International Journal of Smart Computing and Artificial Intelligence
International Institute of Applied Informatics
2022, Vol. 6, No. 1, IJSCAI684

Realization for Finger Character Recognition Method by Similarity Measure of Finger Features

Takuma Nitta *, Shinpei Hagimoto *, Ari Yanase *,
Ryotaro Okada *, Virach Sornlertlamvanich *,
Takafumi Nakanishi *

Abstract

In this paper, we present a novel finger character recognition method in sign language using dimension reduction finger character feature knowledge base for similarity measure. A sign language communication is crucial method for deaf or hearing-impaired people. One of the most important problems is that very few people can understand a sign language. Essentially, there is not enough image data set for finger character learning. In addition to aligning a corpus of images of finger character, *it is necessary to realize an automatic recognition system for finger characters in a sign language. We construct a knowledge base for finger character features and apply it to realize a novel finger character recognition. Our method enables finger character recognition by similarity measure between the input finger character features and a knowledge base. The experimental results show that our approach efficiently utilizes the knowledge base generated from a small amount of finger character images. We also present our prototype system and experimental evaluation.

Keywords: finger character knowledge base, finger character recognition, sign language, similarity measure

1 Introduction

We use various languages to realize communication in the world. Millions of people around the world are handicapped by deafness and hearing loss. Sign language is one of the most necessary languages for communication. There is a communication divide between signers and speakers of other languages. One of the important issues is to solve the communication divide and realize a new global communication tool in our life. One of the possible ways to support the communication for deaf or hearing-impaired people is to realize an automatic sign language translation system. By realizing the system, we can construct an environment that facilitates social advancement and interaction with them.

In general, it is necessary to prepare a large corpus of training data to implement an automated recognition system. A sign language varies from country to country and region to region, it is

* Musashino University, Tokyo, Japan

costly to prepare such a large data set. Especially, there are not enough image data for finger character learning in the case of Japanese sign language. In order to realize a finger character recognition system for Japanese sign language, it is inevitable to construct a Japanese sign language finger character knowledge base and to design an effective appropriate approach using the knowledge base constructed from a small image data set as its training data.

In this paper, we present a novel finger character recognition method in sign language by similarity measure based on the finger character feature knowledge base. Our method consists of an extraction module for skeletal coordinates of the body for finger characters, similarity measure module with the extracted finger character features, and the knowledge base stored the reference finger character features. We construct a knowledge base for finger character features and apply it to realize a novel finger character recognition. Our method enables finger character recognition by similarity measure between the input data and the knowledge base consisting of a small amount of training finger character data.

The main features of this paper are as follows.

- We construct a finger character knowledge base consisting of pairs of reference finger features and their corresponding characters.
- We develop our prototype system for recognizing sign language by similarity measure using a finger character feature knowledge base. We also conduct some evaluation experiments on our prototype system.

This paper is organized as follows. In Section 2, the related works are discussed. In Section 3, we present our proposed method—a finger character recognition method by applying similarity measure using dimension reduction finger character feature knowledge base for similarity measure. In Section 4, we present our prototype system and describe the experimental results. Finally, in Section 5, we summarize the approach and the evaluation results.

2 Related Works

Essential components for finger character recognition consists of feature extraction and feature processing for character identification. In this section, we describe the existing methods of each component.

There are two main types of methods for extracting the features of finger characters as follows:

1. Directly extraction method for the features of finger characters using a data glove with multiple sensors
2. Indirect extraction method for the features of finger characters using devices equipped with depth sensors or monocular cameras

The first type is a method of directly extracting the characteristics of finger characters using a data glove with multiple sensors [1][2]. The glove can accurately extract changes in the distance or opening angle between each finger, and the coordinate information of the X-Y-Z axis due to the movement of finger characters from multiple sensors. While accurately extracted data greatly contributes to improving the accuracy of sign language or finger character recognition, it is inconvenient for users who perform sign language or finger characters to wear the gloves, considering people's daily lives.

The second type is a method of indirectly extracting the characteristics of finger characters using devices equipped with depth sensors or monocular cameras [3]. Depth sensors can extract changes of coordinate information of the X-Y-Z axis of the body skeleton including hands and fingers due to the movement of finger characters. Since depth sensors can extract not only plane coordinates but also depth coordinates, it can extract features more accurately than monocular cameras for sign language or finger characters that involves three-dimensional movements. Cameras can capture the motion of sign language or finger characters and get RGB images or videos. There are roughly four methods for extracting features from RGB image or video data. The first one is extracting the ratio of the skin area hand or finger in particular area, the ratio of the length within the area, and the shape of the skin area as features using the color information of the images [4]. It is difficult to recognize due to the influence of hand shape, skin color, background, ambient light. The second one is dividing a video into image frames by frames for image recognition. The third one is extracting vectors of object motion between adjacent frames from a video using an Optical Flow algorithm [5][6]. The fourth one is extracting knuckle coordinates or joint angles from video or RGB images using a deep learning model that extracts skeletal coordinates [6][7].

