Sensor Glove Approach for Continuous Recognition of Japanese Fingerspelling in Daily Life

Yuhki Shiraishi* Akihisa Shitara†, Fumio Yoneyama* and Nobuko Kato*
*Faculty of Industrial Technology, Tsukuba University of Technology, Japan
Email: {yuhkis, yonefumi, nobuko}@a.tsukuba-tech.ac.jp
†Graduate School of Library, Information, and Media Studies, University of Tsukuba, Japan
Email: theta-akihisa@digitalnature.slis.tsukuba.ac.jp

Abstract—**To achieve smooth communication between the deaf and hard of hearing and hearing people, we developed a Japanese fingerspelling (JF) recognition system based on sensor gloves. A light and inexpensive sensor glove was adapted for the daily use of the system. We conducted evaluation experiments using a convolutional neural network (CNN) to recognize 76 characters in JF. The target JF alphabet included 35 characters for dynamic fingerspelling, and required both finger and wrist movement. The experimental results show that the average recognition rate of the developed system was approximately 70.0%. Additionally, we conducted a continuous fingerspelling recognition experiment using CNNs and long short-term memory (LSTMs) networks, aiming to recognize consecutive fingerspelling. We proposed a dataset to exploit the characteristics of JF and selected 64 words according to the finger flexion, direction, and movement differences among various signers. Using the collected data, we then conducted evaluation experiments with seven types of neural networks. The overlapping characteristics present in JF were exploited because finger flexion, finger extension, hand direction, and hand movements vary significantly among people currently learning sign language, people corresponding in Japanese sign language (JSL), and people using JSL in their daily lives. Consequently, the average recognition rate (micro F-measure) of 76 JF characters was approximately 92.1%. Based on the results of single fingerspelling and continuous fingerspelling recognition experiments, we discussed the issues concerning the recognition of JF characters and development of sign language recognition systems.**

Keywords–Sign language; Japanese fingerspelling; Sensor glove; Recognition; Convolutional neural network; Long short-term memory.

I. Introduction

In this study, we developed a Japanese fingerspelling (JF) recognition system incorporating a sensor glove and deep learning to achieve smooth communication between the deaf and hard of hearing (DHH) and hearing people, and investigated the recognition rate of the JF alphabet.

This study extends our initial study [1] on a sensor-glove-based JF recognition system for using deep learning to realize smooth communication between DHH and hearing people.

In recent years, interest in speech recognition and information technology devices with voice input functions has increased. Various applications, such as UDtalk [2], KoeTra [3], and cloud-speech-to-text services [4] have been released to provide information accessibility to the DHH based on speech recognition. Consequently, the DHH can read text corresponding to the speech of hearing people.

As a primary communication method, sign language is used in everyday conversations among the DHH. However, hearing people find reading sign language difficult; this results in a communication gap between DHH and hearing people.

Sign language has different characteristics from spoken language. It is expressed itself through finger extensions and flexion, hand directions, hand movements, and facial expressions. Hence, learning and reading sign language is difficult. Therefore, a system for converting sign language into voice information or text information (i.e., a sign language recognition system) is necessary (see Figure 1).

Research has been conducted on information accessibility systems for sign language recognition [5]–[12]. However, compared with information accessibility systems based on speech recognition that are fast reaching maturity, the development of a practical sign language recognition system remains in progress.

In this context, even in a specific country, for example, Japan, differences exist in sign language expressions in the daily lives of people new to sign language, those using signed exact Japanese (SEJ), and those using Japanese Sign Language (JSL). A person learning sign language for the first time learns sign language using a dictionary and other teaching materials. Sign language dictionaries contain many standardized finger expressions for people to imitate and practice, and for slowly and carefully expressing themselves. People using SEJ express themselves one word at a time, as in Japanese, and not using facial expressions. Conversely, people using JSL express themselves using their fingers, hand directions, hand movements, facial expressions, etc. People with relatively little experience in sign language tend to express themselves slowly, whereas people with more experience tend to express themselves quickly.

In addition to using sign language, the DHH people use fingerspelling, e.g., to express their names, proper nouns, and words not present in JSL. As mentioned earlier, finger flexion, hand directions, and hand movements can vary depending on if the person is new to sign language, uses SEJ, or uses a JSL. For example, although the finger positions of “ka” and “ga” are identical, the hand movements are different. In this case, an evident difference exists between the hand movements among the three groups, i.e., those new to the sign language, those using an SEJ, and those using a JSL.

In this study, we developed a JF recognition system based on a sensor glove, deep learning, and acquired data on finger flexion, hand directions, and hand movements for JF signing used in daily life.

A sign language recognition system must recognize hand positions, directions, shapes, and motions. Methods for recognizing sign language can be classified broadly into non-contact

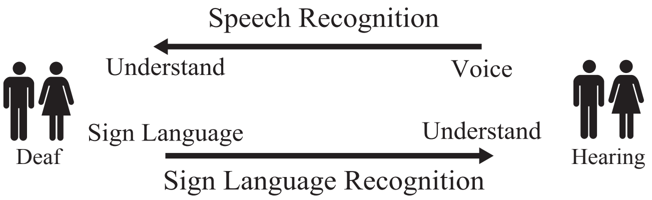


Figure 1. Information accessibility system.

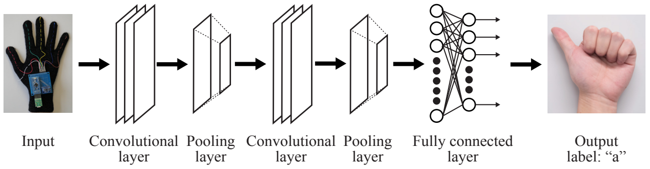


Figure 2. Recognition diagram.

TABLE I. NUMBER OF FINGERSPELLING CHARACTERS IN DIFFERENT COUNTRIES.

Language	*Dynamic*	*Static*	*Sum*
American	2	24	26
French	3	23	26
Japanese	35	41	76

approaches such as recognition using cameras [5] [6] [10], and contact approaches, such as those using sensor gloves [7] [8] [11] [12].

Luzhnica et al. [7] reported a recognition accuracy of 98.5% for sign language using a sensor glove; however, they only considered approximately 30 recognition candidate classes, making this method insufficient for practical use.

In recent years, technologies based on deep learning have attracted significant attention. By increasing the number of hidden layers in a neural network, we can improve the recognition rates of deep learning, which is a type of machine learning. Various techniques for applying deep learning have been reported for improving the gesture recognition accuracy based on image recognition [5].

A camera is a non-contact-type sensor, but is difficult to use for sign language recognition in daily life because it is easily affected by environmental factors, complicating its use in different environments. In addition, when standing in front of people and a camera at a lecture, a speaker tends to speak to the camera without looking at the people as necessary, thereby constraining the speaker from making a connection with the audience. In contrast, hand shape recognition using contact sensors such as sensor gloves is easier, because the sensors are attached directly to the hands.

We were motivated by the goal of improving recognition accuracy using conductive fiber weaving technology [13], as this technology can reduce the weight and cost of sensor gloves and simplify hand movements used in daily life for easy recognition by deep learning (see Figure 2).

In our experiments, we evaluated our developed system by classifying 76 characters of the JF alphabet, including dynamic (non-static) fingerspelling characters; those are a unique feature of JF compared to other fingerspelling systems, as shown in Table I.

The evaluation experiments for a single JF were conducted using a convolutional neural network (CNN) as a learning model (this type of model performed the best in previous studies) to reduce the data reduction by calculating the moving averages of the data acquired from gyro sensors. In these experiments, all 76 JF characters of JF were included as recognition targets, as were dullness, semi-voiced sounds, diphthongs, and long vowels. These experiments were conducted using all the collected data under various experimental conditions.

In the continuous JF recognition evaluation experiment, we utilized the neural network constructed for the single JF recognition task as a learning model, introduced a long short-term memory (LSTM), and built seven types of neural networks. As in the evaluation experiment for single-finger character recognition, all 76 characters of JF are used for recognition. Furthermore, evaluation experiments were conducted for consecutive finger characters with two or more characters. In this experiment, we propose a dataset that exploits the characteristics of JF, and selected 64 words owing to the differences in finger flexion, directions, and movements differences among people new to sign language, people using SEJ, and people using JSL. We then conducted evaluation experiments with the seven types of neural networks using the collected data.

This study provides the following contributions:

- the development and evaluation of a fingerspelling recognition system using an inexpensive and lightweight sensor glove;
- the development and evaluation of a continuous JF recognition system using CNNs and LSTMs; and
- the proposal and evaluation of a dataset for fingerspelling recognition in the daily lives of various signers.

In Section II, we introduce the related research results. In Section III, we describe the single fingerspelling recognition experiments. In Section IV, we describe the continuous fingerspelling recognition experiments. In Section V, we provide our conclusions.

II. RELATED WORK

Previous research on fingerspelling recognition has proposed two types of sensors for recognizing a series of operations in fingerspelling: contact-type sensor gloves and non-contact-type cameras.

