimated to have entered from the L4 direction, and when equation (7) is satisfied, it is estimated to have entered from the L2 direction. In this case, y2 represents the y-coordinate of the midpoint of the lower edge of the bounding box when the vehicle passes L3, and y1 represents the y-coordinate of the midpoint of the lower edge of the bounding box when the vehicle is first detected within the area enclosed by the section line.

$$y_2 - y_1 > 0 \tag{6}$$

$$y_2 - y_1 < 0 \tag{7}$$

In the vehicle classification process, the counted vehicles are categorized into three classes: car, bus, and truck. The flow of this process is illustrated in Fig.3c. First, the classification results from the YOLOv8x model are recorded for vehicles detected within the area enclosed by the cross-sectional lines. Then, the recorded results are used to determine the vehicle type by applying majority voting to the classification outcomes for each vehicle ID, thereby finalizing the classification.


L4
L3
L1
L2
Cross-sectional lines for verifying the direction of inflow and outflow
Auxiliary lines for vehicles entering from the L1 and L3 directions
Auxiliary lines for vehicles entering from the L2 and L4 directions

a. Example of cross-sectional and auxiliary line settings

passage time	travel direction
1.1	L4_L2
6.6	L2_L4
8.9	L2_L4
10.3	L4_L2
10.3	L2_L4

Inflow from the L2 and L4 directions
L4
L3
L1
L2
Outflow from L4
Outflow from L3
Estimated to have flowed in from L2
If the y-coordinate decreases, it is estimated that the vehicle entered from L2
If the y-coordinate increases, it is estimated that the vehicle entered from L4
Record of correctly counted vehicles
Understanding of traffic conditions

b. Illustration of the interpolation process

Classified as car
Classified as car
Classified as car
…
Classified as car
Majority Voting
Classified as car

c. Conceptual diagram of vehicle type classification process

Fig.3. Processing flow of the proposed method

2.3 Validation of the Proposed Method's Effectiveness

In this study, two validations were conducted to verify the effectiveness of the proposed method. In the first validation, to assess the effectiveness of the interpolation process in the proposed method, we applied both the existing method (Sumiyoshi et al., 2024) and the proposed method to video images of intersections and compared the counting accuracy of turning movement counts. In the second validation, to evaluate the vehicle classification accuracy of the proposed method, vehicle types were classified for the vehicles counted using the proposed method in the first validation. Both validations used video images recorded for 25 minutes at an intersection in Tokyo. The target road consisted of four lanes in total, with two lanes in each direction (see Fig.4). During the recording, approximately 33 vehicles per minute were observed traveling through the intersection. The weather during the recording was cloudy. The video images were captured using a GoPro HERO11 mounted on a survey pole, which was extended to a height of approximately 4.0 meters above the ground. During the recording, the GoPro HERO11 was set to a resolution of 5.3K at 30 fps. In both validations, the number of correctly identified vehicles was verified through manual observation. Precision, recall, and F1-score were calculated to evaluate the performance. However, in Validation 1, vehicles exiting from L1 to L2 and from L3 to L4 had a passage count of zero, making it impossible to calculate evaluation metrics. Therefore, these cases were excluded from the evaluation.

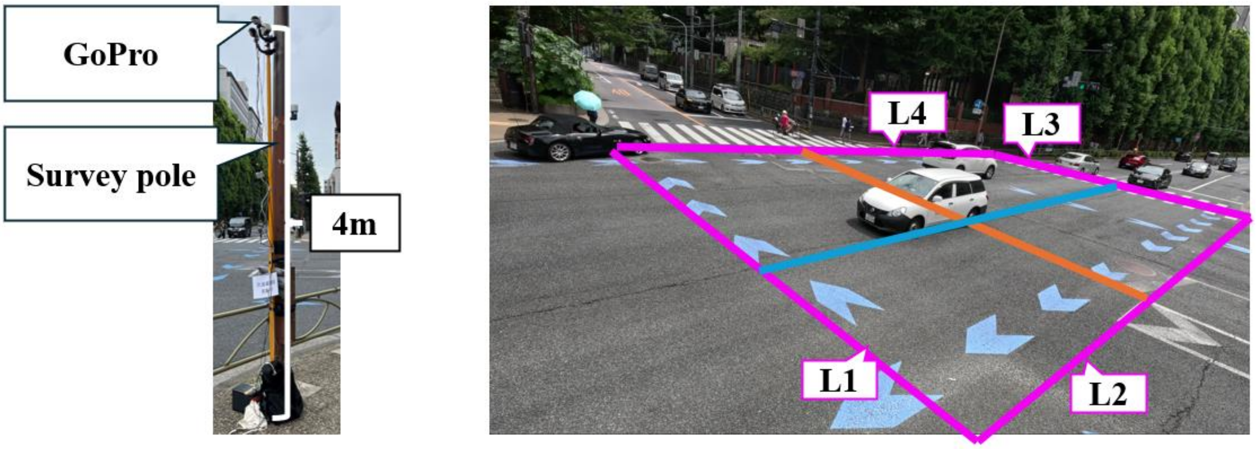


Fig.4. Equipment installation diagram and camera angle during shooting

3. Results

3.1 Validation of Counting Accuracy for Turning Movement Counts

The results of turning movement counts counting are shown in Table 1. First, upon examining the F1-scores, it was found that the proposed method achieved higher scores than the existing method across all directions. Furthermore, in the proposed method, the F1-scores for all directions except for vehicles traveling from L2 to L1 were 0.950 or higher, demonstrating an accuracy comparable to manual observations. Additionally, in the existing method, vehicles traveling in the L3 direction, such as from L1 to L3 and L4 to L3, exhibited low recall rates, indicating a higher incidence of undercounting. The likely cause is that L3 is the farthest cross-sectional line from the camera, making it more prone to occlusion. Similarly, vehicles traveling from L1 to L4 experienced undercounting due to the large number of vehicles traveling from L1 to L3, which caused frequent occlusions. On the other hand, examining the recall rates of the proposed method revealed improvements over the existing method, with vehicles traveling from L1 to L3 achieving a recall of 0.926, from L2 to L3 achieving 1.000, and from L4 to L3 achieving 0.969. Additionally, for vehicles traveling from L1 to L4, the recall rate was 1.000, indicating that detection omissions were successfully mitigated. This indicates that the implementation of the interpolation process in the proposed method, which estimates inflow directions from outflow directions, has the potential to count left-turning vehicles and occluded vehicles, addressing the challenges faced by the existing method. However, for vehicles traveling from L2 to L1, even the proposed method resulted in an F1-score below 0.800. Vehicles traveling from L2 to L1 pass closest to the camera, resulting in their upper sections being visible for only a short duration. This condition led to instances of undercounting. The intersection targeted in this experiment was a large one, with a distance of approximately 70 meters from the camera's position to the farthest crosswalk. Therefore, depending on the size of the intersection, it can be considered that installing two cameras along the diagonals of the intersection can ensure counting accuracy. Furthermore, for vehicles traveling from L1 to L4, fluctuations in the bounding boxes caused by straight-moving vehicles resulted in excessive counting when the midpoint of the lower edge of the bounding box crossed the sectional line (see Fig.5). In this case, because the straight-moving vehicles travel in front of the vehicles that are excessively counted, the upper edge of the bounding box exhibits less movement compared to the lower edge. Therefore, by focusing on the displacement of the upper edge of the bounding box, it may be possible to achieve improvements.

