Vol.2022-DPS-191 No.38
Vol.2022-MBL-103 No.38
Vol.2022-ITS-89 No.38
2022/5/27

Human Identification Based On Point Cloud Captured By Small-Size LiDAR.

Shota Yamada[1,a)] Hamada Rizk[1,b)] Hirozumi Yamaguchi[1,c)]

Abstract: The demand for safety-enhancing solutions is on the rise, especially due to COVID-19's rapid spread. In order to track infected cases and hence restrict the spread of the virus, real-time life-logging is an essential application. This application highlights the necessity for a precise human identification technique in situations when cameras are not feasible owing to privacy concerns. The potential of the LiDAR sensor to represent the surrounding world in the form of a 3D point cloud has recently gained interest. In this paper, we present a new wearable device with a small-sized LiDAR that may be used to create an onboard human identification system for life-logging. Our proposed system starts with clustering to remove noise and background. Then fisher features are extracted from them. After that, the collected characteristics are utilized to train classifiers to identify the subjects. We conducted two different experiments to evaluate the suggested system. We collected six and thirteen subjects for each experiment. The results show that the proposed system can effectively remove noise and accurately identify subjects with at least 95% accuracy in both experiments.

Keywords: point cloud-based recognition, LiDAR, Human identification, user tracing for COVID-19

1. Introduction

In order to control the spread of COVID-19 by identifying infected patients and maintaining social distance, research institutes and industry are working on the development of new technologies that integrate human identification techniques with location information [1]. These technologies are also intended to provide useful information that aids in understanding the virus's transmission as well as individual actions to prevent the spread of infection via life-logging systems. Furthermore, in public closed places such as kinder gardens, schools, and universities, this technology is essential. Despite the popularity of proximity-based solutions such as Bluetooth, they are heavily dependent on the availability of Bluetooth-enabled cellphones, which are not always readily available, particularly in educational institutions. As an alternative, many studies have been conducted based on cutting-edge technology in human identification using biometric data such as the face [2] and voice [3]. Face recognition based on infrastructure cameras have been done using computer vision techniques. Due to the limited coverage of fixed cameras, this area of research limits the practicality of an aiming system. Voice-based person identification using the smartphone's internal microphone. However, this requires the user's speech to be recorded for identification, which may contain sensitive or private information. Unsurprisingly, the use of cameras or microphones in all locations, such as restrooms, has resulted in a lot of privacy concerns.

Using eye-safe lasers (class 1A, near-infrared spectrum), Light Detection and Ranging can identify nearby objects and compute distances to target objects with sub-decimeter errors. It is used for security monitoring, people counting, tracking in public space, path planning, marketing, and other applications in big indoor locations such as malls, museums, and government buildings. 3D point cloud data of 3D dimensional coordinates points captured by LiDARs does not contain any personal information.

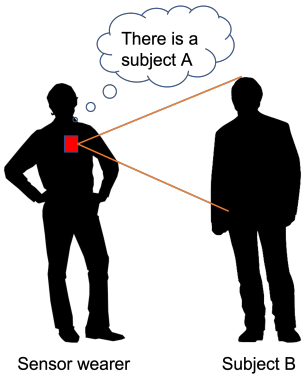


Fig. 1 The typical situation of our system usage

As illustrated in Fig. 1, we present a system for enabling person identification with an inexpensive wearable LiDAR in this paper. We offer the first small-sized LiDAR system capable of scanning and representing the surrounding environment in a privacy-preserving way. This representation can be used to identify the user who is contacting the sensor user without revealing any sensitive data like RGB images or voices. The suggested method, in particular, creates a signature database that can be used to identify the individuals. These signatures are derived using a computationally efficient fisher vector representation from each user's

[1] Grad. Sch. of Info. Sci. and Tech., Osaka University, Osaka, Japan
[a)] sho-yamada@ist.osaka-u.ac.jp
[b)] hamada_rizk@ist.osaka-u.ac.jp
[c)] h-yamagu@ist.osaka-u.ac.jp

point cloud representations. Then, to classify the user in question, a random forest classifier is trained.

Nonetheless, the proposed approach must prepare for a variety of obstacles of the point cloud. To begin with, unlike visual images, collecting meaningful context is a difficult task due to its unordered, unstructured, and variable size nature. Second, the proposed small-size LiDAR has restricted sensing performance in terms of range and noise. Finally, it is necessary to take into account the edge device's limited computational power. To address these challenges, we present a number of novel modules, including noise detection and removal over cascaded point cloud frames utilizing spatial-temporal density-based clustering. The fisher vector approach is used to extract fixed-size features by taking advantage of symmetric functions. After that, the collected features are used to build a computationally efficient classifier for person recognition.

In our lab, the proposed approach was tested on six and thirteen different subjects of students and employees. According to the results, the proposed method achieved a human identification accuracy of 92%. We also demonstrate the effectiveness of noise removal and processing time when numerous consecutive frames are taken into account. These findings establish the suggested system as a cutting-edge human identification system based on the point cloud.