A. Image recognition

Several methods have been proposed for recognizing hand shapes based on processing images of fingerspelling as captured by cameras. Mukai et al. [9] reported that a fingerspelling recognition method targeting 41 immobile characters in JSL resulted in an average recognition accuracy of 86%. They used a classification tree and machine learning based on a support vector machine to classify individual images. Hosoe et al. [10] used deep learning for recognition and achieved a recognition rate of 93%, but only for static fingerspelling. Jalal et al. [6] reported a recognition rate of 99% for American sign language (ASL) images based on a deep learning algorithm for static fingerspelling (i.e., excluding “J” and “Z”). However, the recognition accuracy could not be considered as sufficient for practical recognition in JF. Additionally, relatively few recognition results have been reported for dynamic fingerspelling (i.e., fingers moving when expressing a character). In a study of dynamic fingerspelling in JSL [14], the identification of hand shapes was performed using a kernel orthogonal mutual subspace method from images of hand regions obtained from distance images, and the classification of movements was performed using decision trees based on center-of-gravity coordinates. These results yielded a 93.8% identification rate. However, the recognition accuracy was insufficient for the practical recognition required for JF.

B. *Sensor glove recognition*

Several methods have been proposed for recognizing hand shapes based on measurement data acquired by contact-type sensor gloves. These methods can measure finger flexion, hand positions, and directional data. The measurement data are then sent to a personal computer, and a classification algorithm is used to recognize hand shapes. Cabrera et al. [11] paired the Data Glove 5 Ultra [15] sensor glove with an acceleration sensor to acquire information regarding the degree of flexion of each finger and wrist direction. They conducted test classification using 24 static fingerspelling characters in ASL, excluding “J” and “Z.” Their neural network was trained using 5 300 patterns and achieved a recognition rate of 94.07% for 1 200 test patterns. Mummadi et al. [12] prototyped a sensor glove with multiple embedded inertial sensors. They collected French sign language fingerspelling data from 57 people and achieved an average recognition rate of 92% with an F1-score of 91%. Kakoty et al. [16] reported on a dataset of one-handed Indian sign language alphabets (C, I, J, L, O, U, Y, W), ASL alphabets (A to Z), and signed numbers (0 to 9), using a radial basis function with 10-fold cross-validation Using a kernel-supported vector machine, they achieved an average recognition rate of 96.7% and reported that the data were converted to speech. Chong et al. [17] placed six inertial measurement units (IMUs) on the back of the palm and on each fingertip to capture their motion and orientations. Ultimately, 28 proposed word-based sentences in ASL were collected, and 156 features were extracted from the collected data for classification. Using the long short-term memory (LSTM) algorithm, the system achieved an accuracy of up to 99.89%. Notably, 12 people cooperated with us in the data collection experiment, but whether they were deaf or hearing people was unclear. Yu et al. [18] reported on the architecture of a data glove system comprising a stnm32MCU, flex4.5 bending sensor, mpu6050 six axis sensor, Bluetooth transmission module, and cellphone voice application. The system was developed and connected to a Java-based processing software. They reported that their system recognized sign language movements and could output the words to be said using the intelligent voice system. However, the glove does not feature global movement and rotation tracking. Glauser et al. [19] demonstrated a glove’s performance in a series of ablation experiments while exploring various models and calibration methods. However, the glove does not come with a global translation and rotation tracking. Realizing a sign language recognition system requires hand orientations and motions. Among the various methods for performing JF recognition, the conductive fiber braid method [13] uses gloves woven with conductive fibers instead of flexion sensors. These gloves can recognize hand shapes and movements as they are directional gyro sensors incorporated into them. However, the recognition rate for JF (“a,” “i,” “u,” “e,” “o”) based on Euclidean distance has been reported as only 60%.

C. *Data collection*

Regarding image recognition, several large-scale continuous sign language recognition (CSLR) benchmarks have been published [20]. For example, we introduced three large-scale CSLR benchmarks: PHOENIX-2014, Chinese sing language (CSL), and PHOENIX-2014-T. PHOENIX-2014 is a publicly available German Sign Language dataset and the most famous CSLR benchmark. This corpus is taken from broadcast news regarding the weather. The CSL dataset consists of 100 sign language sentences and 178 words related to everyday life. Fifty signers performed each sentence, resulting in 5,000 videos in total. A matched isolated CSL database containing 500 words is also provided for pre-learning. Each word was performed 10 times by 50 signers. PHOENIX-2014-T annotates the new videos with two annotations: the sign language terms for the CSLR task, and the German translation for the a sign language translation (SLT) task. The vocabulary consists of 1,115 terms for sign language and 3,000 for German. This dataset is available in [21]. However, the data of these three large-scale CSLR benchmarks are insufficient to realize a highly accurate sign language recognition system using deep learning. Further research is being conducted to increase the amount of available data.

Extensive data for image recognition can be obtained from online sources. For example, the Shi et al. [22] dataset contains clips of fingerspelling sequences cut from sign language “in the wild” videos obtained from online sources such as YouTube and dafvideo.tv [23]. The datasets contain 5,455 training sequences from 87 signers of “ChicagoFSWild,” 981 development (validation) sequences from 37 signers, and 868 test sequences from 36 signers, without overlapping signers among the three sets. Another dataset, “ChicagoFSWild+,” contains 50,402 training sequences from 216 signers, 3115 development sequences from 22 signers, and 1,715 test sequences from 22 signers. Compared to ChicagoFSwild, the crowdsourcing setup of ChicagoFSWild+ enables the collection of considerably more training data while significantly reducing the efforts of experts and researchers.

Danielle et al. [24] expressed privacy concerns regarding contributing to a filtered sign language corpus, using very expressive avatars and blurred faces, which may affect the willingness to participate. Training on filtered data may improve the recognition accuracy. In the case of camera recognition, the look of the face is also captured; thus, privacy must also be considered. In contrast, sensor glove recognition does not require pictures of the face; thus privacy concerns are reduced and the data can be more simply collected.

III. SINGLE FINGERSPELLING RECOGNITION EXPERIMENT

To achieve smooth communication in real-world environments, we designed a system for communicating information using lightweight and comfortable sensor gloves for recognizing fingerspelling with high accuracy in real time. The

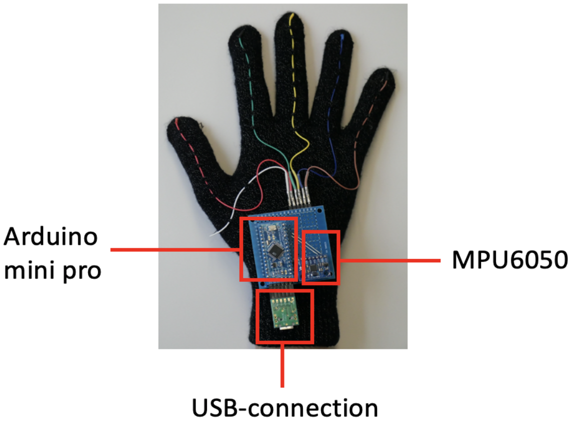


Figure 3. Prototype of sensor glove.

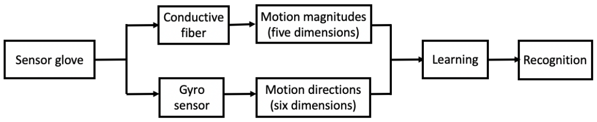

Sensor glove
Conductive fiber
Motion magnitudes (five dimensions)
Gyro sensor
Motion directions (six dimensions)
Learning
Recognition

Figure 4. Software structure.

developed system consists of a sensor value measurement unit and recognition unit. Figure 3 shows the JF recognition system developed in this study. Figure 4 shows the corresponding software architecture.

A. *Sensor glove*

To efficiently recognize fingerspelling efficiently based on hand, finger, and wrist data, detecting motion magnitudes and directions using the sensor glove is necessary. In this study, we adopted a hand shape recognition technique using conductive fiber sensor gloves, which are more comfortable, less expensive, and lighter than traditional sensor gloves. Motion directions are detected using a gyro sensor, whereas motion magnitudes are detected based on resistance changes in the conductive fibers of the gloves. The motion detection board is an Arduino board and the measurement values from the sensor glove are transferred from the detection board to a PC, where they are saved in comma-separated-value format. The machine learning and motion recognition are performed using Python implementations on a PC. The sensor readings for JF motion from the data gloves have different scales depending on the wearer. Therefore, the data are subjected to linear normalization in consideration of the differences in movement. Additionally, because the activation and likelihood functions of the proposed system are based on probabilities, as a prepossessing for the network inputs, we perform scale conversion to a range of zero to one.

The motion magnitudes are detected based on the resistance changes in the conductive fibers during flexion and extension of the fingers. We use partial pressure values to calculate the input voltages based on (1).