Table 1. Counting results of directional traffic volume

Inflow Direction	Outflow Direction	Ground Truth （vehicles）	Existing Method			Proposed Method		
			Precision	Recall	F1-score	Precision	Recall	F1-score
L1	L3	244	1.000	0.820	0.901	1.000	0.926	0.962
	L4	44	0.909	0.909	0.909	0.917	1.000	0.957
L2	L1	29	0.905	0.655	0.760	0.909	0.690	0.784
	L3	8	1.000	0.875	0.933	1.000	1.000	1.000
	L4	101	1.000	0.980	0.990	1.000	1.000	1.000
L3	L1	223	1.000	0.960	0.979	1.000	0.973	0.986
	L2	24	0.960	1.000	0.980	0.960	1.000	0.980
L4	L1	47	0.947	0.766	0.847	0.978	0.936	0.957
	L2	72	1.000	0903	0.949	1.000	0.931	0.969
	L3	32	0.955	0.656	0.778	0.969	0.969	0.969
All		824	0.986	0.880	0.930	0.989	0.949	0.968

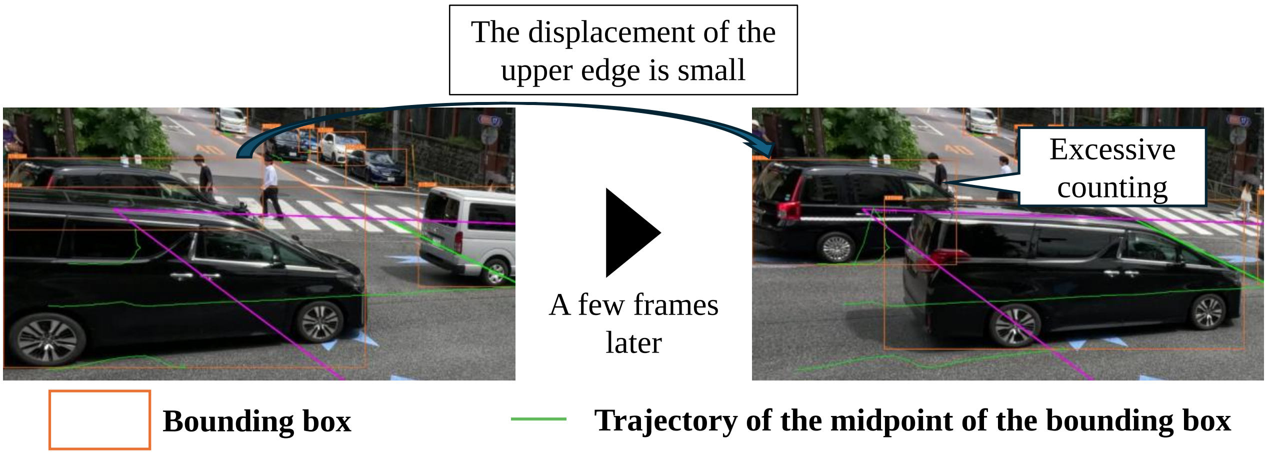

The displacement of the upper edge is small
A few frames later
Excessive counting
Bounding box
Trajectory of the midpoint of the bounding box

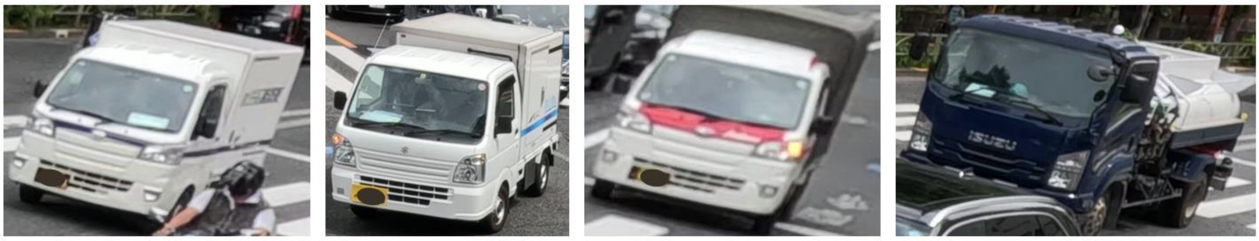
Fig.5. Examples of counting failures

3.2 Validation of Vehicle Classification Accuracy

The results of vehicle classification are shown in Table 2. Upon examining the results, it was found that the F1-scores for car, bus, and truck were 0.900 or higher. Furthermore, when aggregating the classification results across all classes, it was found that the F_1-score was 0.973, demonstrating a higher accuracy than the manual counting accuracy of 95.0%. Furthermore, for vehicles that exhibited misclassifications during tracking, it was found that using time-series data allowed for correct classification through majority voting. Upon examining the images where trucks were misclassified as cars, it was observed that such misclassifications occurred predominantly in scenarios where the front of the vehicle was prominently visible. As shown in Fig.6, this issue could potentially be mitigated by collecting images that prominently feature the front view of vehicles and fine-tuning the YOLOv8x model accordingly. Furthermore, among the 12 trucks misclassified as cars, 11 were small-sized trucks. In turning movement counts surveys, it is a common practice to classify vehicles into categories such as small and large vehicles, with cars typically falling under the category of small vehicles. Therefore, from a practical application perspective, these 11 trucks can be considered to have been correctly classified. On the other hand, upon reviewing the images misclassified as trucks, it was found that a significant number featured boxcars. Similar to the case with trucks, this issue could potentially be addressed by creating training data from images containing boxcars and fine-tuning the model accordingly. Additionally, we have demonstrated that the application of deep-learning-based image classification methods enables highly accurate vehicle type classification when measuring cross-sectional traffic volume by vehicle type. Therefore, when measuring turning movement counts by vehicle type, it is considered feasible to classify vehicles into small and large categories by extracting images of vehicles classified as trucks and applying image classification methods.

Table 2. Classification results by vehicle type

Class	Ground Truth （Vehicles）	Predicted Count （Vehicles）	True Positive Count （Vehicles）	Precision	Recall	F1- score
Car	673	676	664	0.982	0.987	0.984
Bus	7	7	7	1.000	1.000	1.000
Truck	111	108	99	0.917	0.892	0.904
All	791	791	770	0.973	0.973	0.973



※The license plate is manually masked

Fig.6. Examples of images misclassified as car

4. Conclusion

In this study, a method for measuring turning movement counts by vehicle type was developed using deep learning techniques. The results of the empirical experiments demonstrated that implementing an interpolation process to estimate the inflow direction from the outflow direction improved the counting accuracy for left-turning vehicles and occluded vehicles, which had been a limitation of existing methods. Furthermore, it was demonstrated that, except for one direction, the method achieved a counting accuracy equivalent to or exceeding 95.0%, which is the accuracy level typically achieved through manual measurement. In addition, using the existing YOLOv8x model, the method successfully classified the three classes—car, bus, and truck—with an accuracy exceeding 90.0%. In the future, a method will be devised to prevent overcounting by focusing on the upper edge of the bounding boxes during detection. Furthermore, the generalizability of the proposed method will be validated by applying it to videos captured from various angles and under diverse traffic conditions. Additionally, a method will be developed for measuring turning movement counts separately for small and large vehicles using image classification techniques, aiming for practical application in turning movement counts surveys.