We focus on finger features such as finger joint coordinates, finger joint angles, and vectors obtained by Optical Flow algorithms to realize an automatic recognition system of finger character in sign language. The methods to realize the recognition function using the above features are using Long Short Term Memory (LSTM) [8][9], Hidden Markov Models (HMM) [10], and Dynamic Time Warping (DTW) [11], or by classifying them using machine learning algorithms such as Support Vector Machine (SVM) [12]. In the case of features such as the image itself or the shape of the skin region of a hand obtained by image processing, image recognition algorithms such as Convolutional Neural Network (CNN) [6][13] are often used to identify and estimate characters. These methods require a huge amount of image data sets. There are other recognition methods using similarity matching [14]. Our method enables finger character recognition by a similarity measure with a knowledge base consisting of a small amount of training finger character data extracted by combining the camera and the deep learning model to extract the skeletal coordinates.

The one of the important methods for construction of a knowledge base and representation of multimedia data as vectors for contextual semantic measure as unique distance measure is Mathematical Model of Meaning [15][16].

Our method constructs a knowledge base of extracted finger features and measures their similarity to the target finger features to realize finger character recognition.

3 Finger Character Recognition Method by Applying Similarity Measure

In this section, we present our proposed method for finger character recognition by applying similarity measure.

3.1 Overview

Our method extracts finger character features by using a general-purpose device, a monocular camera and a deep learning model to extract skeletal coordinates and identify the finger characters by performing a similarity measure between the input finger character features and a knowledge base consisting of reference finger character features. In order to reduce the computational cost and improve the recognition accuracy, character switching detection and dimension reduction are performed. In addition, the method ensures robustness in employing finger joint coordinates as a feature by normalizing the differences in the size of the hands and fingers in the frame by distance from the camera on a video-by-video basis and the differences in finger joint coordinates on a video-by-video basis by the position of the hands and fingers in the frame.

The overview of our proposed method is shown in Figure 1. Our method consists of two modules, feature extraction and processing part for construction of a knowledge base and the recognition system part. The former module further consists of three modules, the annotation part, the feature extraction part, and the knowledge base construction part. The latter module further consists of three modules, a feature extraction part, a data screening part, and a similarity measure part.

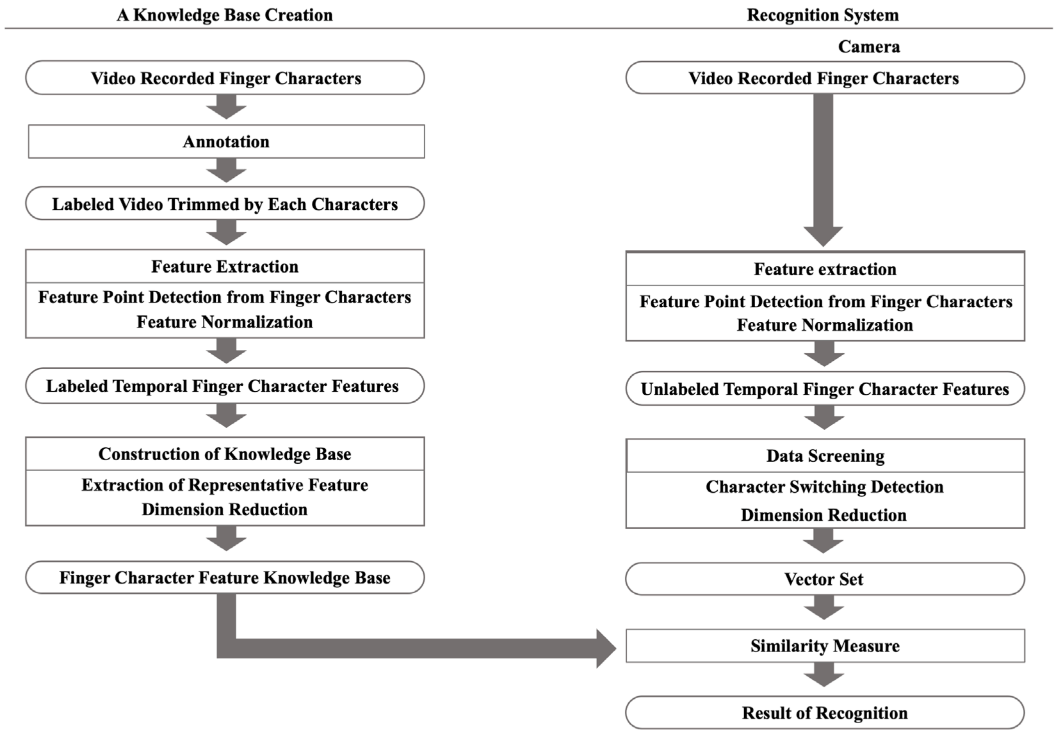


Figure 1: Overview of our proposed method. Our method consists of two modules, construction of a knowledge base and the recognition system. The construction of a knowledge base consists of annotation, feature extraction, extraction of representative feature and dimension reduction. The recognition system consists of feature extraction, data screening, and similarity measure.

3.2 Feature Extraction and Processing for Construction of a Knowledge Base

In this subsection, we present feature extraction and processing for construction of a finger character knowledge base. In Subsection 3.2.1, we present the annotation of videos recorded finger characters for creation of data set. In Subsection 3.2.2, we present the finger features extraction method including feature point detection from finger characters, missing value completion, feature normalization. In Subsection 3.2.3, we present the construction of a finger character feature knowledge base.

3.2.1 Annotation for Dataset Construction

In order to prepare the data set needed to construct the finger character feature knowledge base, the videos recorded finger characters are trimmed and annotated to one character at a time.