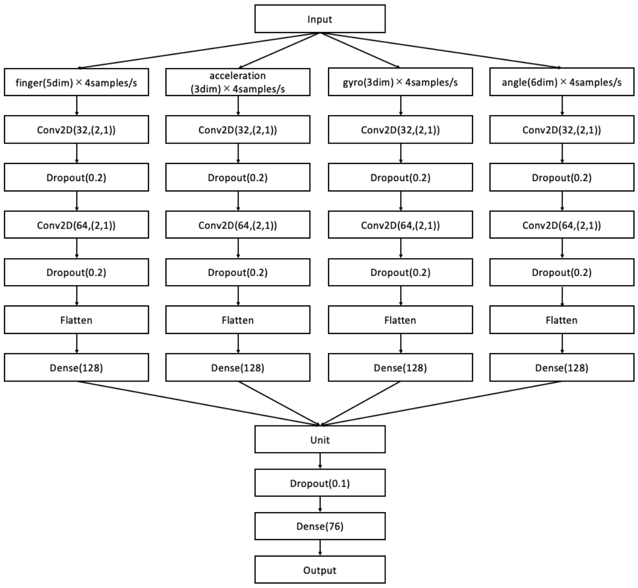


Figure 5. Architecture of the convolutional neural network.

$$V_{in} = \frac{R_1}{R_1 + R_2} * V_{out} \tag{1}$$

In this equation, V_{in} is the estimated motion magnitude, V_{out} is the reference voltage, R_1 is the variable resistance of the conductive fibers, and R_2 is a fixed resistance. When a finger is stretched, the resistance value of the conductive fiber increases. When a finger is bowed, the resistance value of the fiber decreases.

B. *Recognition algorithm*

In this study, we adopted a CNN. This type of network has achieved high recognition rates in previous studies. The CNN and k-fold cross-validation are implemented using the open-source libraries, TensorFlow [25] and scikit-learn [26]. We also adopted the RMSprop training algorithm [27]. The activation function is a rectified linear unit, as shown in (2). The error function is the cross-entropy function shown in (3), where t_k is the correct label (one-hot expression) and y_k expresses the network output.

$$f(u) = max(u, 0) \tag{2}$$

$$E = -\sum_k t_k \log y_k \tag{3}$$

The main features of CNNs are the convolutional and pooling layers. These layers are updated as their feature values are extracted during the training process. We transform the measurement data acquired by the sensor glove into two dimensions based on training and evaluation trials. The motion magnitudes, accelerations, and gyro readings are branched at the time of input. Through the CNN (typical layer size of 32 to 64 nodes), these data are coupled using “Flatten” and “Dense” operations (128 nodes). Finally, the outputs are generated using

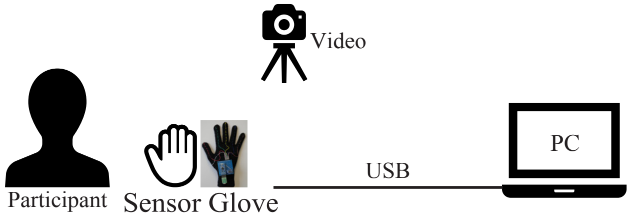


Figure 6. Data acquisition experiment.

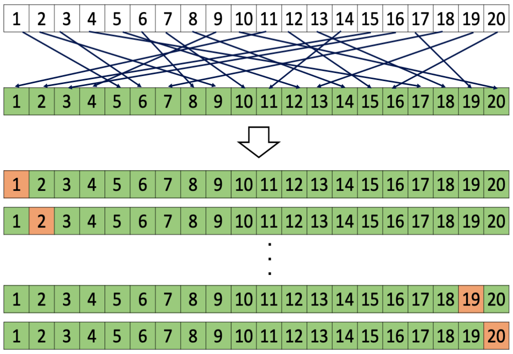


Figure 7. Twenty-fold cross-validation by shuffling data.

TABLE II. Twenty-fold cross-validation results.

k	Learning data (%)	Validation data (%)
1	93.6	65.0
2	94.1	75.5
3	94.8	68.7
4	93.1	69.7
5	94.2	66.3
6	93.9	73.2
7	92.9	67.9
8	93.5	71.1
9	93.0	67.4
10	94.6	70.5
11	93.4	71.6
12	93.0	66.1
13	94.6	68.9
14	94.3	70.3
15	93.0	69.7
16	93.4	68.4
17	92.9	71.3
18	93.1	71.1
19	94.5	74.2
20	94.5	72.4
Average	93.7	70.0

TABLE III. Misrecognition patterns.

Teacher	a	sa	ku	yo	ke	te	ki	chi	chi	tsu	ni	ha	ne	ma	hi	re	wo	xya	gi	di	ge	de	di	du	zo	bu	
Prediction	sa	a	yo	ku	te	ke	chi	ki	tsu	chi	ha	ni	ma	ne	re	hi	xya	wo	di	gi	de	ge	du	di	bu	zo	
Rate (%)	21.0	19.0	14.0	20.0	12.0	28.0	12.0	12.0	34.0	32.0	20.0	22.0	13.0	11.0	19.0	23.0	11.0	13.0	12.0	13.0	29.0	20.0	39.0	35.0	14.0	15.0	

an additional Dense operation (76 nodes) corresponding to the number of JF characters, outputs are generated. Figure 5 presents a system overview of the CNN. In the CNN, inputs are initially separated based on the physical meanings of each signal. The separated signals are eventually combined to recognize JF characters.

C. Data collection

To target the 76 JF characters, we recruited 20 participants (from 20 to 27 years old). In our experiments, each participant wore a sensor glove and performed the motions of the finger-spelling characters in sequence for 1 s at a time according to directions provided by a moderator. As shown in Figure 6, video was also recorded to capture the motions of the wrists and fingers of the participants. For each 1 s motion and at a rate of 200 samples per second (sps), the sensor gloves captured five dimensions of motion magnitude data, three dimensions of acceleration data, and three dimensions of gyro data, to obtain data for 11 dimensions. Data labeling was conducted manually and simultaneously with the data collection. This series of motions was repeated five times. Therefore, with five repetitions per participant, 76 JF characters, and 200 sps for 1 s, a total of 76,000 motion measurement data were collected for each participant. We were able to collect a total of 1,520,000 data samples from all 20 participants. These experiments were conducted with approval from the Tsukuba University of Technology Research Ethics Committee (approval number: H30-17).

First, we performed extensive data cleaning and feature selection operations. T o prevent gyro drift, we used Madgwick filters [28] to calculate angles from the values of the acceleration and gyro sensors in real time. This enabled calculations of three angle dimensions from the acceleration and gyro data. To clarify the hand directions, the angles were converted into sine and cosine data. The resulting six dimensions were combined with the aforementioned motion magnitudes (five dimensions) and motion directions (six dimensions) mentioned above to generate a total of 17 dimensions. Next, we conducted a review of the sampling frequency. Although 200 sps could be acquired without leakage, noise and training times were included in these samples. Therefore, the number of data was reduced by calculating a moving average to achieve a final value of 4 sps.

D. Evaluation experiments

The collected data were evaluated using the CNN (Figure 5) and k-fold cross-validation (k = 20). In our evaluation experiments, data shuffling was performed using Google Colaboratory [29]. The number of folds for the k-fold cross-validation was set to 20 according to the number of participants. Additionally, confusion matrices and accuracy rates were generated using 20-fold cross-validation for all data shuffling evaluations (see Figure 7).

E. Results and discussion

The experimental results from the 20-fold cross-validation are listed in Table II. This table reveals an average recognition rate of approximately 70.0%.

As shown in Figure 8 and Table III, various misrecognition patterns occurred. We believe these patterns occurred because