Author Contributions

Conceptualization, Y.Y., M.N., R.S., W.J., D.K., and R.I.; methodology, Y.Y., M.N., R.S., W.J., D.K., and R.I..; software, Y.Y., M.N. and R.S.; validation, Y.Y., M.N., R.S., W.J., D.K., and R.I.; formal analysis, Y.Y., M.N., R.S., W.J., D.K., and R.I.;; investigation, Y.Y., M.N., R.S., W.J., D.K., and R.I.;; resources, Y.Y., M.N., W.J., D.K., and R.I.;; data curation, Y.Y., M.N., R.S., W.J., D.K., and R.I.; writing—original draft preparation, R.S.; writing—review and editing, Y.Y., M.N., R.S., W.J., D.K., and R.I.; visualization, Y.Y., M.N., R.S., W.J., D.K., and R.I.; supervision, R.I.; project administration, R.I.; funding acquisition, Y.Y., M.N., W.J., D.K., and R.I. All authors have read and agreed to the published version of the manuscript.

Funding

This research received no external funding.

Acknowledgments

We would like to express our gratitude to SURVEY RESEARCH CENTER Co., Ltd., CHUO KENSETSU CONSULTANT. Co., Ltd., and NIPPON INSIEK Co., Ltd., for providing us with video images and advice for the practical application of this study.

Conflicts of Interest

The authors declare no conflicts of interest.

References

Aharon, N., Orfaig, R. and Bobrovshy, B, Z. (2022). BoT-SORT: Robust associations multi-pedestrian tracking. https://arxiv.org/abs/2206.14651

Hamamura, S., Abe, K., Yamane, S. and Nakamura, H. (2023). Development of a real-time cross-sectional traffic volume measurement system using AI. *Intelligence, Informatics and Infrastructure*, *4*(3), 458-465. (in Japanese)

Horii, D., Sugawara H., Kikuchi, Y. and Okubo J. (2022). A study on automatic measurement of precise traffic engineering indicators volume by intersection direction using deep learning. *Intelligence, Informatics and Infrastructure*, *3*(J2), 819–825. (in Japanese)

Japan International Cooperation Agency. (2018). Traffic surveys and traffic demand forecasting studies in developing countries. https://openjicareport.jica.go.jp/pdf/12339867.pdf (in Japanese)

Jocher, G., Chaurasia, A. and Jing qiu. (2023). Ultralytics YOLOv8, https://github.com/ultralytics/ultralytics

Metropolitan Police Department. (2023). Traffic Volume Statistics Table. https://www.keishicho.metro.tokyo.lg.jp/about_mpd/jokyo_tokei/tokei_jokyo/ryo.html (in Japanese)

Ministry of Land, Infrastructure, Transport and Tourism. (2019). Direction of Traffic Volume and Travel Speed Survey Utilizing ICT. https://www.mlit.go.jp/road/ir/ir-council/ict/pdf03/02.pdf (in Japanese)

Okubo, J., Sugawara, H., Fujii, J. and Ozasa, K. (2020). Object Tracking on Traffic Counting Improve Method. *Intelligence, Informatics and Infrastructure*, *1*(1), 235-241. (in Japanese)

Shiomi, Y. (2022). Estimation on Intersection Turning Volume by Using Traffic Detector and Probe Data. *Journal of Traffic Engineering*, *8*(2), A_213-A_221. (in Japanese)

Streetlight Data. (2024). Turning Movement Counts Explained: Leveraging TMC Analytics for Better, Safer Planning. https://www.streetlightdata.com/turning-movement-count-analytics-explained/

Sumiyoshi, R., Imai, R., Yamamoto, Y., Nakahara, M., Kamiya, D. and Wenyuan, J. (2024). Fundamental Study on Automated Counting Method of Turning Movement Counts at Intersections Using Video Images. *Journal of Digital Life*. (in Japanese)

Varghese, R. and Sambath, M. (2024). YOLOv8: A Novel Object Detection Algorithm with Enhanced Performance and Robustness, *2024 International Conference on Advances in Data Engineering and Intelligent Computing Systems*, 10.1109/ADICS58448.2024.10533619

Watanabe, K., Nakano, K., Nakazawa, M. and Naganuma, K. (2023). Development and its evaluation of a counterline optimization method for directional traffic surveys using MOT. *Transactions of Information Processing Society of Japan*, *64*(2), 511–520. (in Japanese)

Digital Life

Journal of Digital Life 2025,5,S5

Special Issue: Measurement, control, and analysis of motion using ICT and AI
Article

Detecting Near-Miss Actions and Estimating Physical Fatigue among Construction Workers Using Wearable Sensors

Yoshimasa Umehara [1], Toshio Teraguchi [2], Yuhei Yamamoto [3], Taiga Kobayashi [4] and Ryuichi Imai [5]*

[1]Faculty of Business Administration, Setsunan University, 17-8 Ikeda-nakamachi, Neyagawa-shi, Osaka 572-8508, Japan
[2]Faculty of Economics, University of Marketing and Distribution Sciences, 3-1 Gakuen-nishimachi, Nishi-ku, Kobe-shi, Hyogo 651-2188, Japan
[3]Faculty of Environmental and Urban Engineering Department of Civil, Kansai University, 3-3-35 Yamate-cho, Suita-shi, Osaka 564-8680, Japan
[4]Graduate School of Engineering and Design, Hosei University, 2-33 Ichigaya-tamachi, Shinjuku-ku, Tokyo 162-0843, Japan
[5]Faculty of Engineering and Design, Hosei University, 2-33 Ichigaya-tamachi, Shinjuku-ku, Tokyo 162-0843, Japan

Abstract
Labor shortages in the construction industry have become a serious issue in developed countries, particularly in Japan, where workforce aging and declining recruitment of young workers are significant challenges. In this context, ensuring worker safety has become increasingly critical. While occupational accidents in Japan's construction industry have decreased annually due to proper safety measures, the construction industry still has the highest number of fatalities among all industries. Falls from height and falls on the same level are the leading causes of injuries and fatalities. Therefore, detecting near-miss incidents (such as tripping and slipping) that precede falls, along with physical fatigue, could help prevent occupational accidents. This study investigated the feasibility of detecting near-miss incidents and estimating fatigue levels using wearable sensors suitable for continuous monitoring at construction sites. We conducted validation experiments simulating near-miss actions and fatigue conditions. Results showed that applying a Convolutional Neural Network (CNN) to data collected from an iPhone® placed in workers' trouser pockets achieved an F1-score of 0.95 in detecting near-miss actions. Additionally, by comparing body sway magnitudes before and after fatigue, we confirmed the potential for estimating physical fatigue.