3.2.2 Finger Feature Extraction

In this subsection, we describe how to detect feature points from the user's finger characters and how to extract the features to recognize the finger characters.

3.2.2.1 Feature Point Detection from Finger Characters

The feature points detection from finger characters is performed indirectly through a combination of a general-purpose device, a monocular camera, and a deep learning model that extracts joint coordinates from the skeleton. We apply MediaPipe Hands [17] to detect joint coordinates for deep learning model. MediaPipe Hands can extract the coordinates of 21 of the each hand. Joints extracted from MediaPipe Hands and assigning numbers to joints are shown in Figure 2. Finger character features are 42-dimensional data using the X and Y axis coordinates of 21 joint of the right hand extracted from MediaPipe Hands.

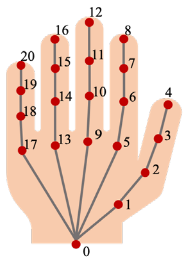


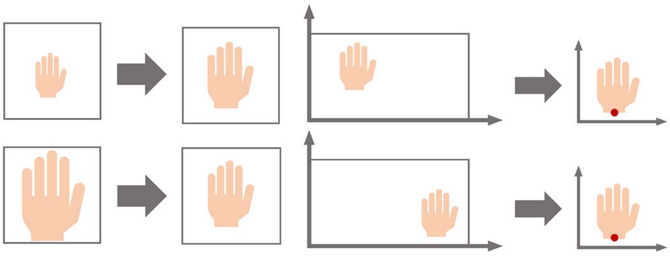
Figure 2: The right hand skeleton joints and numbers employed our proposed method. The coordinates of these joints are extracted by MediaPipe Hands. Each joint is represented as "key00, key01, … , key20".

3.2.2.2 Missing Value Completion

Deep learning models that extract skeletal joint coordinates such as MediaPipe Hands sometimes fail to extract the coordinates. Since words and sentences are expressed using a continuous character, it is necessary to perform missing value completion to eliminate missing values before processing the data in a sequence of time. Missing value completion is performed by replacing the value of the frame containing the missing value with the value of the previous frame without the missing value.

3.2.2.3 Feature Normalization

One of the difficult tasks in a finger character recognition is the difference in scale and position of hand taken in a video due to the camera and hand relative position. These differences cause the hand coordinates obtained by MediaPipe Hands to be completely different in each video, even though the same finger characters are shown in the video. In order to correct the differences for improving the accuracy of our proposed finger character recognition, scale normalization and positional corrections are performed as shown in Figure 3. In this subsection, we present detail of feature normalization method.



(a) (b)

Figure 3: The overview of normalization. Figure (a) shows scale normalization. Whether the hand in the video is small or large, the size of the hand is normalized according to the standard scale. Figure (b) shows coordinate position correction. Wherever the hand is positioned in the video, the position of the hand is normalized according to the standard coordinates.

The hand joint coordinates extracted by MediaPipe Hands are normalized by the following formula (1) basic on the base absolute position and scale ratio.

$$\vec{J}' = \frac{\vec{J} - \overrightarrow{(key00_x,\ key00_y)}}{\sqrt{(key01_x - key00_x)^2 + (key01_y - key00_y)^2}} * S + \vec{V} \quad (1)$$

Where $\vec{J}$ is the joint coordinates, $\vec{J}'$ is the normalized joint coordinates, key00x, key00y, key01x and key01y are the coordinates of key point 00 and 01, and $\vec{V}$ is the base vector for position correction.

Any coordinates of the key point can be applied to the absolute position, by calculating the scaling ratio from the distance between the wrist joint (key point 00) and the base of the first joint (key point 01). The base ratio is set so that the distance between the wrist joint and the thumb joint is always constant. The coordinates and ratios are normalized, but it makes each coordinates dense and the motion is captured small, so they are expanded to a constant size with S. In this paper, we set the value of S to 1000. The coordinates after 00 of the origin, 01 to 20, are scaled one by one based on the formula presented in (1) which using key point 00 and 01 values. The position of each coordinate is corrected relative to the position of the wrist joint as the origin.

3.2.3 Construction of Knowledge Base

Our system performs similarity measure between the input finger character data and the knowledge base. The knowledge base consists of a matrix of 41 static finger characters without movement in 46 Japanese finger characters and features obtained by dimension reduction of the 42-dimensional finger character features extracted from each character. In this subsection, we describe a method for extracting the reference finger character features required to construct a knowledge base, and dimension reduction targeting finger character features to reduce computational costs and improve accuracy.

3.2.3.1 Extraction of Representative Features

We evaluated the accuracy of several methods to construct a better knowledge base in our previous work [18]. This paper is based on the adoption of skeletal features when there is higher reliability in the estimation of each coordinate in the detection of 21 skeletal coordinates. In MediaPipe Hands, skeletal features are extracted only when the reliability of the detected coordinates is higher than a threshold value, so the reliability can be used to construct a knowledge base for improving accuracy, as shown in our previous work [18]. The method that had the best accuracy in this paper was to select the features at all times when the average of the reliability of each of the 42 features was greater than a threshold value, calculate the average value for each dimension of the 42-dimensional features, and use the set of values as the new 42-dimensional features. This new feature is defined as a representative feature in each finger character and stored in the knowledge base as a representative feature for each character.