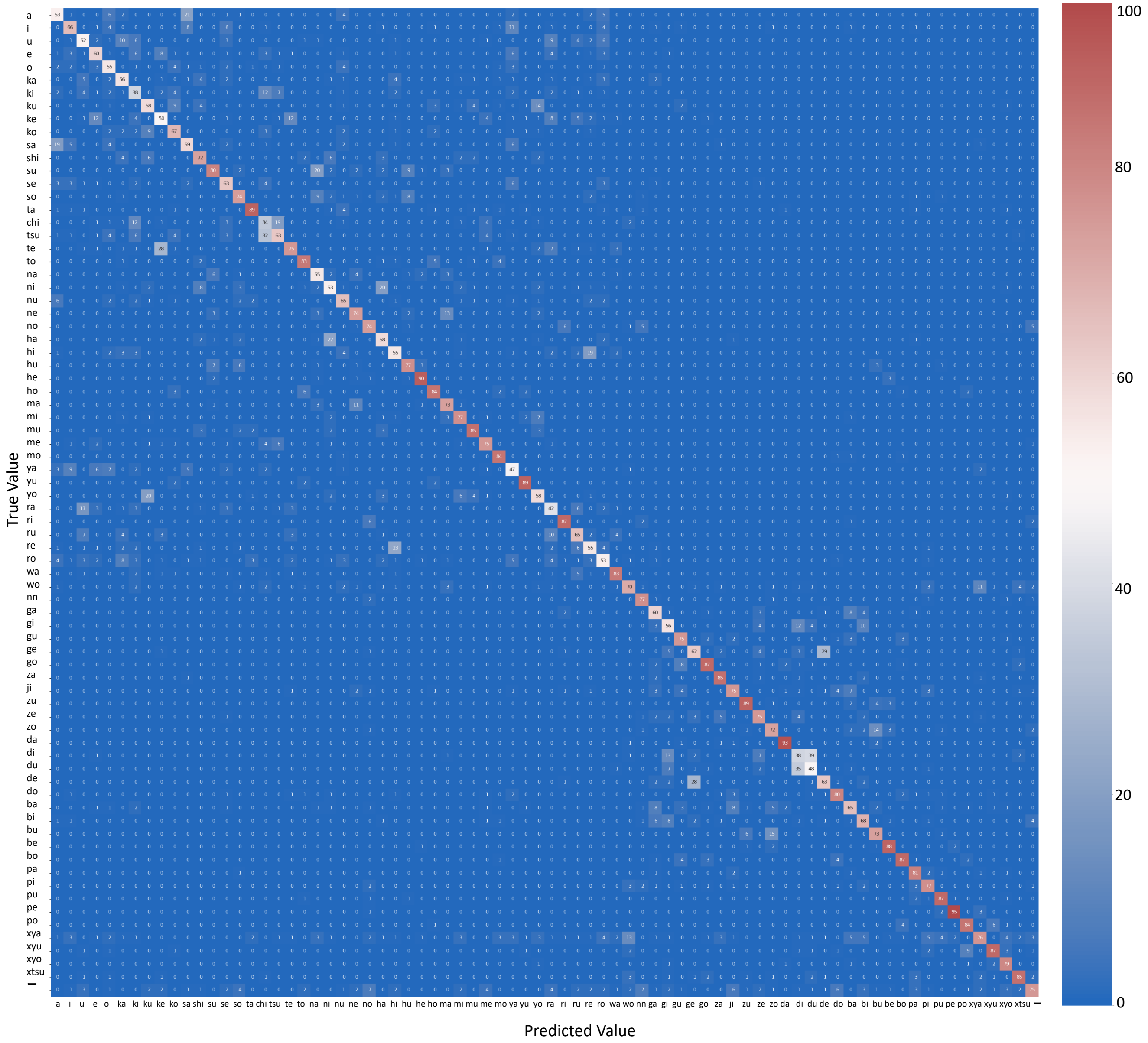


Figure 8. Confusion matrix.

the conductive fibers are firmly attached to the sensor gloves. We confirmed that the hand directions for “ha” and “ni”, which are JF characters, varied among participants. Additionally, “ne” and “ma” appear to be confused based on both hand bending and finger bending.

Figure 9 presents the sample input data leading to misrecognition for the JF characters “te” and “ke”. By analyzing the data, it was confirmed that the close contact between the fingers caused these errors. Notably, the thumb sometimes contacted the forefinger. Additionally, depending on the participant, the hand could be widely opened or the fingers could be in close contact.

Figure 10 presents examples of acquiring data from two participants using the sensor glove for dynamic fingerspelling. This figure clearly highlights the individual differences in fingerspelling between the participants, particularly in the strength of the finger bending (including noisy signals), timing of hand movements, and shapes of the fingers. Therefore, it is necessary to improve the recognition algorithms and data glove devices (e.g., to detect hand movement periods and construct more robust glove devices).

Based on the aforementioned results, we determined that

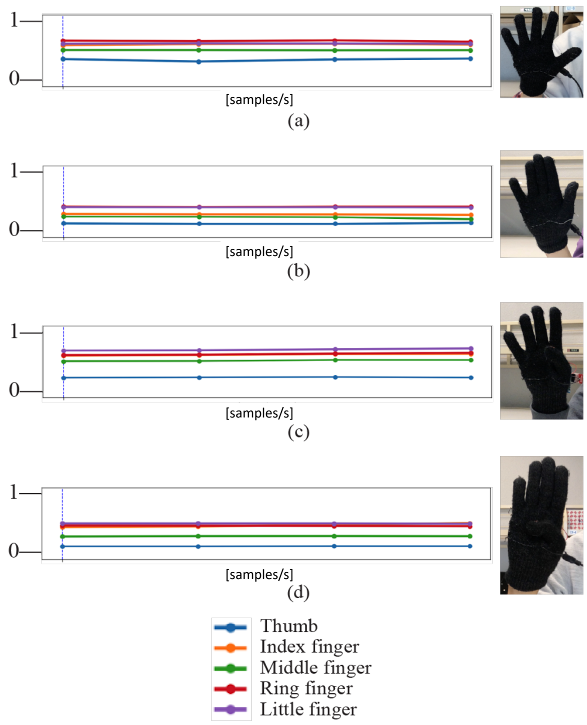


Figure 9. Example input data (only five dimensions): (a) predict “te” as “te” correctly, (b) predict “te” as “ke” incorrectly, (c) predict “ke” as “te” incorrectly, (d) predict “ke” as “ke” correctly.

the recognition errors largely occurred based on variance in the flexion and direction of the fingers. We also confirmed that finger expressions varied based on individual differences, which could be attributed to different home and social environments (making recognition more difficult).

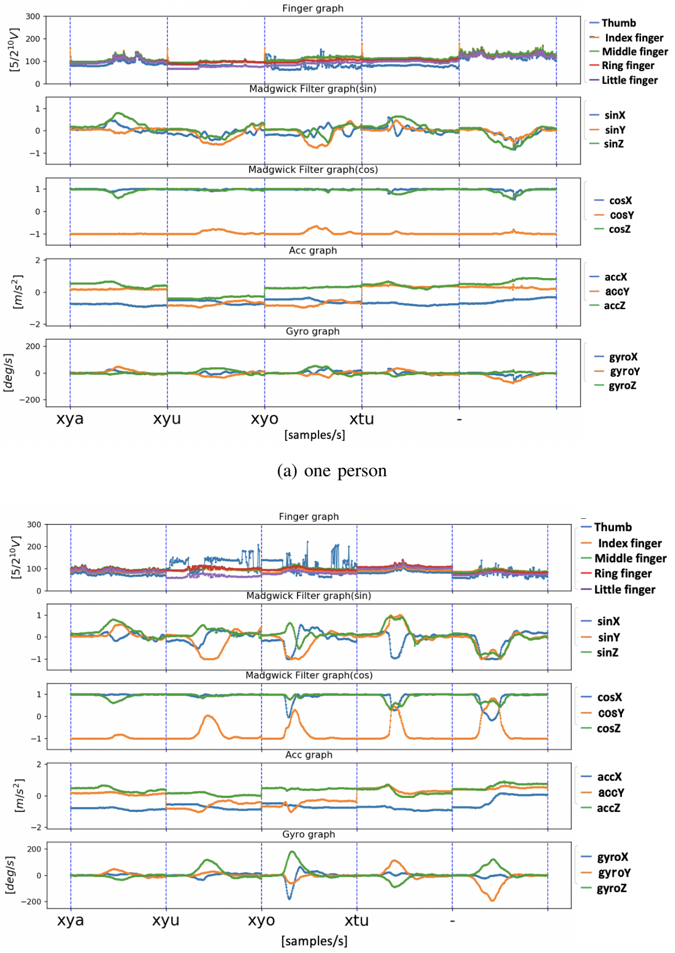
However, JF is widely used for displaying proper names and technical terms. Therefore, the recognition of JF is essential for construction a JSL recognition system.

IV. CONTINUOUS FINGERSPELLING RECOGNITION EXPERIMENT

This section describes the selection of words for data collection and for construction of a new neural network for continuous fingerspelling recognition experiments based on the system constructed in Section III. First, we describe the word selection.

A. Word selection

In a previous study [30], we proposed a method for recognizing fingerspelling words using linguistic information based on a word dictionary. We separated the recognition of actions from the recognition of hand shapes: thus, fingerspelling could be recognized even despite action recognition errors. In this experiment, we proposed 18 patterns because the number of JF patterns is more significant than those of other countries, particularly in dynamic fingerspelling, as described in Section I. Furthermore, finger and hand movements vary from person to person. Each pattern is illustrated and explained. In a previous study [30], the number of words selected was 64; thus, we selected words corresponding to that number and suitable for the 18 proposed patterns (see Figure 11). The errors were characterized as follows:



(a) one person

(b) another person

Figure 10. Example of acquiring data.

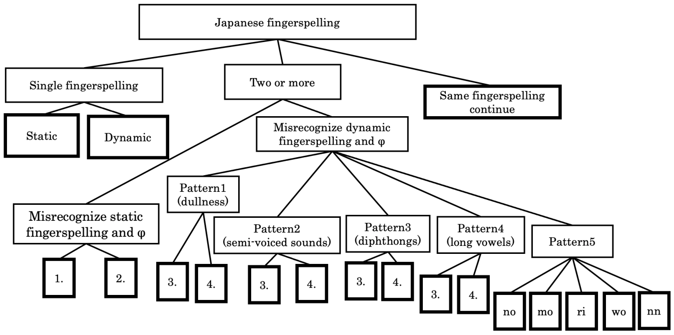


Figure 11. 64 word patterns.

1 denotes the misrecognition of static fingerspelling as the transition movements between fingerspellings;

2 denotes the misrecognition of transition movements as

static fingerspelling;

3 denotes the misrecognition of dynamic (non-static) fingerspelling as transition movements;

4 denotes the misrecognition of transition movements as dynamic fingerspelling. The tasks and groups are explained in more detail below.