This is how the rest of the paper is organized. In Section 2, we do a literature review. We give an overview of the proposed system in Section 3, and we go into details in Section 4. We discuss the data gathering procedure and how the system is tested in Section 5. Finally, in Section 6, we conclude the paper.

2. Related Work

The most relevant literature to the proposed system is discussed in this section.

2.1 Human Identification

Many studies for human-centric applications have been proposed, such as tracking [4–14] or identification [2, 3, 15]. The epidemic drew a lot of attention to human identification depending on a variety of human signatures. For example, face recognition [2], gait recognition [15], and voice recognition [3] can be used to identify humans. Face and voice recognition have demonstrated superior subject discrimination abilities. However, these strategies frequently raise privacy concerns, which make them difficult to implement in practice.

Face recognition systems began with the Eigenface [16] technique, which uses specific distribution assumptions to generate a lower-dimensional representation. However, this method fails to treat uncontrollable facial changes that deviate from their preconceived notions. Learning-based local descriptors were proposed [17, 18], but they couldn't guarantee robustness against facial changes. After AlexNet [19], which obtained the top accuracy in the ImageNet competition at the time, deep learning approaches became popular. On the LFW benchmark in 2014, DeepFace [20] achieved 97.35% accuracy, which is very similar to human performance (97.53%). Many academics then turned their attention to deep learning, which has so far attained state-of-the-art performance with an accuracy of up to 99.80%.

Many industries, such as computer vision, are interested in gait-based human recognition. This is accomplished by extracting features from the subject's body while a subject is moving. The system in [21] achieves this by using 3D convolutional neural networks to learn the gait from various viewing angles. A temporal-based graph LSTM network is used to learn a person's bones and joint attributes [22]. However, there are a number of difficulties with this area of research, including the variety of gait patterns and the difficulty of recognizing people in busy places.

Many strategies [23, 24] have been presented for human(speaker) recognition by speech, exploiting the strong capability of deep learning. The system in [23], for example, offered the concept of d-vector to improve the speaker's recognition. This is accomplished by training a model to extract features from which the d-vector can be computed, allowing for speaker recognition.

In contrast to vision-based or auditory approaches, the suggested system relies on a 3D point cloud to protect users' privacy. This is ideal for many situations when privacy is required, such as fitting rooms in stores..

2.2 Point cloud-based learning

LiDAR-based (i.e. 3D point cloud-based) applications have been increasingly popular in recent years in a variety of fields, including autonomous driving, archaeology, agriculture, and so on. LiDAR is used in autonomous driving to detect objects such as pedestrians and other vehicles. On top of cars, a high-quality LiDAR with a wide range and 360-degree angles scans the surrounding information, allowing for a deeper understanding of the surroundings. There are two ways for object detection. i.e., approaches based on region proposals and single-shot methods. Methods based on region proposals construct many regions containing objects, then extract features from each region to identify which class the objects belong to. These approaches can be implemented in three ways: *Multiview-based*, *Segmentation-based*, and *Frustum-based*. MultiView-based methods combine features from various view maps, such as a 2D image and a bird's eye view, for each proposal. Segmentation-based methods [25–27] divide points into foreground and background points using semantic segmentation techniques. After that, background points are deleted to reduce processing time, and only high-quality proposals for foreground points are generated. Frustum-based approaches [28] propose 2D candidate regions of objects using 2D object detectors and generate a 3D frustum proposal for each 2D candidate region. It's great for making suggestions on possible object locations.

Single-shot approaches [29] employ a single-stage network to predict class probabilities and produce 3D bounding boxes for objects. Because they don't have a step for region proposal, their processing speed is faster than region proposal-based approaches.

Deep learning-based approaches can be divided into two categories: indirect and direct [30]. In direct ways, point clouds are turned into regular structures such as multi-views [31] and voxel grids [32]. On the other hand, direct methods take advantage of raw point clouds. e.g., PointNet [33] uses raw point clouds as in-

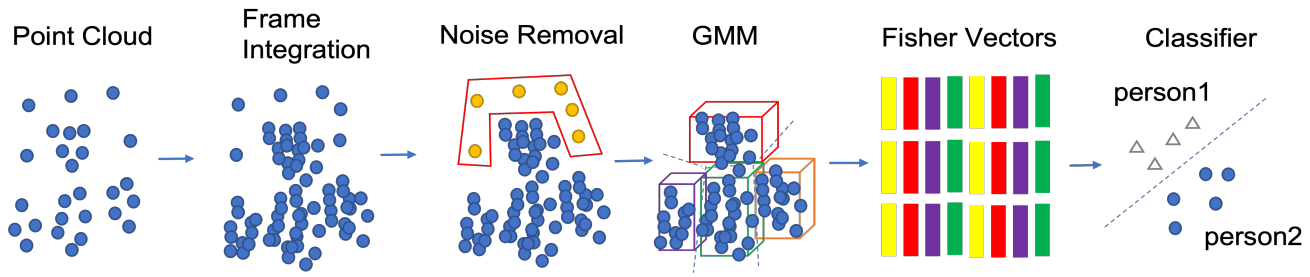
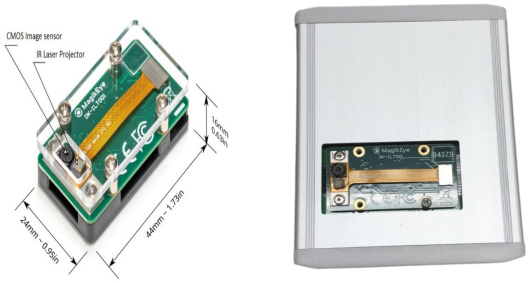


Fig. 2 System flow over the different modules.