Keywords: *Machine Learning, Human Activity Recognition, Fatigue Estimation, Wearable Sensor, Near-Miss Action*

1. Introduction

While global economic growth has led to increased construction demand, the construction industry in developed countries faces severe labor shortages. In the United States, the Infrastructure Investment and Jobs Act of 2021 has outlined a $1.2 trillion infrastructure development plan. However, 88% of U.S. construction companies are experiencing difficulties in securing construction workers (Associated General Contractors of America, 2023). Under these circumstances, Japan's Ministry of Land, Infrastructure, Transport and Tourism is promoting i-Construction to improve safety and labor productivity in the construction industry, resulting in a 6.6% decrease in workplace accidents compared to 2018 (Ministry of Health, Labour and Welfare, 2023a). This reduction can be attributed to the implementation of safety measures, such as KY (*Kiken Yochi*, or hazard prediction) activities that anticipate potential dangers at construction sites and 5S (Sort, Set in order, Shine, Standardize, and Sustain) activities that focus on organization and cleanliness, which are well-known among site managers. Moreover, the Ministry of Health, Labour

Received: 21 November 2024, Revised: 29 January 2025, Accepted: 27 February 2025, Published: 26 March 2025
* Correspondence: imai@hosei.ac.jp

and Welfare formulated the 14th Occupational Safety & Health Program (Ministry of Health, Labour and Welfare, 2023b) in April 2023, which emphasizes the promotion of digital transformation. This program encourages the introduction of cutting-edge technologies for safety measures, such as proximity detection of workers using ICT construction machinery and management of workers' locations and vital data using wearable sensors. These initiatives are believed to contribute to the reduction in the number of occupational accidents. However, the construction industry still has the highest number of fatalities among all industries (Ministry of Health, Labour and Welfare, 2023a). This can be attributed to the inherently high-risk nature of construction work, which involves tasks such as working at heights and operating heavy machinery. Furthermore, physical fatigue resulting from manual labor is believed to affect workers' attentiveness and concentration levels, potentially leading to accidents. According to Heinrich's Law (Heinrich, 1931), a well-known empirical rule in occupational safety, for every serious accident, there are 29 minor accidents, and behind these, there are 300 near-miss incidents. Near-miss incidents in the construction industry include reports of cargo collapse during material loading and cases where outriggers, used to ensure the stability of mobile cranes, sink into the ground (Ministry of Health, Labour and Welfare, 2012). Furthermore, focusing on the factors of occupational accidents that occur in the construction industry, falls from height are the most frequent, followed by falls on the same level (Ministry of Health, Labour and Welfare, 2023a). Therefore, it can be inferred that safety management through dynamic monitoring of construction workers is important. Thus, detecting near-miss actions such as stumbles and slips, which are precursors to falls and trips, could enable the prevention of occupational accidents before they occur. Moreover, estimating the fatigue level of construction workers could help reduce the risk of occupational accidents. By detecting near-miss actions and estimating worker fatigue, the number of occupational accidents can be decreased, and fatalities can be prevented, thereby contributing to the improvement of safety at construction sites. Moreover, estimating the fatigue level of construction workers could help reduce the risk of occupational accidents. By detecting near-miss incidents and estimating worker fatigue, the number of occupational accidents can be decreased, and fatalities can be prevented, thereby contributing to the improvement of safety at construction sites.

In existing research, detecting near-miss actions and estimating physical fatigue have been approached through distinct methodologies. First, focusing on the detection of near-miss actions among construction workers, given that occupational accidents in the construction industry frequently involve falls from height and falls on the same level (Ministry of Health, Labour and Welfare, 2023a), the implementation of wearable sensors has been proposed as a means of detecting falls. Notable examples of wearable sensors currently deployed at construction sites include the Spot-r by Triax Technologies, Inc., and the fall detection device by Takenaka Engineering Co., Ltd. On the other hand, fall detection methods using cameras and LiDAR, which offer higher visibility compared to wearable sensors, have become widespread in the healthcare and welfare sectors, notably the mirAI-EYE by GLORY Ltd., and fall detection sensor for elderly people by FAJ Inc. However, these methods are designed for care recipients and can only be applied in stable environments without blind spots, with a detection range of within 7 meters. Therefore, their adoption in construction sites is hindered by the occurrence of blind spots due to complex structures and the difficulty of wide-area detection. Murata Manufacturing Co., Ltd., has developed a worker safety monitoring system as a wearable sensor capable of detecting near-miss incidents such as trips and slips that may precede falls on construction sites. However, the specific definitions of near-miss actions and their detection algorithms have not been made public. Therefore, we focus on fall and near-miss action detection methods in the sports and welfare fields, where action recognition research has been advancing. In the field of sports, a method for detecting falls of soccer players using the deep learning model LSTM (Naruo et al., 2023) has been devised. The reason this method is effective is that soccer has a limited duration and range of movement, and the players' motion patterns are relatively consistent. On the other hand, LSTM learns patterns from long-term time-series data. Therefore, we infer that it would be difficult to apply this method to construction workers, whose behavior varies greatly depending on differences in site terrain, structure, equipment used, and job type. In the welfare field, methods for detecting near-miss actions using machine learning models, such as SVM and Decision Tree, as well as thresholds (Pang et al., 2019), have been devised. However, these methods are limited to elderly individuals during walking or daily activities. Consequently, they are difficult to apply to construction workers, who exhibit a wide range of complex behaviors, such as working at heights or operating heavy machinery. Therefore, we believe that by proposing a method suitable for detecting near-miss actions of construction workers who perform a wide variety of tasks, we can contribute to the prevention of occupational accidents.

Next, we focus on estimating the fatigue level of construction workers. Generally, in fatigue level estimation, analyses based on physiological indicators from vital sensors, such as heart rate, electromyography, and oxygen consumption, are conducted. However, the utilization of vital sensors remains challenging on construction sites due to factors such as the impact on heart rate for specific occupations (Akagawa et al., 2020) and the contact between fall protection equipment and vital sensors. Consequently, there are still issues hindering their widespread adoption. In the field of sports, where research on fatigue level estimation has been advancing, it is possible to estimate fatigue levels based on exercise load by calculating the amount of exercise from the distance and speed of movement during play, measured using wearable sensors (Yamada et al., 2023). However, the amount of exercise used for estimating fatigue levels is currently calculated from the distance traveled obtained by GNSS positioning (Yamada et al., 2023). This method is difficult to apply to construction sites with complex structures where multipath effects are likely to occur. As a fatigue estimation method that does not use GNSS positioning, there is a method for estimating fatigue levels by measuring body sway using a force plate, since body sway increases with muscle fatigue (Paillard, 2012). Specifically, body sway is measured by having subjects stand upright on a force plate for 30 seconds before and after fatigue. The fatigued state is reproduced by running on a treadmill for 30 minutes, and the results of the experiment show that body sway significantly increases after fatigue (Derave et al., 2002). However, current methods for measuring body sway are limited to precise methods using force plates or cameras, and a method for measuring body sway using wearable sensors has not yet been established. Therefore, if it becomes clear that the fatigue level of construction workers can be estimated based on body sway, which can be measured by wearable sensors, it will be possible to take breaks and reallocate workers according to their fatigue level, which is expected to help prevent the risk of occupational accidents.