Task 1: Single fingerspelling

1-1 Static fingerspelling

This comprises fingerspelling other than 1-2 dynamic fingerspelling. It is characterized by absence of hand movements.

1-2 Dynamic (non-static) fingerspelling

There are four types of dynamic fingerspellings: dullness, semi-voiced sounds, diphthongs, and long vowels. Dynamic fingerspelling is characterized by hand movements.

Task 2: Two or more fingerspellings

2-1 Misrecognizing static fingerspelling and transition movements.

2-1-1 Misrecognizing transition movements as static fingerspelling

For example, “[ta]” may be misrecognized as “[ta][ta].” The user may misrecognize “[ta]” two or more times in succession.

2-1-2 Misrecognizing static fingerspelling as transition movements

For example, “[ta][ta]” may be misrecognized as “[ta].”

2-2 Misrecognizing dynamic fingerspelling and transition movements

2-2-1 Pattern1 dullness

2-2-1-1 Misrecognizing transition movements as dynamic fingerspelling

For example, “[ta][da]” may be misrecognized as “[da].”

2-2-1-2 Misrecognizing dynamic fingerspelling as transition movements

For example, “[da]” is misrecognized as “[ta][da].”

2-2-2 Pattern2 semi-voiced sounds

2-2-2-1 Misrecognizing transition movements as dynamic fingerspelling

For example, “[pa][pa]” may be misrecognized as “[pa].”

2-2-2-2 Misrecognizing dynamic fingerspelling as transition movements

For example, “[pa]” may be misrecognized as “[pa][pa].”

2-2-3 Pattern3 diphthongs

2-2-3-1 Misrecognizing transition movements as dynamic fingerspelling

For example, “[tsu][tsu]” may be misrecognized as “[tsu][xtsu][tsu].”

2-2-3-2 Misrecognizing dynamic fingerspelling as transition movements

For example,“[tsu][xtsu][tsu]” may be misrecognized as “[tsu][tsu].”

2-2-4 Pattern4 -(long vowels)

2-2-4-1 Misrecognizing transition movements as dynamic fingerspelling

For example, “[hi][-(long vowel)]” may be misrecognized as “[- (long vowel)].”

2-2-4-2 Misrecognizing dynamic fingerspelling as transition movements

For example, “[- (long vowel)]” may be misrecognized as “hi- (long vowel).”

2-2-5 Pattern5

2-2-5-1 Misrecognizing “[no]” as transition movements

For example, “[no]” may be misrecognized as “[hi][no].”

2-2-5-2 Misrecognizing “[mo]” as transition movements

For example, “[mo]” may be misrecognized as “[to][mo].”

2-2-5-3 Misrecognizing “[ri]” as transition movements

For example, “[ri]” may be misrecognized as “[u][ri].”

2-2-5-4 Misrecognizing “[wo]” as transition movements

For example, “[wo]” may be misrecognized as “[o][wo].”

2-2-5-5 Misrecognizing “[nn]” as transition movements

For example, “[nn]” may be misrecognized as “[hi][nn].”

Task 3: Identical fingerspelling problems

For example, “[ta][ta]” may be misrecognized as “[ta].”

The words are selected using the 18 patterns described in Figure 11. Because finger and hand movements vary from person to person, we organized the words into 26 groups. Table IV shows the fingerspelling of each group.

Group 1: [no][u][ni][xyu][u], [u][ri][xyu][u], [ri][yu][u]

We need to identify “[u][ni]”, but it may be “[u][ri]”. The transitions from “[u]” to “[ni]” and from “[u]” to “[ri]” have similar hand movements, with there being significant difference in the speed of the fingers. In this experiment, we will collect data on three words, “[u][ri][xyu][u]”, “[no][u][ni][xyu][u]”, and “[ri][yu][u]”, and investigate the differences in speed.

Group 2: [su][u][ri][xyo][u], [su][ri][yo][u]

Although the subject of the experiment can correctly express “[su][u][ri]”, the possibility that he will misrecognize it as “[su][ri]” exists. The words “[u]” and “[ri]” are difficult to distinguish and have the same finger positions, and depend on whether hand movements are used or not. Therefore, we can expect that “[u]” may be recognized as transition movements, which would imply “[su][ri].” We will use the two-word data of “[su][ri][yo][u]” and “[su][u][ri][xyo][u]” in the recognition experiment.

Group 3: [ru][-][ru], [ru][ru]

Many people expressing “[ru][-][ru]” do not express long vowels. Therefore, a possibility exists of misrecognizing “[ru][-][ru]” as “[ru][ru]”. In this experiment, we conduct a recognition experiment using two sets of data: “[ru][-][ru]” and “[ru][ru]”. We then investigate the need to distinguish the differences between “[ru][-][ru]” and “[ru][-][ru]” to realize a fingerspelling recognition system.

Group 4: [su][su][gi], [su][zu][ki]

The word “[su][su][gi]” is expressed using a downward extension of the thumb, index finger, and middle finger. In addition to the use of “[su]” twice, we also investigate whether the participants could discriminate between the two words “[su][su][gi]” and “[su][zu][ki]” by adding a murmur. This task enables us to consider the algorithms necessary for obtaining sufficient information from moving fingers.

TABLE IV. 64 WORDS INTO 26 GROUPS

Item	Group	Word	1-1	1-2	2-1-1	2-1-2	2-2-1-1	2-2-1-2	2-2-2-1	2-2-2-2	2-2-3-1	2-2-3-2	2-2-4-1	2-2-4-2	2-2-5-1	2-2-5-2	2-2-5-3	2-2-5-4	2-2-5-5	3
1	1	[no][u][ni][xyu][u]	*			*														
2		[u][ri][xyu][u]	*	*													*			
3		[ri][yu][u]	*	*													*			
4	2	[su][u][ri][xyo][u]	*	*		*														
5		[su][ri][yo][u]	*	*	*															
6	3	[ru][-][ru]	*	*										*						
7		[ru][ru]	*																	*
8	4	[su][su][gi]	*	*																*
9		[su][zu][ki]	*	*				*												
10	5	[hu][re][-][mu]	*	*					*					*						
11		[pu][re][mu]	*	*						*			*							
12	6	[tsu][tsu][mi]	*			*	*													*
13		[tsu][du][mi]	*	*			*	*												
14		[chi][di][mi]	*	*																
15		[tsu][mi]	*		*															
16	7	[po][tsu][ri]	*	*																
17		[po][xtsu][tsu][ri]	*	*								*								
18	8	[me][xtsu][ki]	*	*								*								
19		[me][tsu][ki]	*								*									
20	9	[hi][nn][to]	*	*									*						*	
21		[hi][-][to]	*	*										*						
22		[pi][-][to]	*	*						*										
23		[bi][-][to]	*	*						*										
24		[no][-][to]	*	*										*	*					
25	10	[ro][-][so][nn]	*	*		*														
26		[ro][-][nn]	*	*	*															
27	11	[hi][ku]	*		*															
28		[no][ku]	*	*											*					
29	12	[ka][tsu][wo]	*	*														*		
30		[ka][tsu][o]	*																	
31	13	[a][nn][za][nn]	*	*																
32		[te][nn][ke][nn]	*	*																
33		[de][nn][ge][nn]	*	*																
34	14	[ma][tsu][ya]	*																	
35		[ma][chi][ya]	*																	
36		[ma][xtsu][chi][ya]	*	*																
37	15	[ni][ba][i]	*	*				*												
38		[ni][ha][i]	*				*													
39	16	[pa][pa]		*						*										*
40		[ha][ha]	*						*											*
41	17	[mo][to][mo][to]	*	*												*				
42		[to][mo][do][mo]	*	*												*				
43	18	[ta][ta]	*			*	*													*
44		[ta][da]	*	*			*	*												
45		[da][da]		*				*												*
46	19	[ne][sa][se][ru]	*																	
47		[ne][za][sa][se][ru]	*	*				*												
48	20	[he][ya]	*						*											
49		[pe][ya]	*	*						*										
50	21	[ko][so][gu]	*	*		*														
51		[go][zo][ku]	*	*				*												
52		[go][hu][ku]	*	*				*												
53		[ko][bu][ku]	*	*				*												
54	22	[wa][ro][shi]	*																	
55		[wa][nu][shi]	*																	
56	23	[he][ya][gi]	*	*			*													
57		[be][ki]	*	*	*															
58	24	[ho][e][ki]	*				*													
59		[bo][e][ki]	*	*				*												
60	25	[shi][se][i]	*																	
61		[ji][ra][i]	*	*																
62		[shi][ze][i]	*	*																
63	26	[ka][chi][na][no][ri]	*	*																
64		[ga][i][su][u]	*	*																
		Count	62	46	5	6	7	10	3	5	1	2	2	4	2	2	2	1	1	7

Group 5: [hu][re][-][mu], [pu][re][mu]

This task investigates if the system can distinguish between "[hu][re]" and "[mu]." We want to identify "[hu][re][-][mu]," but it could be misidentified as "[pu][re][mu]" owing to fast hand movements. Therefore, we collect two types of data, "[hu][re][-][mu]" and "[pu][re][mu]," in this experiment and conducted a recognition experiment. While analyzing the hand movements, we investigate if we can effectively obtain information regarding the hand movements.