(a) The small size LiDAR used at the core of the proposed system.

(b) Sensor and compute module.

Fig. 3 Devices of Hitonavi-μ.

puts for classification and segmentation and covers the common difficulties of point clouds. All of these solutions are demanding a lot of computing power, which makes them difficult to implement on edge devices.

Unlike other methodologies, this is the first work to consider the human identification-based 3D point cloud of LiDAR device. Furthermore, the suggested system is designed to be computationally efficient, making it suitable for use as an edge device on wearable LiRAR.

3. System Overview

The proposed system includes two stages: **the sensing stage** and **the learning stage**. The design of a completely new small LiDAR, which we call Hitonavi-μ, is required for the sensing step. The sensor is designed to be worn as a necklace (around the user's neck) and produces 3D point clouds for the environment. We gather point cloud data for several subjects while standing and walking. The learning stage's goal is to extract key characteristics (signatures) for each subject. The first step in this stage is to gather point cloud data for each subject, from which a signature database will be created. Then, we design a noise and background removal module using the Density-based clustering algorithm. This is accomplished by combining many successive scans to increase the point density, allowing for improved detection of forms, objects, and noise. Following that, discriminative features are recovered from the noise-free point cloud scans using the Fisher feature representation. The signature database for all subjects is the output of this stage. Finally, the Fisher feature is fed into a classifier to train.

4. The System Details

In this section, the sensing and data gathering step, as well as the learning stage, are described in detail. The system architecture is shown in Fig. 2.

4.1 The Sensor stage

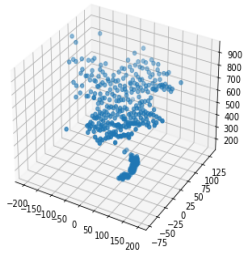
The properties of Hitonavi-μ sensor are described in this section. Hitonavi-μ is a miniature version of Light Detection and Ranging (LiDAR) with a battery and computing unit. LiDAR is a type of remote sensing technology that measures the distance between objects' surfaces. It calculates the distance by measuring the time it takes a laser pulse to travel from the sensor to the surrounding objects/surfaces and back. As a result, it is widely employed in a variety of applications, including detecting the presence of obstacles in autonomous vehicles [30] and indoor navigation for monitoring humans and robots [34]. All of these applications are based on large-scale LiDAR sensors with advanced sensing capabilities. As a result, its price is expensive, preventing widespread deployment as a privacy-preserving tool.

Therefore, we present a miniature version of the LiDAR sensor in this paper. The MagikEye firm produced this sensor (shown in Fig. 3(a)). In comparison to traditional LiDAR devices, this one is smaller and lighter. It is 44, 24, and 16 inches wide, deep, and tall, correspondingly. The goal of this design is to make it easier to use the sensor as a wearable device, which will pave the way for the next generation of privacy-preserving technologies. For processing our algorithms, the LiDAR sensor is connected to a processor unit of a battery-powered Raspberry Pi 4 Model B [35]. The Raspberry Pi 4 Model B is a small single-board computer with a 1.5GHz ARM Cortex-A72 quad-core CPU and 4GB RAM that can be utilized for mobile devices. In Fig. 3(b), Hitonavi-μ consists of a small-size sensor and a compute module. The sensor has around 30 frame per second in our environment.

4.2 The Data Collection

To create a signature database, we collect the 3D point clouds of various subjects. This is accomplished by using the sensor to capture the scene and sending the resulting point cloud to our server via the Hitonavi-μ's onboard WiFi module. The data is collected while the subjects are mobile(walking) and immobile (standing in conversation), aiming for the life-logging application. The sensor is worn around the subject's neck and on his chest. The data collection environment is depicted in Fig. 10. The sensor's typical use scenario is shown in Fig. 4(a), and its corresponding acquired point cloud is depicted in Fig. 4(b). As can be seen, only the x, y, and z coordinates are available for each cloud point, and there are some noises in the frame.

(a) Actual scene



(b) Point cloud as captured by the sensor

Fig. 4 Typical use case scenario.

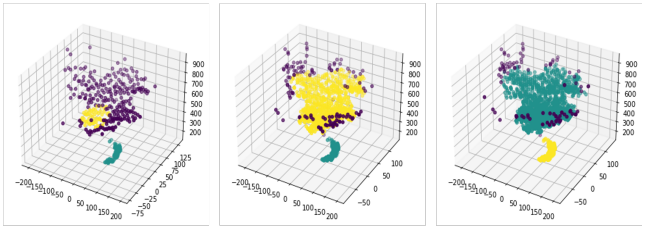


Fig. 5 The effect of the Noise Removal module.