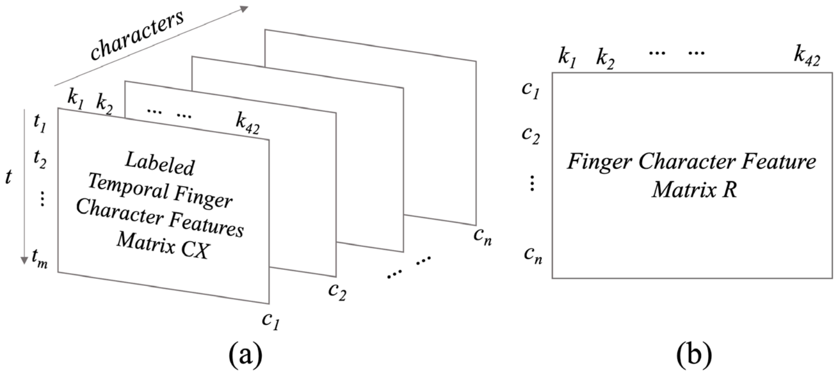


Figure 4: Data structures generated in the pre-processing for the construction of the knowledge base. Figure (a) shows a matrix CX consisting of temporal finger character features annotated per character; Figure (b) shows a matrix R consisting of reference finger character features for each character.

3.2.3.2 Dimension Reduction

Performing a similarity measure between the 42-dimensional vector of input finger character features and the 42-dimensional vector of finger character features in the knowledge base (Figure 5) would be computationally expensive. In addition, even if there are 42-dimensional features, some features may be uncertain. Therefore, dimension reduction is performed on a knowledge base containing 42-dimensional features to reduce the computational cost and to improve the accuracy of similarity measure by highlighting the features (Figure 6 (a), (b)). Although the computational cost of the dimension reduction algorithm is not low, it contributes to a significant computational cost reduction since it is done once, and subsequent similarity measures are performed by lower-dimensional features.

There are various methods for dimension reduction, but in this paper, the method with the best accuracy is adopted as our proposed method according to the accuracy comparison among three methods, Latent Semantic Analysis (LSA), Principal Component Analysis (PCA), and Independent Component Analysis (ICA) [19] in 4.3 Experiment-2 (Optimal Dimension Reduction Method and Number of Dimensions) of Chapter 4.

Latent Semantic Analysis (LSA) is a dimension reduction method that uses Singular Value Decomposition (SVD). Latent Semantic Analysis (LSA) represents the original data as much as possible while reducing the computational cost by truncating the singular values obtained by singular value decomposition from a certain standard. Principal Component Analysis (PCA) represents high-dimensional data in low-dimensional data with minimal loss of information by using the dependencies between variables. The data is represented by a new basis of principal components that are uncorrelated and have maximum variance. Independent Component Analysis (ICA) performs dimension reduction to search for the independent component that minimizes the statistical dependence between components by linear transformation.

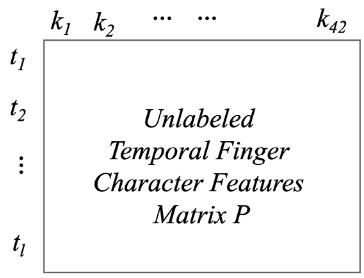


Figure 5: Data structure of the matrix *P*. This matrix is consisting of unlabeled temporal finger character features.

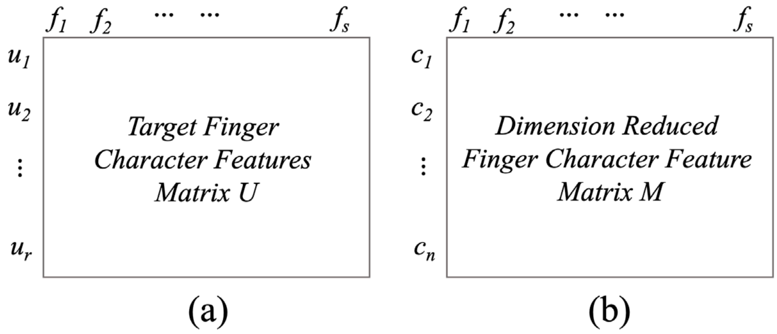


Figure 6: Data structures of matrices generated by data screening. Figure (a) shows a matrix *U* consisting of dimensions-reduced finger character features in both the row and column directions. Figure (b) shows matrix *M* consisting of dimension reduced finger character features matrix.

3.2.3.3 Several Models in Knowledge Base

In our method, a knowledge base is provided in order to ensure robustness to individual differences in finger characters. The knowledge base contains several models of each finger character. A knowledge base constructed from features extracted from finger characters performed by a single person is prepared for multiple people, which allows for similarity measure for individual differences. Each matrix $M_1, M_2, \dots, M_\eta$ is a dimension reduced finger character feature matrix (Figure. 6 (b)). The knowledge base *S* consists of the matrices as shown in (2).

$$S = \{M_1, M_2, \dots M_\eta\} \qquad (2)$$

3.3 Recognition System

In this subsection, we mainly describe a series of processes for finger character recognition, the data screening method and similarity measure for the realization of the finger character recognition system.

3.3.1 Feature Extraction

The method has described in Subsection 3.2.2.