Group 6: [tsu][tsu][mi], [tsu][du][mi], [chi][di][mi], [tsu][mi]

We need to identify "[tsu][tsu][mi]", but cases exist in which "[tsu]" is used twice and the finger flexion becomes "[chi]", which is similar. Therefore, we collect data on four words to conduct a recognition experiments: "[tsu][tsu][mi]", "[tsu][du][mi]", "[chi][di][mi]", and "[tsu][mi]", and conducted a recognition experiment. We examine the hand movements for "[tsu][tsu][mi]" and "[chi][di][mi]."

Group 7: [po][tsu][ri], [po][xtsu][tsu][ri]

For group 6, the word "[tsu][xtsu][mi]" is unavailable. Hence, we prepared the words "[po][tsu][ri]" and "[po][xtsu][tsu][ri]." Thus, in Group 6, "[tsu][tsu][mi]" and "[tsu][du][mi]" are prepared to investigate their possibility of it being recognized as "[tsu][tsu][mi]" by expressing "[xtsu][tsu]" twice in succession.

Group 8: [me][xtsu][ki], [me][tsu][ki]

Unlike group 7, group 8 recognizes only one character of each of "[xtsu]" and "[tsu]." We investigate whether the group misrecognizes "[me][xtsu][ki]" as "[me][tsu][ki]." This is where dynamic fingerspelling is misrecognized as transition movements or static fingerspelling. The same is valid for hand movements where "[xtsu]" is a diphthong. We investigate whether the system can distinguish between "[me][xtsu][ki]" and "[me][tsu][ki]" by distinguishing the difference between "[xtsu]" and "[tsu]."

Group 9: [hi][nn][to], [hi][-][to], [pi][-][to], [bi][-][to], [no][-][to]

This task examines what type of information can be obtained to identify "[hi][nn][to]." The four words "[hi][-][to]," "[pi][-][to]," "[bi][-][to]," and "[no][-][to]" are chosen because they have the potential to produce the same hand movements as "[hi][-][to]." For example, in the "[hi][-]" and "[no][-]" parts, the finger flexion is the same, but the hand movements are different. After distinguishing the hand movements, we investigate whether the participants can identify the five words "[hi][nn][to]", "[hi][-][to]", "[pi][-][to]", "[bi][-][to]", and "[no][-][to]." During the expression of "[hi][-][to]", when "[hi]" ends, the fingers either turn to the right or the left. As the direction depends on the person, this is also be investigated.

Group 10: [ro][-][so][nn], [ro][-][nn]

We need to identify "[ro][-][so][nn]," but it is possible to misidentify as "[ro][-][nn]." Some difficulty seems to exist in discriminating between "[-][so][nn]" and "[-][nn]" seems to exist. The three characters "[-]", "[-][so]", and "[-][nn]" all extend the index finger in the same manner, but the hand direction and movement may differ per person. Therefore, we investigate whether the two words "[ro][-][so][nn]" and "[ro][-][nn]" can be discriminated.

Group 11: [hi][ku], [no][ku]

A possibility exists that "[hi][ku]" may be misrecognized as "[no][ku]." In this experiment, we compare the transition from "[hi][ku]" to "[ku]" and from "[no]" to "[ku]" and discover that distinguishing between "[hi][ku]" and "[no]" at the crossing portion is more challenging. We also investigate whether the discrimination between "[hi]" and "[no]" is possible.

Group 12: [ka][tsu][wo], [ka][tsu][o]

The possibility of misrecognizing "[ka][tsu][wo]" as "[ka][tsu][o]" exists. To determine if distinguishing between "[wo]" and "[o]" is possible, we investigate the misrecognition of dynamic fingerspelling as transition movements or static fingerspelling.

Group 13: [a][nn][za][nn], [te][nn][ke][nn], [de][nn][ge][nn]

We investigate "[ke]," "[te]," "[ge]," and "[te]," "[a]," and "[sa]." In this experiment, we also investigate a problem concerning dullness.

Group 14: [ma][tsu][ya], [ma][chi][ya], [ma][xtsu][chi][ya]

We investigate whether the system can identify the three lower case characters "[tsu]," "[chi]," and "[xtsu]." The hand movements of certain people may be difficult to distinguish when they transition from "[ma]" to "[tsu]," "[chi]", and "[xtsu]." In this experiment, we collect the data of the three characters for a recognition experiment and investigate whether they are affected by the person.

Group 15: [ni][ba][i], [ni][ha][i]

"[ni]" and "[ha]" have the same finger flexion but different hand directions. Similarly, "[ni][ba][i]" and "[ni][ha][i]" have the same finger flexion but different hand direction. In this experiment, we collect the data of the two words for the recognition experiment and investigate whether distinction between the direction, dullness, and transition movements is possible.

Group 16: [pa][pa], [ha][ha]

We investigate whether "[pa][pa]" can be misrecognized as "[ha][ha]". This is a task in which dynamic fingerspelling is misrecognized as transition movements. This experiment specifically examines whether dynamic fingerspelling is misrecognized as "[ha][ha]" or "[pa][pa]." The finger movement may cause transition movements, which may lead to the misrecognition of "[pa]" as "[ha][ha]." Therefore, we collect by collecting "[ha][ha]" and "[pa][pa]" data.

Group 17: [mo][to][mo][to], [to][mo][do][mo]

There is a high possibility that the discrimination between "[mo]" and "[to]" will be difficult. We investigate whether it is possible to discriminate between "[to]" and "[mo]" using a series of fingerspelling tasks where it is possible to discriminate only one character. It is essential to focus on the speed, as it may be affected by the person.

Group 18: [ta][ta], [ta][da], [da][da]

We investigate three forms of "[ta][da]" two consecutive forms, i.e., the alternating forms of "[ta][da]" and "[da]" and two consecutive forms of "[da]." We investigate whether the the system can distinguish between "[ta]" and "[da]" as "[ta]" and "[da]" must each be expressed once. The possibilities of misrecognizing "[ta][ta]" as "[ta][da]" or "[da][da]" and vice versa are investigated.

Group 19: [ne][sa][se][ru], [ne][za][sa][se][ru]

The first word contains "[sa]" and the second does "[za]." The task is to distinguish the difference between transition movements and dullness and through this experiment, we investigate whether the system can distinguish between "[sa]" and "[za][sa]."

Group 20: [he][ya], [pe][ya]

The possibility of misrecognizing "[he][ya]" as "[pe][ya]" exists. The necessary information on the hand movements is efficiently obtained by comparing the hand movements of transitions from "[he]" to "[ya]" and from "[pe]" to "[ya]." After collecting the data on "[he][ya]" and "[pe][ya]," we conduct a recognition experiment to determine whether human factors affect the results.

Group 21: [ko][so][gu], [go][zo][ku], [go][hu][ku], [ko][bu][ku]

"[go][zo][ku]" can be misidentified as "[ko][bu][ku]", "[ko][so][gu]", or "[go][hu][ku]." We believe that the inclusion of dynamic fingerspelling in the transition movements complicate the identification of the word as "[so]," "[bu]," or "[zo]." In this experiment, we investigate if we can discriminate between "[go][zo][ku]," "[ko][bu][ku]," "[ko][so][gu]," and "[go][hu][ku]."

Group 22: [wa][ro][shi], [wa][nu][shi]

"[wa][nu][shi]" may be misrecognized as "[wa][ro][shi]." We investigate whether the difference between "[ro]" and "[nu]" can be correctly identified. In particular, we investigate the degree of discrimination between the transitions from "[wa]" to "[ro]" and from "[wa]" to "[nu]."

Group 23: [he][ya][gi], [be][ki]

"[be][ki]" can be misrecognized as "[he][ya][ki]." In this experiment, we investigate whether "[be][ki]" is misrecognized as "[he][ya][ki]" by transitioning from "[be]" to "[ki]." We also consider the effects of different signers.

TABLE V. JAPANESE FINGERSPELLING COUNT

fingerspelling	a	i	u	e	o
count	1	6	6	2	1
fingerspelling	ka	ki	ku	ke	ko
count	3	6	5	1	2
fingerspelling	sa	shi	su	se	so
count	2	4	5	3	2
fingerspelling	ta	chi	tsu	te	to
count	2	4	8	1	7
fingerspelling	na	ni	nu	ne	no
count	1	3	1	2	4
fingerspelling	ha	hi	hu	he	ho
count	2	3	2	2	1
fingerspelling	ma	mi	mu	me	mo
count	3	4	2	2	2
fingerspelling	ya	yu	yo		
count	5	1	1		
fingerspelling	ra	ri	ru	re	ro
count	1	7	4	2	3
fingerspelling	wa	wo	nn		
count	2	1	6		
fingerspelling	ga	gi	gu	ge	go
count	1	2	1	1	2
fingerspelling	za	ji	zu	ze	zo
count	2	1	1	1	1
fingerspelling	da	di	du	de	do
count	2	1	1	1	1
fingerspelling	ba	bi	bu	be	bo
count	1	1	1	1	1
fingerspelling	pa	pi	pu	pe	po
count	1	1	1	1	2
fingerspelling	xya	xyu	xyo	xtsu	-[long vowels]
count	1	1	1	1	8

Group 24: [ho][e][ki], [bo][e][ki]

When transitioning from “[ho]” to “[e],” the direction of the hand changes depending on the person. When “[ho]” ends, the hand turns to the right to express it. At this time, it may become “[bo].” In this experiment, we analyze data and video recordings.