4.3 The Learning Part

The learning stage is made up by the following three modules. Noise Removal, Feature Extraction, and Subject Identification.

4.3.1 Noise Removal

This module is designed to reduce noise and background that can degrade the quality of identification by deceiving the model. The Density-Based Spatial Clustering of Applications with Noise algorithm (DBSCAN) [36] is employed to detect and remove noise. DBSCAN has two benefits that prompted us to use it for noise reduction. First, it can function independently of domain knowledge, such as the number of clusters which K-Means [37] demands. Second, DBSCAN may also construct clusters with any forms and densities, unlike other clustering approaches. [38, 39].

DBSCAN is based on the notion that clusters are dense locations in space, such as the subject's arm, chest, and head, separated by lower density regions. DBSCAN can thus detect clusters by examining the local density of the points. *Eps* and *MinPts* are required parameters for DBSCAN. The radius of the circle to be generated around each point to check the density is indicated by *Eps*, and the minimum number of points necessary inside that circle for that point to be classed as a *Core* point is indicated by *MinPts*. DBSCAN constructs a *Eps* radius circle around each point and identifies it as a *Core* point, *Border* point, or *Noise* point. If the circle around a point has at least *MinPts* points, it is a *Core* point. If the number of points is less than *MinPts*, it is categorized as a *Border* point, and if there are no other points within a radius of *Eps*, it is classed as *Noise*.

The efficacy of DBSCAN is improved by taking into account the temporal effect across successive point cloud frames. To accomplish it, the points from n successive frames are added together to create an integrated frame. As a result, DBSCAN is used to cluster points in that integrated frame. The idea behind this technique is that by combining human and noise clusters, the density difference between them would increase, making noise detection and removal easier.

This module produces a number of clusters and labels one of the clusters whose size of points is the largest as 'human'. After that, an only 'human' cluster is kept while others are regarded as noise and removed.

Fig. 5 shows the performance of the noise removal module in three cases: single frame, integrating three frames and integrating five frames. The purple points represent noise while other colors represent clusters of the scene. In a single frame, DBSCAN does not work well because most of the points representing the person are classified as noise. The situation gets worse as the point of the noise are going to be classified as human because their size is the largest. In the case of three and five integrating frames, DBSCAN properly divides points into clusters and noises compared to the single frame case. Thus, the figure confirms that the more frames to integrate, the better noise detection is achieved. This can be justified as the integration increases the density of the main clusters of the subject while keeping the noise sparse at low density. This boosts the noise detection and clustering skills of DBSCAN.

4.3.2 Feature Extraction

We propose a feature extraction technique based on Fisher Vector (FV) representations in this section. The FV representation is a suitable choice for representing point cloud data since it is independent of sample sizes of point clouds. FV representation is used to define discriminative signatures of diverse subjects with varying sizes (as in 3D point cloud frames). It also represents the spatial locality of points implicitly. The deviation of 3D points from a generative model (e.g., Gaussian Mixture Model (GMM)) is characterized as this signature by calculating the gradients of the sample's log-likelihood with respect to the model parameters (i.e., weight, mean, and covariance). FV also has a fixed-size grid structure, which makes it a simple input to any classifier.

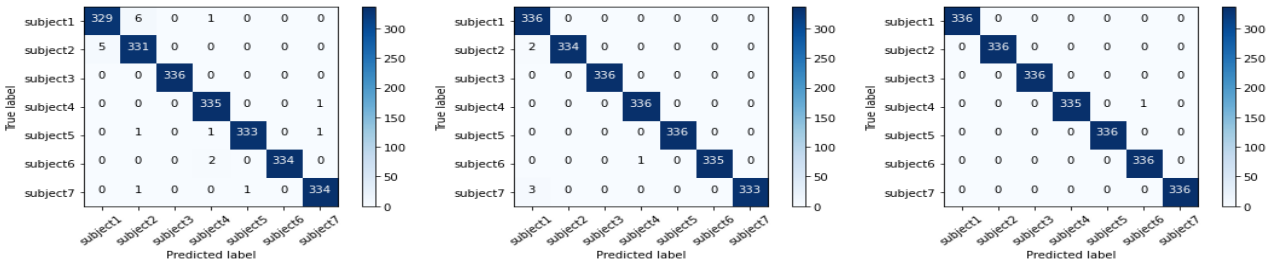
4.3.3 Subject Identification

In this section, we'll show how we create a classification model that can accurately distinguish between various subjects based on the Fisher representations. Random Forest Classifier is used in this case. The random forest has a number of decision trees; each one produces a class prediction, and the model with the most votes is chosen.

The motivation for employing Random Forest is the capacity to train a large number of generally uncorrelated models (trees) working as an ensemble of more accurate predictors than individual ones. Random Forest is also a flexible, easy-to-use supervised learning algorithm that, even without hyper-parameter adjustment, gives excellent results the of the time [40]. Because of its simplicity and versatility, it is also one of the most widely used algorithms. The features extracted by the feature extraction module are fed into the classifier as input. The best results were obtained with 10 trees using the split metric *entropy*. After the model has been trained, the extracted Fisher representation can be used to recognize the object in question.