This method allows us to obtain a matrix *P* consisting of unlabeled and temporally varying finger joint features $(k_1, k_2, \ldots, k_{42})$.

3.3.2 Data Screening

In order to reduce computational cost and improve recognition accuracy, we reduce the number of frames by character switching detection in the row direction of matrix *P*, and reduce the number of feature dimensions by applying dimension reduction algorithms in the column direction of matrix *P*. The matrix obtained by data screening for matrix *P* is the matrix *U*. Similarly, the matrix *M* is a matrix whose dimensions are reduced in the column direction for matrix *R*.

3.3.2.1 Character Switching Detection

It is extremely computationally expensive to perform a similarity measure against the knowledge base for all the input data extracted from 30-60 frames per second of video. In addition, it is meaningless to perform similarity measure using features extracted from finger shapes during the process of switching from one finger character to another. By performing the character switching detection and adopting features of only a few frames when each character is shown for later processing, significant data can be reduced. Furthermore, extracting the data when the hand movement is stable while indicating one character can contribute to improve the accuracy of character recognition obtained from a result of similarity measure. The character switching detection is carried out by using the formula shown in (5) to capture the large changes in the X or Y coordinates of each joint, or both, that occur when the hand shape is switched to indicate a finger character.

$$A_t = \sum_{i=0}^{20} [\{key(t,i,x) - key(t-1,i,x)\}^2 + \{key(t,i,y) - key(t-1,i,y)\}^2] \quad (3)$$

$$A_{t-1} = \sum_{i=0}^{20} [\{key(t-1,i,x) - key(t-2,i,x)\}^2 + \{key(t-1,i,y) - key(t-2,i,y)\}^2] \quad (4)$$

$$\frac{A_t - A_{t-1}}{A_{t-1}} \quad (5)$$

In the function $key(t, i, [x, y])$ contained in the formulas (3) and (4), the first argument represents the time, the second argument represents the joint number indicating the joint ID, and the third argument represents whether the x or y coordinate is used. The joint number *i* ranges from

0 to 20. Three arguments are given to the function key, and the coordinates at the time of those parameters are returned.

A_t in formula (3) is the calculated sum of the squares of the differences between the coordinates of each of the 21 joints of the hand in the current frame and one previous frame at time t. Similarly, A_{t-1} in formula (4) is the calculated sum of the squares of the differences between the coordinates of each of the 21 joints of the hand in the current frame and one previous frame at time t-1. The formula (5) calculates the change rate of the sum of the squares of the differences of the coordinates of each of the 21 joints of the hand at each of the times calculated in equations (3) and (4). The value is defined as the degree of change in the shape of the finger, like the degree of anomaly in anomaly detection. In the time series of changes in the value indicating the degree of change in the state of the finger, if no significant change is shown for several frames after the last time the ratio of change exceeded the threshold, then the last time the ratio of change exceeded the threshold is defined as the character switching point. The matrix *P'* is the matrix obtained by adopting the character switching detection in the row direction for matrix *P*.

3.3.2.2 Dimension Reduction

The method has described in Subsection 3.2.3.2. The matrix *M* is the matrix obtained by performing dimension reduction on the matrix *R* in the column direction. The matrix *U* is obtained by reducing the dimension of the matrix *P'* in the column direction, using the model when matrix *R* is reduced to a lower dimension. The relationship among the matrices transformed by dimension reduction is shown in Figure 7.

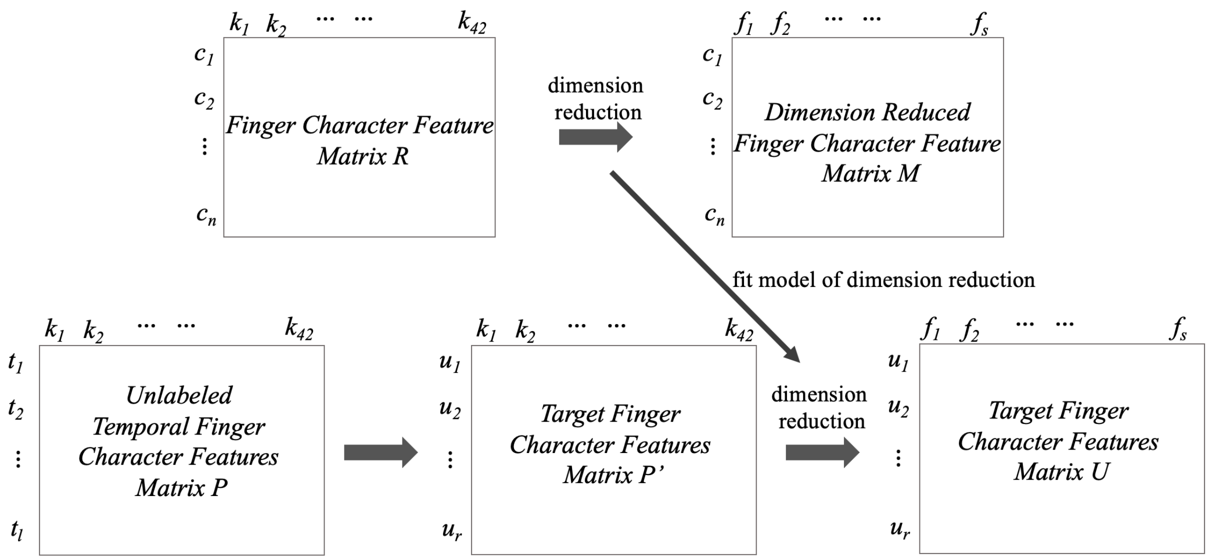


Figure 7: The relationship among the matrices transformed by dimension reduction.