Group 25: [shi][se][i], [ji][ra][i], [shi][ze][i]

We investigate whether the three characters “[se],” “[ra],” and “[ze]” can be identified. We also investigate transition movements and dullness.

Group 26: [ka][chi][na][no][ri], [ga][i][su][u]

We investigate whether the system can correctly identify the differences between “[ka]” and “[ga],” “[chi]” and “[i],” “[na]” and “[su],” and “[ri]” and “[u].”

Table V shows the number of fingerspellings used for the 64 selected words.

B. Data collection

In the continuous fingerspelling recognition experiment, we collected data using a video recording of the hand and a sensor glove to record the bending and movements of the fingers, as shown in Figure 12. Consequently, we collected data from 33 people (nine people aged 20, 13 aged 21, eight aged 22, two aged 23, and one aged 24). As described above, there were a total of 64 words (see Table IV). There were five repetitions for each word. Eleven dimensions (five for the hand, three for the acceleration, and three for the gyro data) were used for each word. The number of samples was 120 sps × 8 s = 960 samples. The time to acquire a word was 8 s. Particularly, the time that the hand was placed on the desk before and after the word was expressed was 3 s and the time for expressing the word was 5 s, for a total of 8 s. For the word expression time in this experiment, we defined the maximum number of characters in an acquired word to be five, with each character to be expressed in approximately 1 s. A camera (*1 in Figure 12) was installed to record finger movements. During the data collection experiment, the collaborator (*2 in Figure 12) wore the sensor glove and expressed words displayed on an iPad. In addition, another camera (*3 in Figure 12) was installed to record the experiment. These experiments were conducted with approval from the Tsukuba University of Technology Research Ethics Committee (approval number: 2020-12)

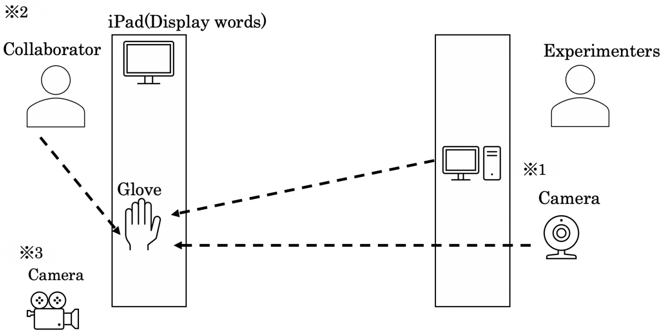


Figure 12. Data acquisition experiment of continuous Japanese fingerspelling.

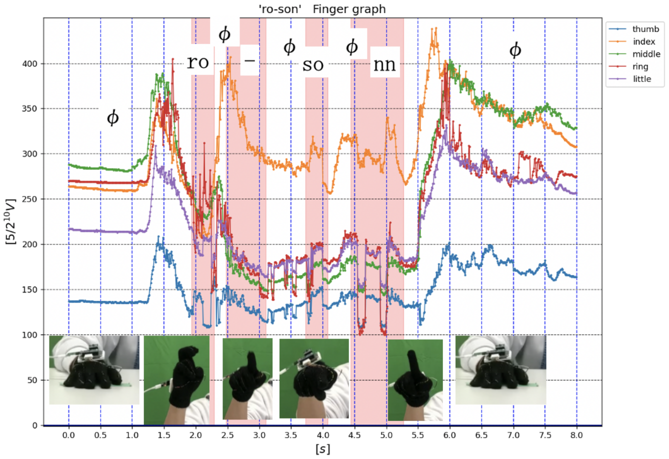


Figure 13. Five fingers of “ro-sonn.”

The acceleration and gyro data collected in the experiment were used to calculate the angle using the Madgwick filter. Next, we labeled the data using ELAN software, dividing the time for each one character. For the two instances in which the hand left and was placed on the desk, and for the transition movements between characters, the blank symbol “ϕ” was inserted. To visualize the flow in a graph, “[ro][-][so][nn]” is shown as an example in Figures 13–17.

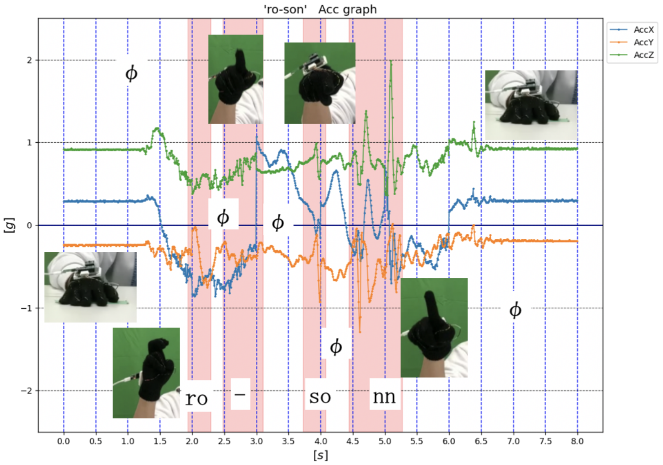


Figure 14. Accelerations of “ro-sonn.”

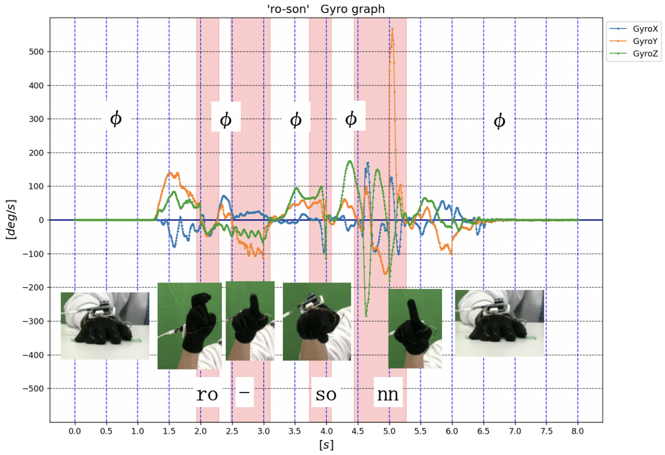


Figure 15. Gyros data of “ro-sonn.”

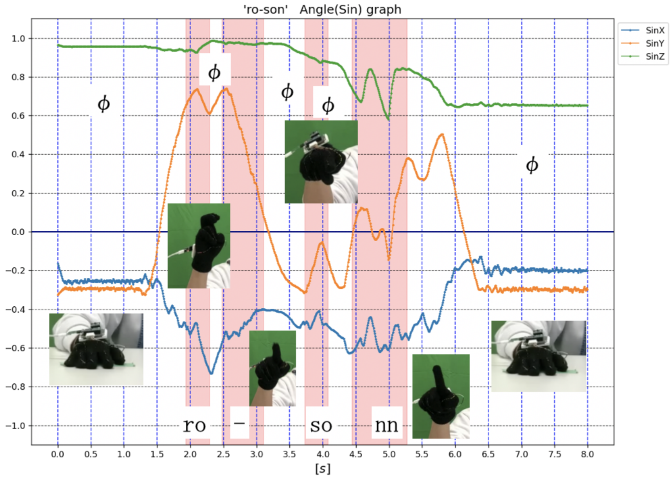


Figure 16. Sine of “ro-sonn.”

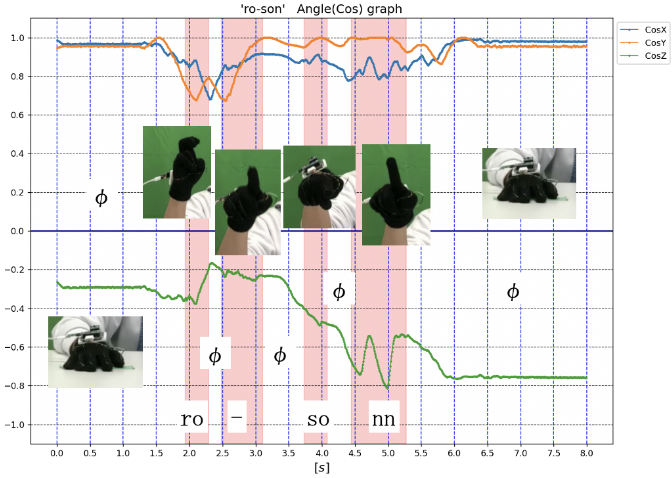


Figure 17. Cosine of “ro-sonn.”