5. Evaluation

We conducted a preliminary experiment(Experiment1) and a life-logging experiment(Experiment2) to evaluate the end-to-end performance of the proposed system. Experiment1 was con-



(a) 1 Frame (b) 3 Frames (c) 5 Frames

Fig. 6 Confusion matrices of experiment1.

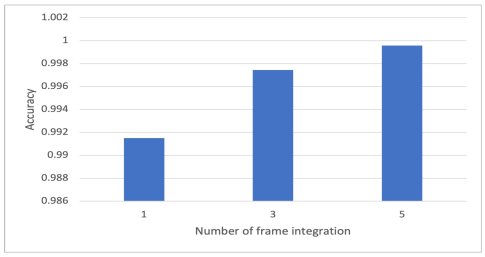


Fig. 7 Accuracy of the proposed system in experiment1.

ducted to test the device and our code. We collected data in a small number of subjects under loose conditions in which subjects are standing and other movements are not restricted. Experiment2 was conducted under conditions in which subjects could practice standing and walking activities. Regarding the walking activity, we collected eight different directions. Table 1 shows the number of subjects, activities, and data acquisition time of each experiment. We also need to mention that there are implicit hyperparameters that are different from the number of Frame integration and DBSCAN parameters(eps,minPts). e.g., the height of the sensor or the restriction of subjects' movement. In experiment2, we fixed these parameters. As for parameters of fisher features, we predefined 125 Gaussians with equal weights(0.008) and variance(0.05). Regarding the parameters of Random Forest, we set 10 trees and the entropy as a split metric.

5.1 Experiment 1

The experiment involves *six* subjects, including students and staff members in our lab. The distance between the worn sensor and the target subject is arbitrary and varies based on the subjects' preferences during a natural conversation. For each targeted subject, the point cloud signatures are captured over the course of *three* minutes. We used around 50 seconds-data and divide it into 80% training and 20% testing.

Fig. 7 shows the classification accuracy of the *six* subjects when changing the number of integrated frames. As can be seen, integrating more frames enhances the overall subject identification accuracy up to 99.9% in the case of integrating *five* consecutive frames.

Finally, we evaluate per subject identification accuracy at each case of integrated frames. Fig. 6 shows the confusion matrices of the *six* subjects when tested at each case of integrated frames. The confusion matrices show the gain earned by integrating more frames and the superiority of five-frames-based clustering. The results also confirm the validity of the proposed system for identifying different subjects even when a single frame-based clustering. This highlights the promise of the proposed system as the next generation of the privacy-preserving life-log system.

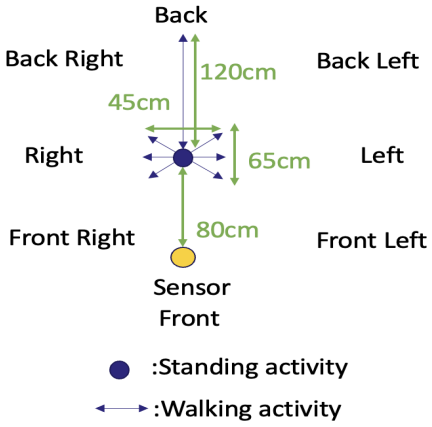


Fig. 8 Data collection environment.

5.2 Experiment 2

We collected standing and walking data for life-logging evaluation. The number of subjects is 13 people. The Bird's eye view of the data collection environment is shown in Fig. 8. Considering a real conversation, the distance between the sensor and subjects is around 80cm with a height of about 120cm(fits wearable scenario).

As for standing activity, the subjects stand still facing the direction of the sensor for a time interval of 1 minute. As for walking activity, we took data in eight different orientations to consider the conceivable circumstances. i.e., the subjects need to come in front of the sensor before starting a conversation and leave after the chat. Its patterns are somehow divided into eight (i.e., front, back, right, left, front right, front left, back right, back left). Each subject has his own way/pattern of mobility and interaction which can be captured from his walking speed and frequency of waving hands.

5.2.1 Preparing Integrated Frames

Fig. 9 shows an example of the way of preparing the integrated frames used in training and testing our model. Firstly, we collect 314 frames from raw point clouds. Then, we employ the windowing approach which considers a window of fifteen frames (1-15) then integrates them into one frame called "Data1". Then, the window is shifted one frame forward for integrating the next fifteen frames (2-16) to create "Data2". This process is repeatedly

Table 1 Information on experiments

Experiments	Number of subjects	Activity		Data acquisition time
Experiment1	6	Standing		3 mins
Experiment2	13	Walking	Right	1 mins
			Left	
			Front Right	
			Front Left	
			Back Right	
			Back Left	
			Front and Back	3 mins

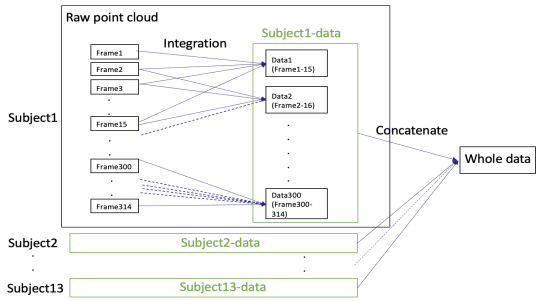


Fig. 9 Way of creating data for training and testing our model.