3.3.3 Similarity Measure

Similarity measure is performed between the knowledge base *S* and the vector u_i *(i=1, 2, …, r)* which representing finger character features obtained by data screening. Similarity measure function sim is shown in (6):

$$v_i = sim(u_i, S) = maxsim\left(recog(u_i, M_1), \ldots, recog(u_i, M_\eta)\right) \quad (6)$$

Function $recog(u_i, M_j)$ returns the character corresponding to the highest similarity between the vector u_i and each row corresponding to a character in the matrix M_j. Function $maxsim$ returns the character corresponding to the highest similarity among $recog(u_i, M_j)$. As a result of the similarity measure, the model of the finger character vectors with the highest similarity (v_i) is determined as the finger characters indicated by the user in the frame on which the input was based.

4 Experiment and Evaluation

4.1 Experiment Environment

All the experiments are performed in the environment shown in Table 1.

Table 1: Environment of experiment

OS	Mac OS Catalina
CPU	2.3GHz Quad-Core Intel Core i5
RAM	16GB
Camera	Monocular camera on iPad Pro 2nd generation

In order to conduct the experiment, we first collected finger character videos and images. We collected 41 static finger characters images and three videos. The videos are collected from the one available on the Internet. As validation data, each finger character shown by 16 subjects was recorded and used. We extracted 41 short videos trimmed down to 1 second from each collected video. Each short video corresponding to a finger character. Since feature extraction by MediaPipe Hands from the videos is done frame by frame, we can get about 30 features per second for a 30 fps video.

4.2 Experiment-1 (Validity of Normalization)

We verify that the feature normalization method and formula (1) described and shown in 3.2.2.3 are valid.

In order to verify the validity of scaling in normalization, prepare two images of the same finger character of the same person, with a small hand in the center and a large hand in the center. We check whether the vertical and horizontal sizes of the hands in the two images are the same.

In order to verify the validity of position correction in normalization, prepare two images of the same finger character of the same person, one with the hand in the upper left corner and the other with the hand in the lower right corner. We check whether the origin of the reference coordinates of the hands in the two images are the same.

Figure 7 shows the before and after normalization of scaling and position correction.

As shown in (a) of Figure 8, before scaling, the vertical and horizontal sizes of the hands were different. Although position correction was not performed to verify the validity of scaling and there is a gap in the coordinates, after scaling, the vertical and horizontal sizes of the hands are constant, thus proving that the scaling is valid.

As shown in (b) of Figure 8, the positions of the hands were different before the position correction, but after the position correction, the positions of joint number 0 marking the wrist are constant, thus proving that the position correction is valid.

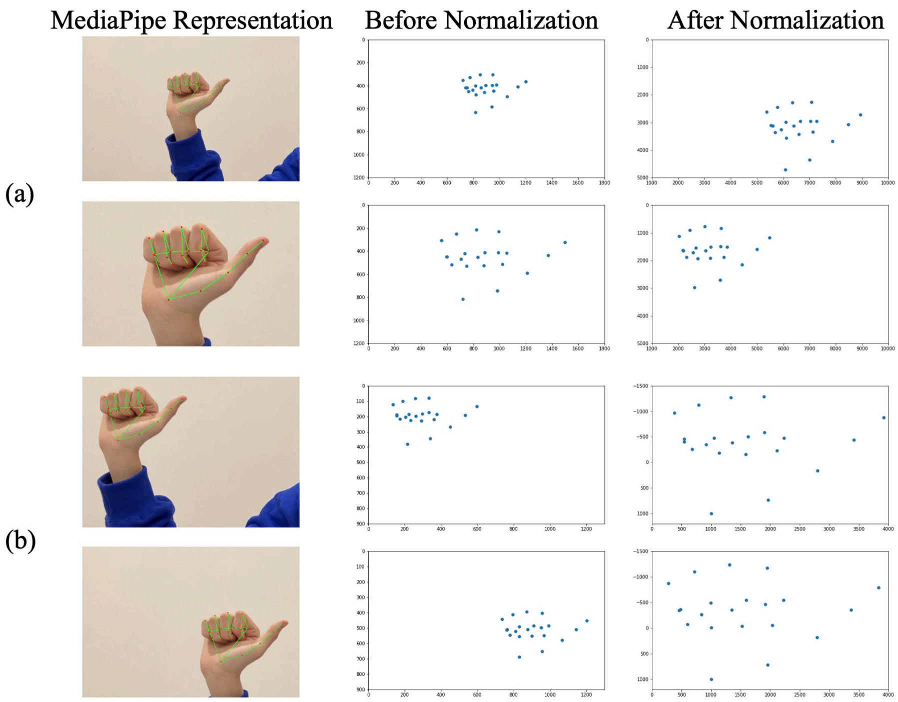


Figure 8: Before and after scaling and position correction in normalization.