Based on the above, this study aimed to investigate the possibility of detecting near-miss actions and estimating fatigue levels by measuring body sway using wearable sensors that enable continuous monitoring even on construction sites where environmental conditions change daily due to ongoing construction work.

2. Methods

2.1 Methods for Detecting Near-Miss Actions

Near-miss action detection is performed utilizing a deep learning model that has been trained to recognize near-miss action patterns. Our approach to near-miss action detection draws upon anomaly detection methodologies from medical and mechanical domains (Masetic et al., 2016), as well as action recognition techniques employing wearable sensors (Inoue, 2016). In accordance with the methodologies, our study employed wearable sensors to acquire triaxial acceleration and triaxial angular velocity measurements. Subsequently, machine learning models were applied to the acquired data. The study evaluated two candidate machine learning models—Random Forest and Convolutional Neural Network (CNN) classifiers—through empirical experimentation to determine the most effective approach for near-miss action detection. The rationale for employing Random Forest lies in its dual advantages: superior generalization performance with overfitting prevention, and computational efficiency. Our Random Forest data application process involves converting and standardizing integer raw data. During the segmentation phase, we calculate statistical features such as maximum, minimum, mean, standard deviation, and interquartile range. In addition to these features, the unit-converted raw data is used as explanatory variables. The reason for using these features is that previous research (Bao et al., 2004) suggests the possibility of classifying operations with high accuracy. Furthermore, although previous research (Bao et al., 2004) indicates the potential effectiveness of FFT-based features, we do not use them in this study because our preliminary experiments showed that similar features were obtained during near-miss incidents and work operations, which could negatively affect model training. The Random Forest parameters were set according to previous research (Breiman, 2001) as follows: the number of trees was set to 100, the random seed was fixed at 42, the Gini function was used as the split criterion, while both the maximum tree depth and the number of features were set to auto-tune.

The adoption of Convolutional Neural Network (CNN) is justified by their capability to effectively learn spatiotemporal features from sensor data through convolutional and pooling layers, as well as their superior performance in action and image recognition tasks. The data processing pipeline for CNN implementation involves unit conversion of raw integer data followed by standardization to generate the explanatory variables. The CNN architecture consists of three convolutional layers and two fully connected layers, following the structure proposed by Zeng (2014). The CNN hyperparameters were configured based on Zeng (2014) as follows: learning rate was set to 0.001 with Adam optimizer for learning rate decay, ReLU was used as the activation function, and cross-entropy was employed as the loss function. The batch size was set to 10, and the model was trained for 1,000 epochs. For early stopping, we set the patience parameter to 30 with a delta value of 0.00001.

2.2 Methods for Estimating Fatigue Levels

In this study, fatigue levels are defined as the amount of change in body sway before and after exercise, and evaluated by comparing the measurement results of body sway before and after exercise. The method for measuring body sway draws from measurement techniques in medical research, including postural sway measurement during quiet standing (Demura et al., 2006) and gait analysis methods based on long-duration walking rhythm patterns (Higashi et al., 2011). These methods analyze parameters such as the geometrical patterns of the center of gravity sway plotted in two dimensions and peak acceleration during the swing phase. However, these methods are difficult to apply to measuring construction workers' body sway as they were conducted under stable conditions that substantially differ from construction site environments. Therefore, this study aims to investigate the feasibility of fatigue estimation by developing a robust and easily applicable method for measuring body sway that can accommodate the variable conditions of construction sites, considering fatigue induced by construction work. The body sway measurement method proposed in this study calculates the mean of differences between moving maximum and minimum values at 3-second intervals from tri-axial acceleration and tri-axial angular velocity data obtained through wearable sensors. If this mean value changes in accordance with the accumulation of worker fatigue and increased physical load, we hypothesize that fatigue levels could potentially be estimated through body sway measurements.

3. Experiment

3.1 Detecting Near-Miss Actions Experiment

The objective of this experiment was to investigate the feasibility of detecting near-miss actions by having participants perform simulated near-miss actions while wearing three different types of wearable sensors.

3.1.1 Experimental Setup and Procedure

The experiment was conducted in front of the Hosei University Shinmitsuke building, under experimental conditions, where near-miss actions and work actions were simulated and performed. By applying Random Forest and CNN to the tri-axial acceleration and tri-axial angular velocity data acquired by three types of wearable sensors during the execution of each action, we verified the wearable sensor and machine learning model suitable for detecting near-miss actions. Table. 1 shows the defined near-miss actions and work actions. Near-miss actions were defined as fall, trip, slip, stagger, and run, which are the most common precursors to falls from height and falls on the same level, which are the most frequent causes of occupational accidents. In addition, there are a vast number of types of work actions performed by construction workers. Therefore, in this study, in order to verify the usefulness of the proposed method, work actions similar to near-miss actions were selected. The work actions were defined as stand up, squat down, sit down, get on all fours, lie down, walk, walk while squatting, crawl under obstacles, and step over obstacles. By simulating the execution of these defined actions and classifying them into two categories: work actions and near-miss actions, we attempted to detect near-miss actions. The number of measurements was 5 times for each of the 9 types of work actions and 10 times for each of the 5 types of near-miss actions per person. By visually checking the videos taken during these measurements and extracting the moments of action, ground truth labels were assigned. The

Table.1 Defined work action and near-miss action

Category	Action
Near-Miss Action	Fall
	Trip
	Slip
	Stagger
	Run
Work Action	Stand up
	Squat Down
	Sit down
	Get on all fours
	Lie down
	Walk
	Walk while squatting
	Crawl under obstacles
	Step over obstacles

measurement time of the extracted work actions and near-miss actions was approximately 3 minutes per person for both.

3.1.2 Materials and Participants

The three types of wearable sensors used in this experiment were: the xG-1 (Yamada et al., 2023) by xSENSING Co., Ltd., as a sports activity tracker used for analyzing exercise and play activities; the iPhone® 12 Pro by Apple Inc. as a smartphone sensor integrated into daily life; and the Apple Watch® Ultra by Apple Inc. as a smartwatch capable of measuring vital signs. The positioning of the wearable sensors is shown in Fig. 1. These three types of wearable sensors were selected because each possesses distinct characteristics, allowing us to determine which wearable sensor is most suitable for detecting near-miss actions. The specific characteristics of each wearable sensor are as follows: the xG-1, attached to the back of the body using a dedicated vest, can acquire high-precision motion data during physical activities; the iPhone® can easily collect everyday motion data; and the Apple Watch® can capture hand movement data during tasks. These wearable sensors were used to collect three-axis acceleration and three-axis angular velocity data at a sampling rate of 50 Hz. The subjects were eight male university students in their 20s, all of whom performed near-miss actions and work actions.