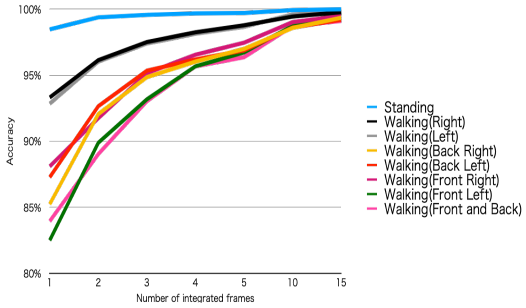
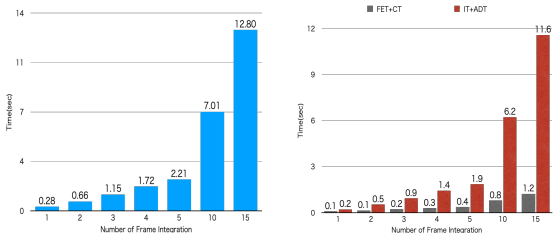


Fig. 11 Best accuracy when the number of frame integration changes.



(a) Total processing time of the proposed system

(b) Processing time for each process

Fig. 10

applied on all frames creating data of 300 integrated frames. This is how each subject's data are constructed and finally concatenates all of the subject's data. This whole data is split into 80% training and 20% testing.

5.2.2 Processing Time

Considering the usage of our system on the edge device, we measured the time it takes to process raw point cloud data and classify. i.e., the time for integrating frames and applying DBSCAN, feature extraction, and classification. Fig. 10(a) shows the time corresponding to each number of integrated frames. The more frames we integrate, the longer time it takes. We also measure the time by splitting processes into the former part and the latter part. The former part includes Integration Time of frames(IT) and Applying DBSCAN Time(ADT), while the latter part includes Feature Extraction Time(FET) and Classification Time(CT). Fig. 10(b) shows that the influential factor for processing time is the Frame Integration and DBSCAN processes.

5.2.3 Parameter Tuning

We select the optimal parameters that maximize our system performance per each activity. We set a range of parameters as follows. The number of frames to integrate in the range of [1,15], The eps in the range of [10,35], and minPts in the range of [5,30]. We tried to find the best combination of parameters among these values.

As we mentioned in the preliminary experiment section, the best combination of DBSCAN parameters(eps, minPts) depends on the number of integrated frames due to variation of density. Therefore, we analyzed classification accuracy in terms of the number of frame integration. The results are shown in Fig. 11. The more frames we integrate, the better accuracy we can get regardless of various activities. This is unexpected results for us because we first expected that the accuracy of mobile activities would drop as we integrate multiple frames due to the disappearance of human figures. The considered reason for the high accuracy of mobile activities is that the model can capture signatures over the time sequences by frame integration. i.e., the model can learn the subject's walking patterns which are unique to every person. In that case, the walking speed of subjects may affect the results.

If real-time processing is essential, we cannot integrate many frames, e.g., 15 frames, because it takes 10 seconds per single human identification estimate (See Fig. 10(a)). It is noteworthy that there is a trade-off between the processing time and the identification accuracy. Thus, the system designer can choose the number of integrated frames based on the demand of real-time processing. Since real-time processing is not crucial for the proposed life-logging application, it wouldn't be a problem to select a window of 15 frames for superior accuracy. However, even with only five frames case (faster by 9.7sec), the overall accuracy reaches 95.42%. Fig. 12 shows the confusion matrix of how the system performs per each subject.

6. Conclusion

We proposed a point cloud-based subject identification system for the purpose of life-logging using the first small-size wearable LiDAR. We presented the details of the system and its ability to detect and remove noise and extract discriminative features facilitating the accurate identification of different subjects. To achieve that, the proposed system leverages a combination of Spatio-temporal clustering for noise removal and a learned rep-

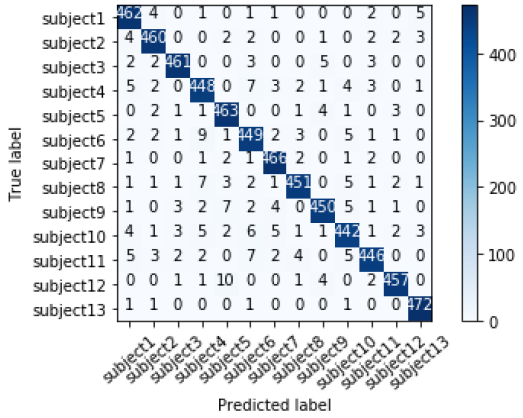


Fig. 12 Confusion matrix of experiment 2

resentation using the Fisher vector. This representation is further harnessed to train a random forest classifier for recognizing the user in the scene. We evaluated our system on two different experiments and achieved high accuracy in both experiments. The first preliminary experiment showed that our system accurately identifies subjects in a stationary case. In the second experiment, we tuned the parameters to maximize our system performance on both stationary and mobile cases. The results showed higher accuracy as we integrated multiple frames. We also measured the time it takes to identify subjects from about one second of data. The processing time increases as the number of integrated frames increases due to the DBSCAN process. We achieved 95.42% accuracy when we select 5 frame integration and tune DBSCAN parameters. Since processing time and accuracy are trade-offs, we can choose the number of integrated frames based on the purpose.