4.3 Experiment-2 (Optimal Dimension Reduction Method and Number of Dimensions)

In order to construct matrix M by dimension reduction of matrix U, which is created in the process of construction of the knowledge base, we experimented to find out what the optimal dimension reduction method and the optimal number of dimensions are, respectively. In order to find the optimal dimension reduction method and number of dimensions, we perform finger character recognition with several cases of dimensions. The method of dimension reduction and the number of dimensions at the time of the best accuracy are adopted as the optimal parameters. Three methods were used for dimension reduction: Latent Semantic Analysis (LSA), Principal Component Analysis (PCA), and Independent Component Analysis (ICA), and the number of dimensions was reduced by one dimension from 42. The experiment was performed using features extracted from 41 static finger character videos of 16 cooperating subjects.

Table 2: Comparison of accuracy in the number of dimensions when the highest recognition accuracy was achieved for each method.

Dimension Reduction Method	Number of Dimensions	Accuracy (max)	Accuracy (min)	Accuracy (average)
-	42	70.73%	43.90%	55.24%
LSA	20	70.73%	43.90%	55.52%
PCA	16	73.17%	48.78%	58.68%
ICA	10	75.61%	43.90%	63.27%

As shown in Table 2, the highest accuracy was achieved when ICA was used as the dimension reduction method and set to 10 dimensions, and this combination is the optimal condition for dimension reduction.

4.4 Experiment-3 (Evaluation of accuracy for our method)

We evaluate the accuracy of our proposed method for finger character recognition. Features of finger characters extracted from 16 subjects were used as validation data. The validation data is normalized as shown in 3.2.2.3 and 4.1, dimension reduction is performed using the dimensionality reduction method and the number of dimensions shown in 3.2.3.2 and 4.3, and the similarity measure shown in 3.3.3 is performed between the knowledge base shown in 3.2. The percentage of correct answers for each character in the finger character is shown in Table 3. The overall accuracy is 63.27% with the highest average of accuracy when set to 10 dimensions using ICA as shown in Table 2.

Table 3: Result of the number of correct answers for each character recognition.

A	93.8%	I	75.0%	U	62.5%	E	62.5%	O	31.3%
KA	0.0%	KI	93.8%	KU	100.0%	KE	75.0%	KO	18.8%
SA	87.5%	SHI	81.3%	SU	0.0%	SE	25.0%	SO	6.3%
TA	81.3%	CHI	50.0%	TSU	43.8%	TE	81.3%	TO	100.0%
NA	93.8%	NI	68.8%	NU	68.8%	NE	81.3%		
HA	50.0%	HI	56.3%	HU	12.5%	HE	37.5%	HO	87.5%
MA	100.0%	MI	87.5%	MU	100.0%	ME	100.0%		
YA	75.0%			YU	81.3%			YO	100.0%
RA	6.3%			RU	31.3%	RE	93.8%	RO	31.3%
WA	87.5%								

4.5 Discussion

In this section, we describe the results of the evaluation experiment of static finger character recognition conducted in section 4.4. As shown in Table 3, there is a significant difference in the recognition accuracy of each finger character. As shown in Figure 9, among finger characters with significantly low recognition accuracy, for instance, “KA,” “SU,” “SO,” “HU,” "HE," and "RO," there are variations in wrist angles and finger angles and openings among different people even though the hand shape is the same. Other cases were that the skeletal coordinates were not detected correctly. This ease of individual differences in particular characters and the existence of finger characters for which skeletal coordinates are difficult to detect are thought to have led to incorrect recognition.

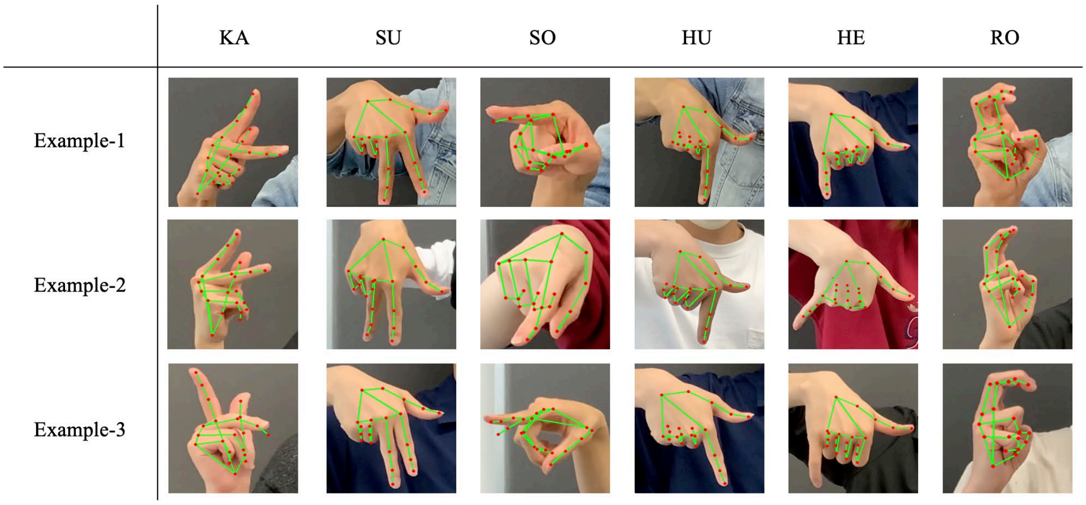


Figure 9: Examples of characters with low recognition accuracy and variations of the same character.