3.1.3 Data Processing and Evaluation

When applying machine learning models, segmentation must be performed on the wearable sensor data where action boundaries are ambiguous before data can be processed. The segmentation process is conceptually illustrated in Fig. 2. As reported in existing literature (Inoue, 2016), conventional segmentation methods typically employ fixed-size windows with constant overlap ratios for data processing. In previous research (Huynh et al., 2005), daily actions were identified by fixing the window slide width to 250ms and setting the window size to 250ms, 500ms, 1,000ms, 2,000ms, and 4,000ms. However, it has been shown that the optimal window size varies for each action, and appropriate settings for the detection target are important. Similarly, the window overlap ratio also requires settings appropriate for the detection target (Inoue, 2016). Based on these findings, since near-miss actions are instantaneous actions, we set the window size to 200ms, 500ms, 1,000ms, and 2,000ms, which are narrower than those in previous research (Huynh et al., 2005), and the overlap ratio to 0%, 30%, 60%, and 90%. We constructed 16 models for each of Random Forest and CNN, for a total of 32 models. By verifying the detection accuracy of near-miss actions using these 32 models, we aim to identify the machine learning model and the window size/overlap ratio during segmentation that are suitable for detecting near-miss actions. The training data for model construction consisted of 7 out of 8 subjects, and the test data consisted of the remaining 1 subject.

For detection accuracy evaluation, we use the F1-score, which is the harmonic mean of precision and recall, with values closer to 1 indicating higher accuracy. The evaluation method compares predicted labels from each window with ground truth labels at each data point. When windows overlap, resulting in multiple predictions for a single data point, the final prediction is determined by majority voting.

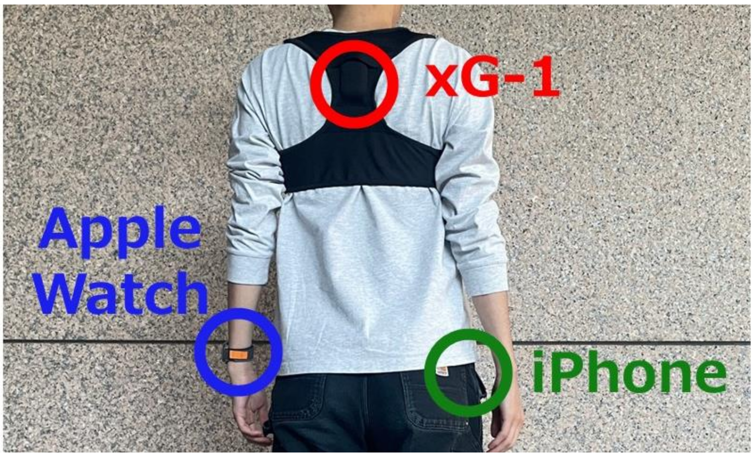


Fig.1 Placement of wearable sensors

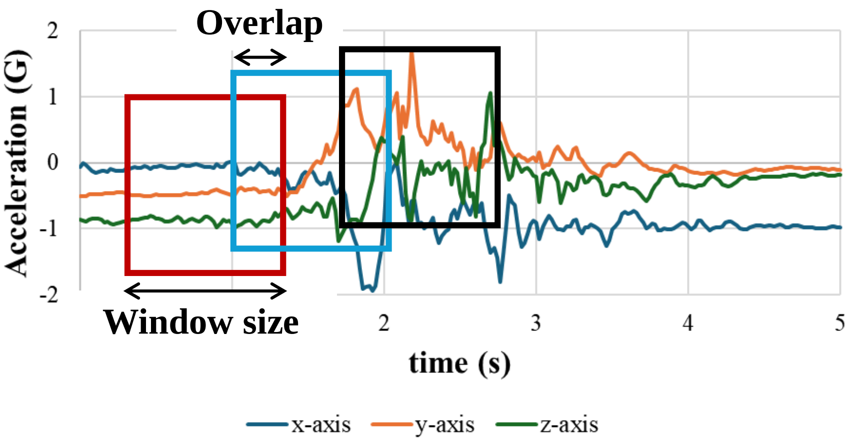


Fig.2 Segmentation image during data loading

3.2 Estimating Fatigue Levels Experiment

The objective of this experiment is to investigate the possibility of estimating fatigue levels by measuring and comparing body sway before and after fatigue using wearable sensors.

3.2.1 Experimental Setup and Procedure

The experiment was conducted in front of the Hosei University Sinmitsuke building. The subject ran 5 km in approximately 20 minutes to induce a state of fatigue. The same actions were performed before and after fatigue, and the change in the amount of body sway was compared. The actions for measuring body sway before and after fatigue were walk, transport, upstairs, and downstairs, which were considered easy to measure body sway due to periodic movements, assuming actual operation at construction sites. Each action was performed at the same pace for 30 seconds before and after fatigue, and data on triaxial acceleration and triaxial angular velocity were acquired.

3.2.2 Materials and Participants

We used xG-1 (Yamada et al., 2023) from xSENSING Co., Ltd., as the wearable sensor. The xG-1 is a sports activity tracker used for analyzing exercise and play patterns, making it suitable for measuring body sway. The sensor placement and data collection methods are identical to those described in Section 3.1.2. The subject was one male student in his 20s.

3.2.3 Data Processing and Evaluation

Since the acquired raw data of triaxial acceleration and triaxial angular velocity are integer values, the integer value of acceleration is converted to G, and the integer value of angular velocity is converted to deg/s. Then, using the proposed method, body sway before and after fatigue is compared, and if a difference is observed between before and after fatigue, it is evaluated that it is possible to estimate the fatigue levels.

4. Results & Discussion

4.1 Experimental Results and Discussion on Detecting Near-Miss Actions

The results of Random Forest application are shown in Fig. 3. In Fig. 3, the left horizontal axis represents the window size, the right horizontal axis shows the overlap rate, and the vertical axis indicates the F1-score, where higher plot points represent higher detection accuracy. The highest detection accuracy was achieved with the xG-1 sensor, using a window size of 200ms and an overlap rate of 90%, resulting in an F1-score of 0.45, indicating that Random Forest could hardly detect near-miss actions. These results suggest that near-miss action detection using Random Forest is challenging. The low detection accuracy of Random Forest can be attributed to two main factors: insufficient utilization of time-series data characteristics and inadequate feature extraction and selection. While Random Forest excels at handling correlations between individual features, it struggles to directly model temporal dependencies. This limitation likely resulted in missing crucial information when detecting near-miss actions, which involve subtle movement changes over short periods. The results of the CNN implementation are presented in Fig. 4. The highest