In the future, we plan to add an activity recognition for automatic adaptation of the system parameters. Secondly, we will work on the reduction of processing time of clustering. Thirdly, we aim to deploy our system on edge to measure how much energy consumes.

Acknowledgment

This work was supported by JST, A-STEP Grant Number JPMJTR20RV, Japan.

References

[1] Kleinman, R. A. and Merkel, C.: Digital contact tracing for COVID-19, *CMAJ*, Vol. 192, No. 24, pp. E653–E656 (2020).

[2] Zafaruddin, G. M. and Fadewar, H.: Face recognition using eigenfaces, *Computing, communication and signal processing*, Springer, pp. 855–864 (2019).

[3] Nidhyananthan, S. S., Muthugeetha, K. and Vallimayil, V.: Human recognition using voice print in labview [J], *International Journal of Applied Engineering Research*, Vol. 13, No. 10, pp. 8126–8130 (2018).

[4] Rizk, H., Yamaguchi, H., Higashino, T. and Youssef, M.: A Ubiquitous and Accurate Floor Estimation System Using Deep Representational Learning, *Proceedings of the 28th International Conference on Advances in Geographic Information Systems*, pp. 540–549 (2020).

[5] Rizk, H., Abbas, M. and Youssef, M.: Device-independent cellular-based indoor location tracking using deep learning, *Pervasive and Mobile Computing*, p. 101420 (2021).

[6] Alkiek, K., Othman, A., Rizk, H. and Youssef, M.: Deep Learning-based Floor Prediction Using Cell Network Information, *Proceedings of the 28th International Conference on Advances in Geographic Information Systems*, pp. 663–664 (2020).

[7] Fahmy, I., Ayman, S., Rizk, H. and Youssef, M.: MonoFi: Efficient Indoor Localization Based on Single Radio Source And Minimal Fingerprinting, *Proceedings of the 29th International Conference on Advances in Geographic Information Systems*, pp. 674–675 (2021).

[8] Rizk, H.: Solocell: Efficient indoor localization based on limited cell network information and minimal fingerprinting, *Proceedings of the 27th ACM SIGSPATIAL International Conference on Advances in Geographic Information Systems*, pp. 604–605 (2019).

[9] Rizk, H., Shokry, A. and Youssef, M.: Effectiveness of Data Augmentation in Cellular-based Localization Using Deep Learning, *Proceedings of the International Conference on Wireless Communications and Networking Conference (WCNC)*, IEEE (2019).

[10] Rizk, H.: Device-Invariant Cellular-Based Indoor Localization System Using Deep Learning, *The ACM MobiSys 2019 on Rising Stars Forum*, RisingStarsForum'19, ACM, pp. 19–23 (2019).

[11] Rizk, H. and Youssef, M.: MonoDCell: A Ubiquitous and Low-Overhead Deep Learning-based Indoor Localization with Limited Cellular Information, *Proceedings of the 27th ACM SIGSPATIAL International Conference on Advances in Geographic Information Systems*, ACM, pp. 109–118 (2019).

[12] Rizk, H., Amano, T., Yamaguchi, H. and Youssef, M.: Smartwatch-based Face-touch Prediction Using Deep Representational Learning, *Proceedings of the 18th EAI International Conference on Mobile and Ubiquitous Systems: Computing, Networking and Services*, EAI, Springer (2021).

[13] Rizk, H., Abbas, M. and Youssef, M.: OmniCells: Cross-Device Cellular-based Indoor Location Tracking Using Deep Neural Networks, *the International Conference on Pervasive Computing and Communications (PerCom)*, IEEE (2020).

[14] Rizk, H., Torki, M. and Youssef, M.: CellinDeep: Robust and Accurate Cellular-based Indoor Localization via Deep Learning, *IEEE Sensors Journal* (2018).

[15] Yoo, J.-H., Hwang, D., Moon, K.-Y. and Nixon, M. S.: Automated Human Recognition by Gait using Neural Network, *2008 First Workshops on Image Processing Theory, Tools and Applications*, pp. 1–6 (online), DOI: 10.1109/IPTA.2008.4743792 (2008).

[16] Turk, M. and Pentland, A.: Eigenfaces for recognition, *Journal of cognitive neuroscience*, Vol. 3, No. 1, pp. 71–86 (1991).

[17] Cao, Z., Yin, Q., Tang, X. and Sun, J.: Face recognition with learning-based descriptor, *2010 IEEE Computer society conference on computer vision and pattern recognition*, IEEE, pp. 2707–2714 (2010).

[18] Lei, Z., Pietikäinen, M. and Li, S. Z.: Learning discriminant face descriptor, *IEEE Transactions on pattern analysis and machine intelligence*, Vol. 36, No. 2, pp. 289–302 (2013).