5 Conclusion

In this paper, we present a novel finger character recognition method in sign language using dimension reduction finger character feature knowledge base for similarity measure. By performing similarity measures using a knowledge base, it is possible to perform fingerprint recognition with very few data sets. In addition, by character switching detection and dimension reduction, the computational cost was reduced, and recognition accuracy was improved. We developed a prototype system for finger character recognition by utilizing our proposed method. In addition, we conducted experiments and evaluations. As a result, we found that the optimal method for dimension reduction of finger character features is Independent Component Analysis (ICA), the optimal number of dimensions is 10, and the overall recognition accuracy is about 63% at that time.

In the future, we plan to develop an application for translating finger characters on mobile devices using our proposed method. We also intend to work on utilizing the method for finger character recognition and sign language recognition in countries other than Japan, as well as other applications such as facilitating communication by recognizing hand gestures in Virtual Reality space.

References

[1] G. Fang, W. Gao, and D. Zhao, "Large vocabulary sign language recognition based on fuzzy decision trees," IEEE Transactions on Systems, Man, and Cybernetics - Part A: Systems and Humans, vol. 34, no. 3, 2004, pp. 305-314.

[2] W.W.Kong and S.Ranganath, "Signing Exact English (SEE): Modeling and recognition," Pattern Recognition, vol. 41, no. 5, 2008, pp. 1638-1652.

[3] K. Cem et al., "Hand pose estimation and hand shape classification using multi–layered randomized decision forests," 2012 Proceedings of the 12th European conference on Computer Vision – Volume Part VI, pp. 852-863.

[4] L. Gu, X. Yuan and T. Ikenaga, "Hand gesture interface based on improved adaptive hand area detection and contour signature," 2012 International Symposium on Intelligent Signal Processing and Communications Systems, Taipei, 2012, pp. 463-468.

[5] K. M. Lim, A.W.C. Tan and S. C. Tan, "Block-based histogram of optical flow for isolated sign language recognition," Journal of visual communication and image repre-sentation, vol. 40, part B, 2016, pp. 538- 545.

[6] D. Konstantinidis, K. Dimitropoulos and P. Daras, "A deep learning approach for analyzing video and skeletal features in sign language recognition," 2018 IEEE International Conference on Imaging Systems and Techniques (IST), Krakow, 2018, pp1-6.

[7] T. Simon et al., "Hand keypoint detection in single images using multiview bootstrapping," 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Honolulu, HI, 2017, pp. 4645-4653.

[8] O. Koller, S. Zargaran and H. Ney, "Re-sign: Re-aligned end-to-end sequence modelling with deep recurrent CNN-HMMs," 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Honolulu, HI, 2017, pp. 3416-3424.

[9] T. Liu, W. Zhou and H. Li, "Sign language recognition with long short-term memory," 2016 IEEE International Conference on Image Processing (ICIP), Phoenix, AZ, 2016, pp. 2871–2875.

[10] D. Guo et al., "Sign language recognition based on adaptive hmms with data augmentation," 2016 IEEE International Conference on Image Processing (ICIP), Phoenix, AZ, 2016, pp. 2876–2880.

[11] J. Zhang, W. Zhou and H. Li, "A threshold-based hmm-dtw approach for continuous sign language recognition," 2014 Proceedings of International Conference on Internet Multimedia Computing and Service, pp. 237-240.

[12] CR. De. Souza and E. Pizzolato, "Sign language recognition with support vector machines and hidden conditional random fields going from fingerspelling to natural articulated words," 2013 Proceedings of the 9th international conference on Machine Learning and Data Mining in Pattern Recognition, pp. 84-98.

[13] Q. Facundo et al., “A study of convolutional architectures for handshape recognition applied to sign language”, 2017 XXIII Congreso Argentino de Ciencias de la Computación, La Plata, 2017, pp.13-22.

[14] A. Saxena, D. K. Jain and A. Singhal, “Sign language recognition using principal component analysis,” 2014 Fourth International Conference on Communication Systems and Network Technologies, Bhopal, 2014, pp. 810-813.

[15] Y. Kiyoki, T. Kitagawa and H. Takanari, “A metadatabase system for semantic image search by a mathematical model of meaning,” ACM Sigmod Record, vol.23, no. 4, 1994, pp. 34-41.

[16] T. Kitagawa and Y. Kiyoki, “A mathematical model of meaning and its application to multidatabase systems,” Proceedings RIDE-IMS `93: Third International Workshop on Research Issues on Data Engineering: Interoperability in Multidatabase Systems,Vienna, Austria, 1993, pp.130-135.

[17] F.Zhang et al., “MediaPipe Hands: On-device Real-time Hand Tracking,” CVPR Workshop on Computer Vision for Augmented and Virtual Reality, Seattle, WA, USA, 2020.

[18] S. Hagimoto et al., “Knowledge Base Creation by Reliability of Coordinates Detected from Videos for Finger Character Recognition,” Proc. 19th IADIS International Conference e-Society 2021, FSP 5.1-F144, 2021, pp.169-176.

[19] P. Comon, “Independent Component Analysis, a new concept?,” Signal Processing, Elsevier, vol 36, 1994, pp.287-314.