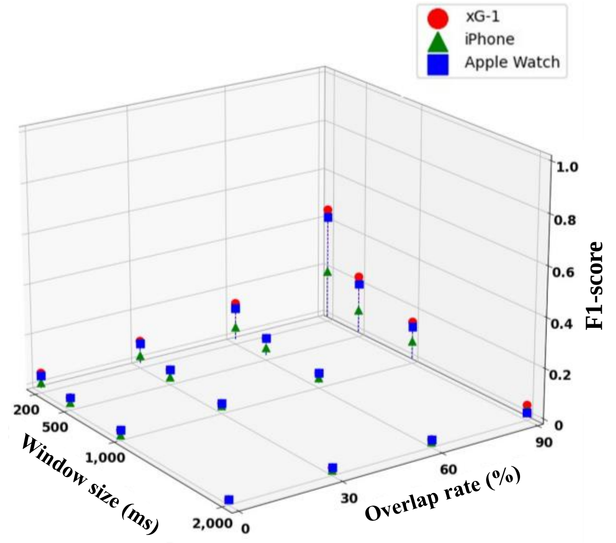


Fig.3 F1-scores for each parameter obtained during Random Forest validation

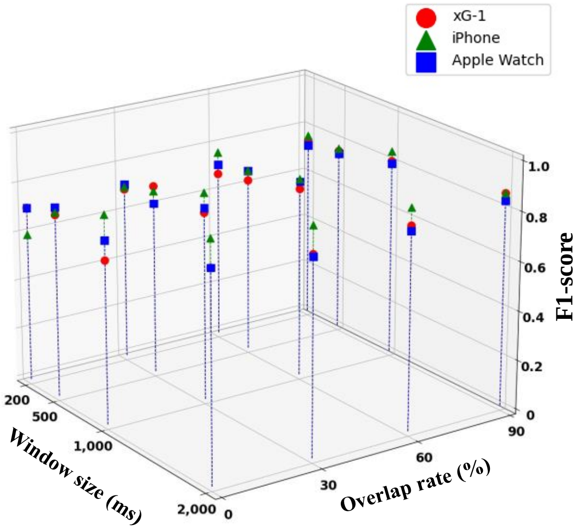


Fig.4 F1-scores for each parameter obtained during CNN validation

detection performance was obtained when using an iPhone® with a window size of 2,000ms and a 0% overlap, achieving an F1-score of 0.95, indicating that CNN effectively detects near-miss actions with high precision. The substantial enhancement in F1-score through CNN implementation can be explained by CNN's inherent capability to recognize local patterns within multidimensional data. The improvement in F1-scores with larger window sizes can be attributed to the increased number of data points, enabling the learning of more features and recognition of overall motion patterns. The F1-score peaked at 0% overlap because each window remains independent, preventing prediction labels from being influenced by other windows. Conversely, when overlap exists, identical data points may be included in multiple windows, potentially resulting in different prediction labels for each window. In this case, prediction labels for identical data points are determined by majority voting, causing incorrect predictions to affect the overall results and decrease accuracy. Therefore, when different prediction labels are obtained for the same data point, alternative methods to majority voting should be considered for label determination. Among wearable sensors, the iPhone® showed the highest detection accuracy when applying CNN. This is likely because near-miss action characteristics are more prominently displayed around the waist area. Furthermore, when comparing detection accuracy across different types of near-miss action, falling motions showed the lowest accuracy. This can be attributed to the similarity between falling motions and lying down actions performed during work tasks.

4.2 Experimental Results and Discussion on Estimating Fatigue Levels

Fig. 5 and 6 present quantitative analyses of relative changes in body sway magnitude, measured via three-axis acceleration and angular velocity, comparing pre- and post-fatigue conditions. The coordinate system establishes the x-axis as an anteroposterior, y-axis as vertical, and z-axis as mediolateral directions. Analysis revealed a consistent pattern of increased post-fatigue body sway across multiple movement patterns, with particular emphasis on carrying out tasks designed to simulate construction worker activities. Stair descent movements exhibited a pronounced susceptibility to increased body sway magnitude. Differential analysis of fatigue-induced changes across individual axes demonstrated minimal perturbation along the vertical y-axis, while substantial variations were observed in both the anteroposterior (x-axis) and mediolateral (z-axis) directions. This axis-specific response pattern can be attributed to the inherent stability of vertical components versus the heightened susceptibility of horizontal plane movements to fatigue-induced oscillations. Of particular significance, stair descent movements demonstrated markedly elevated fatigue-induced body sway compared to other assessed movements, suggesting enhanced sensitivity to physical fatigue. This heightened response during stair descent can be mechanistically linked to the substantial energetic demands associated with controlled vertical displacement during the movement sequence.

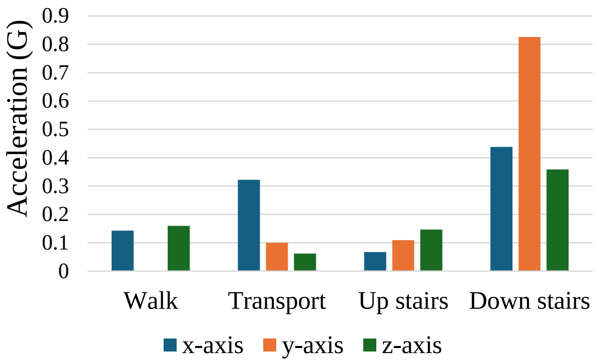


Fig.5 Relative changes in acceleration before and after fatigue

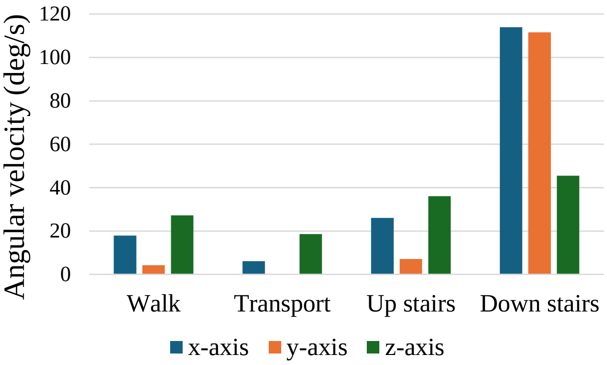


Fig.6 Relative changes in angular velocity before and after fatigue

4.3 Discussion on Generalizing to Actual Construction Environments

In this study, we verified the possibility of detecting near-miss actions and estimating fatigue levels in an experimental environment with limited actions. However, since actual construction sites involve a vast variety of actions and diverse environments, we will discuss how the findings obtained in this study can be generalized.

In detecting near-miss actions, actual construction sites involve work at heights, work in unstable locations such as scaffolding, and work performed by multiple people. Therefore, even if the same action is performed, the actual behavior may differ. Thus, we infer that the robustness of near-miss actions detection can be improved by collecting comprehensive action data of actual construction workers and building a model.

In estimating fatigue levels, this study measured body sway for periodic actions. Since it was shown that body sway increases after fatigue in the action of going down the stairs, the proposed fatigue level estimation method may be applicable to actions that are periodic and require balance at actual construction sites. However, this study used student subjects in their 20s, and it is necessary to consider individual differences due to the diverse age groups and physiques of construction workers in actual operation. Therefore, it is considered that it is possible to estimate the fatigue levels corresponding to individual differences by measuring the body sway data in the state before fatigue of each individual and calculating the amount of increase in body sway by comparing it with that.