[19] Krizhevsky, A., Sutskever, I. and Hinton, G. E.: Imagenet classification with deep convolutional neural networks, *Advances in neural information processing systems*, Vol. 25, pp. 1097–1105 (2012).

[20] Taigman, Y., Yang, M., Ranzato, M. and Wolf, L.: Deepface: Closing the gap to human-level performance in face verification, *the IEEE conference on computer vision and pattern recognition*, pp. 1701–1708 (2014).

[21] Wolf, T., Babaee, M. and Rigoll, G.: Multi-view gait recognition using 3D convolutional neural networks, *2016 IEEE International Conference on Image Processing (ICIP)*, IEEE, pp. 4165–4169 (2016).

[22] Battistone, F. and Petrosino, A.: TGLSTM: A time based graph deep learning approach to gait recognition, *Pattern Recognition Letters*, Vol. 126, pp. 132–138 (online), DOI: https://doi.org/10.1016/j.patrec.2018.05.004 (2019). Robustness, Security and Regulation Aspects in Current Biometric Systems.

[23] Variani, E., Lei, X., McDermott, E., Moreno, I. L. and Gonzalez-Dominguez, J.: Deep neural networks for small footprint text-dependent speaker verification, *the international conference on acoustics, speech and signal processing (ICASSP)*, IEEE, pp. 4052–4056 (2014).

[24] Ravanelli, M. and Bengio, Y.: Learning speaker representations with mutual information, *arXiv preprint arXiv:1812.00271* (2018).

[25] Yang, Z., Sun, Y., Liu, S., Shen, X. and Jia, J.: Ipod: Intensive point-based object detector for point cloud, *arXiv preprint arXiv:1812.05276* (2018).

[26] Shi, S., Wang, X. and Li, H.: Pointrcnn: 3d object proposal generation and detection from point cloud, *the IEEE/CVF conference on computer vision and pattern recognition*, pp. 770–779 (2019).

[27] Yang, Z., Sun, Y., Liu, S., Shen, X. and Jia, J.: STD: Sparse-to-Dense 3D Object Detector for Point Cloud, *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)* (2019).

[28] Qi, C. R., Liu, W., Wu, C., Su, H. and Guibas, L. J.: Frustum pointnets for 3d object detection from rgb-d data, *the IEEE conference on computer vision and pattern recognition*, pp. 918–927 (2018).

[29] Yang, Z., Sun, Y., Liu, S. and Jia, J.: 3dssd: Point-based 3d single stage object detector, *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 11040–11048 (2020).

[30] Zhang, J., Zhao, X., Chen, Z. and Lu, Z.: A review of deep learning-based semantic segmentation for point cloud, *IEEE Access*, Vol. 7, pp. 179118–179133 (2019).
[31] Su, H., Maji, S., Kalogerakis, E. and Learned-Miller, E.: Multi-view convolutional neural networks for 3d shape recognition, *the IEEE international conference on computer vision*, pp. 945–953 (2015).
[32] Maturana, D. and Scherer, S.: Voxnet: A 3d convolutional neural network for real-time object recognition, *the International Conference on Intelligent Robots and Systems (IROS)*, IEEE, pp. 922–928 (2015).
[33] Qi, C. R., Su, H., Mo, K. and Guibas, L. J.: Pointnet: Deep learning on point sets for 3d classification and segmentation, *the IEEE conference on computer vision and pattern recognition*, pp. 652–660 (2017).
[34] Rizk, H., Yamaguchi, H., Youssef, M. and Higashino, T.: Gain without pain: Enabling fingerprinting-based indoor localization using tracking scanners, *Proceedings of the 28th International Conference on Advances in Geographic Information Systems*, pp. 550–559 (2020).
[35] : Raspberry Pi Foundation, https://www.raspberrypi.com/products/raspberry-pi-4-model-b/. Accessed: 2021/11/30.
[36] Ester, M., Kriegel, H.-P., Sander, J., Xu, X. et al.: A density-based algorithm for discovering clusters in large spatial databases with noise., *kdd*, Vol. 96, No. 34, pp. 226–231 (1996).
[37] Hartigan, J. A. and Wong, M. A.: Algorithm AS 136: A k-means clustering algorithm, *Journal of the royal statistical society. series c (applied statistics)*, Vol. 28, No. 1, pp. 100–108 (1979).
[38] Rizk, H., Elgokhy, S. and Sarhan, A.: A hybrid outlier detection algorithm based on partitioning clustering and density measures, *2015 Tenth International Conference on Computer Engineering & Systems (ICCES)*, IEEE, pp. 175–181 (2015).
[39] Elmogy, A., Rizk, H. and Sarhan, A. M.: OFCOD: On the Fly Clustering Based Outlier Detection Framework, *Data*, Vol. 6, No. 1, p. 1 (2021).
[40] Erdélyi, V., Rizk, H., Yamaguchi, H. and Higashino, T.: Learn to See: A Microwave-based Object Recognition System Using Learning Techniques, *Adjunct Proceedings of the 2021 International Conference on Distributed Computing and Networking*, pp. 145–150 (2021).