5. Conclusion

In this study, we investigated the feasibility of detecting near-miss actions and estimating physical fatigue levels using wearable sensors suitable for continuous monitoring at construction sites. Initially, we evaluated various wearable sensors, machine learning models, and segmentation methods appropriate for near-miss action detection. The results demonstrated that applying CNN to data collected from an iPhone® placed in a trouser pocket achieved near-miss action detection with an F1-score of 0.95. This suggests that our proposed detection method could effectively identify near-miss actions at construction sites. Furthermore, the use of smartphones as familiar, unobtrusive sensors integrated into daily life could facilitate widespread adoption among construction workers, potentially contributing to accident prevention in construction environments.

Subsequently, we examined the possibility of fatigue estimation using the xG-1 sports activity tracker. The results indicated that physically demanding activities, such as descending stairs and carrying loads, exhibited notably increased body sway under fatigue conditions. This suggests the potential for estimating construction workers' fatigue levels during their duties. Such fatigue estimation could enable improved site management through appropriate worker allocation, particularly for those prone to fatigue, thereby preventing accidents proactively.

Future research will focus on validating near-miss action detection and fatigue estimation capabilities using data collected from actual construction sites. Additionally, we plan to estimate near-miss action locations through GNSS positioning and correlate them with site conditions to establish the practical applicability of our detection method. Furthermore, we will assess the effectiveness of our fatigue estimation approach by comparing estimated fatigue levels with subjective fatigue assessments obtained through worker questionnaires.

Author Contributions

Conceptualization, Y.U., T.T., Y.Y., T.K. and R.I.; methodology, Y.U., T.T., T.K. and R.I.; validation, Y.U., T.T., Y.Y., T.K. and R.I.; formal analysis, Y.U., T.T., Y.Y., T.K and R.I.; investigation, T.K.; resources, R.I.; data curation, T.K.; writing—original draft preparation, T.K.; writing—review and editing, Y.U., T.T., Y.Y., T.K. and R.I.; project administration, R.I.; All authors have read and agreed to the published version of the manuscript.

Funding
This research received no external funding.

Institutional Review Board Statement

The study was conducted according to the Declaration of Helsinki.

Informed Consent Statement

Informed consent was obtained from all subjects involved in the study.

Acknowledgments
We would like to express our gratitude to xSENSING Co., Ltd., for providing valuable insights into the data characteristics of wearable sports sensors during the execution of this study.

Conflicts of Interest

The authors declare no conflicts of interest.

References

Akagawa, H., Kasai, Y., Iizuka, K., Yamada, S. and Morikawa, N. (2020). Study on Heart Rate Characteristics of Construction Workers Under Hot Conditions. *Report of Obayashi Corporation Technical Research Institute*, *84*(1). (in Japanese)

Bao, L. and Intille, S. S. (2004). Activity Recognition from User-Annotated Acceleration Data. *Lecture Notes in Computer Science*, 3001, 1-17.

Breiman, L. (2001). Random Forests. *Machine Learning*, *45*(1), 5-32.

Demura, S., Kitabayashi, T. and Noda, M. (2006). Measurement and Evaluation of Healthy People's Body-sway. *The Journal of Education and Health Science*, *51*(3), 223-233. (in Japanese)

Derave, W., Tombeux, N., Cottyn, J., Pannier, J. L., and De Clercq, D. (2002). Treadmill Exercise Negatively Affects Visual Contribution to Static Postural Stability. *International Journal of Sports Medicine*, *23*(1), 44-49.

Heinrich, H. W. (1931). *Industrial Accident Prevention: A Scientific Approach*, McGraw-Hill.

Higashi, H., Shigeoka, T., Itokawa, T., Kitasuka, T. and Aritsugi, M. (2011). A Consideration of Features for Fatigue Estimation by Gait Analysis Using Accelerometer. *The Special Interest Group Technical Reports of IPSJ*, 2011-MBL-57(27), 1-8. (in Japanese)

Huynh, T. and Schiele, B. (2005). Analyzing Features for Activity Recognition. *Proceedings of the 2005 Joint Conference on Smart Objects and Ambient Intelligence: Innovative Contextaware Services: Usages and Technologies (sOc-EUSAI '05)*. 159-163.

Inoue, S. (2016). Human Sensing with Wearable Sensors. *Journal of Japan Society for Fuzzy Theory and Intelligent Informatics*, *28*(6), 170-186. (in Japanese)

Ishioka, H. (2020). AI for Digitization of Construction Management: Single Camera Worker Detection, Tracking and Action Recognition in Construction Site. *Technical Research Report of Shimizu Construction Co., Ltd.*, *98*(1), 21-30. (in Japanese)

Masetic, Z. and Subasi, A. (2016). Congestive Heart Failure Detection Using Random Forest Classifier. *Computer Methods and Programs in Biomedicine*, *130*, 54-64.

Ministry of Health, Labour and Welfare. (2012). Visualization of Near-Miss Cases for Participation in Safety and Health Activities, https://safeconsortium.mhlw.go.jp/anzenproject/concour/2012/sakuhin4/images/n096_2.pdf (in Japanese)

Ministry of Health, Labour and Welfare. (2023a). Analysis of Occupational Accidents in 2023. https://www.mhlw.go.jp/content/11302000/001099504.pdf (in Japanese)

Ministry of Health, Labour and Welfare. (2023b). The 14th Occupational Safety & Health Program. https://www.mhlw.go.jp/content/11200000/001253683.pdf

Naruo, T., Yamamoto, Y., Jiang W., Sakamoto, K., Nakamura, K., Tanaka, S., Okazaki, Y. and Yamazaki, Y. (2023). “Research for Detecting Falling of Soccer Players from IMU using LSTM”. *Proc. 85th National Convention of IPSJ*, 331-332. (in Japanese)

Paillard, T. (2012). Effect of General and Local Fatigue on Postural Control: A Review: *Neuroscience & Biobehavioral Reviews*, *36*(1), 162-176.

Pang, I., Okubo, Y., Sturnieks, D., Lord, S. R. and Brodie, M. A. (2019). Detection of Near Falls Using Wearable Devices: A Systematic Review. *Journal of Geriatric Physical Therapy*, *42*(1), 48-56.

Yamada, T., Masaki, H., Matsubayashi, Y., Tanaka, S., Imai, R., Naruo, T., Nakamura, K., Yamamoto, Y., Jiang, W. and Tanaka, C. (2023). Development of Sensing Unit “xG-1” for Visualizing Team Plays. *Journal of Digital Life, 3,* 10.

Zeng, M., Nguyen, L. T., Yu, B., Mengshoel, O. J., Zhu, J., Wu, P. and Zhang, J. (2014). Convolutional Neural Networks for Human Activity Recognition Using Mobile Sensors. *Proceedings of the 6th International Conference on Mobile Computing, Applications and Services (MobiCASE)*, 197-205.