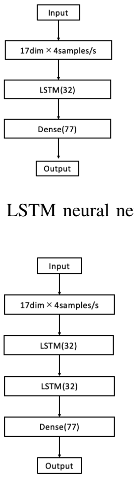
C. Construction of neural network using long short-term memory (LSTM)

Seven neural networks were constructed for the continuous fingerspelling recognition experiment. The LSTM [31] structure can use the short-term memory inside the network for a long time. LSTMs are often used to identify natural and speech processing language; they generally achieve high recognition rates. This experiment compares two networks, one with only the LSTM, and another with both CNN and LSTM, aiming to identify fingers, accelerations, gyro movements, and angles.

Figure 18a shows the neural network with the single LSTM as the baseline. The input data consisted of five dimensions of the hand, three dimensions of acceleration, three dimensions of gyro movements, and six dimensions of the angle, for a total of 17 dimensions × 32 samples (4 sps × 8 s). Next, the number of filters for the LSTM was 32 dimensions. In general, the number of filters is usually set to 16, 32, or 64 dimensions. Therefore, in this experiment, the number of filters of the LSTM was set to 32 dimensions, corresponding to the 77 output dimensions described below: the number of JF characters was 76, and the remaining one was “ϕ.” The latter was used to represents three situations: when the hand left the desk, when the hand was placed on the desk, and when the transition movements existed between characters.

Figure 18b shows a neural network with two LSTM layers. In this approach, the results are re-trains the results after passing through the first LSTM into the second LSTM. The number of filters in the LSTM was set to 32 dimensions, corresponding to the 17 dimensions of the hand, acceleration, gyro, and angle. The input data comprised five dimensions of the hand, three of the acceleration, three of the gyro, and six of the angle for a total of 17 dimensions × 32 samples (4 sps × 8 s). Finally, the data were output using the Dense operation (77 dimensions), i.e., the 76 JF characters and “ϕ”.

Figure 19 shows a neural network with two CNN layers and one LSTM layer. First, we input the data and then branch out into five dimensions × 32 samples (4 sps × 8 s) of the hand, three of the acceleration (4 sps × 8 s), three of the gyro (4 sps × 8 s), and six of the angle (4 sps × 8 s). After passing through the CNN (32 to 64 filters), these data were



(a) One LSTM neural network

(b) Two LSTM neural network

Figure 18. Two types of LSTM neural networks.

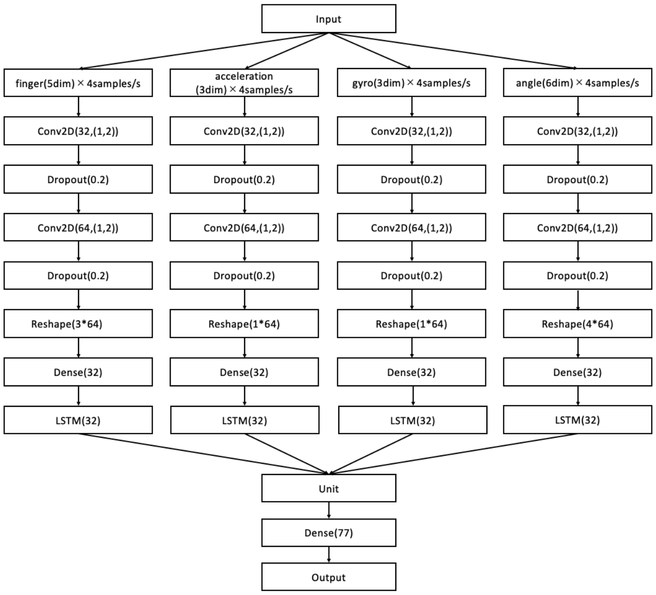


Figure 19. CNN-CNN-LSTM-unit neural network.

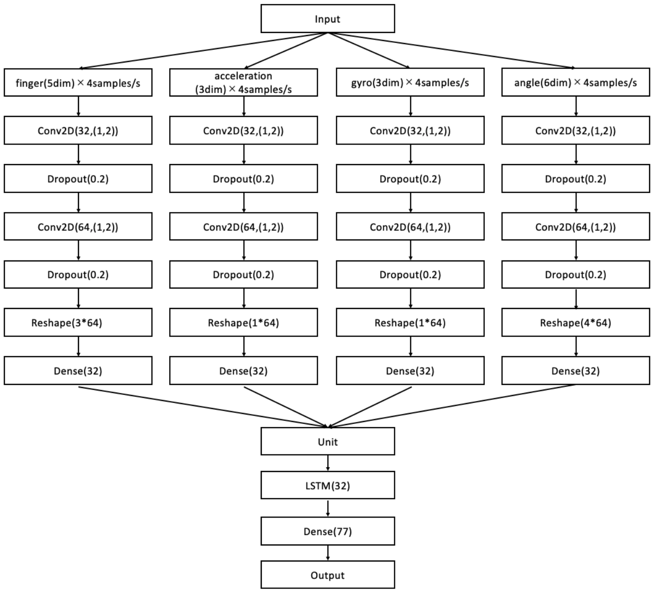


Figure 20. CNN-CNN-unit-LSTM neural network.

transformed to accommodate the Dense operation (32 nodes). Then, after passing through the LSTM (three nodes), they were combined. Finally, a Dense operation (77 nodes) corresponding to the number of characters ("ϕ" and the 76 JF characters) was applied to produce the output.

Compared to Figure 19, Figure 20 presents a different neural network in which the LSTM is added. The LSTM is interposed after combining the four datasets of the hand, acceleration, gyro, and angle, the LSTM is interposed. After inputting the data, we branched out into five dimensions × 32 samples of the hand (4 sps × 8 s), three of the acceleration (4 sps × 8 s), three of the gyro (4 sps × 8 s), and six of angle (4 sps × 8 s). These data were transformed after passing through the CNN (32 to 64 filters) to accommodate the Dense operation (32 nodes). They were then combined, and after passing through the LSTM (32 nodes), Dense operations (77 nodes) corresponding to "ϕ" and the number of characters in the JF were applied to produce the output.

Figure 21 shows the neural network with an additional LSTM for the finger, acceleration, gyro, and angle data. This re-trains the results after passing the first LSTM into the second LSTM. After inputting the data, we split the data into five dimensions × 32 samples (4 sps × 8 s of the hand, three of the acceleration (4 sps × 8 s), three of the gyro (4 sps × 8 s), and six of the angle (4 sps × 8 s). After passing through the CNN (32 to 64 filters), these data were transformed to accommodate the Dense operation (32 nodes). Then, after passing through the two LSTM layers (32 nodes), they were combined. Finally, a Dense operation (77 nodes) corresponding to the number of characters ("ϕ" and the JF characters) was applied to produce the output.

Figure 22 shows the neural network after combining the hand, acceleration, gyro, and angle data with another LSTM. First, after inputting the data, the five dimensions of hand and finger are split into 32 samples (4 sps × 8 s), three dimensions of the acceleration (4 sps × 8 s), three of the gyro (4 sps × 8 s), and six of the angle (4 sample/s × 8 s). These data were transformed after passing through the CNN (32 to 64 filters) to accommodate the Dense operation (32 nodes). They were then combined, and after passing through the two LSTM layers (32 nodes), Dense operations (77 nodes) corresponding to the number of characters in "ϕ" and the JF were applied to produce the output.

Figure 23 shows a neural network with one LSTM before and after merging. The LTSM before merging learns the results for the fingers, acceleration, gyro, and angle, after passing through the CNN. The LSTM after merging learns the results after merging the hand, acceleration, gyro, and angle datasets. First, the data ware input; then the network branched into 32

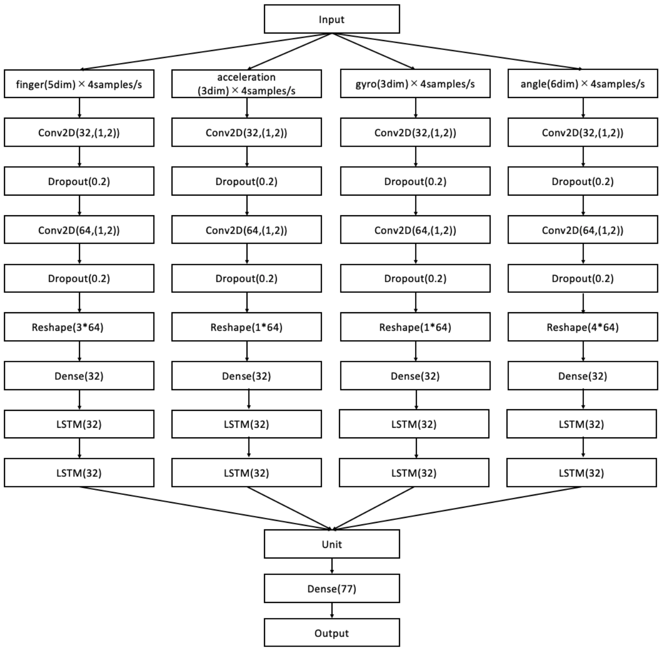


Figure 21. CNN-CNN-LSTM-LSTM-unit neural network.

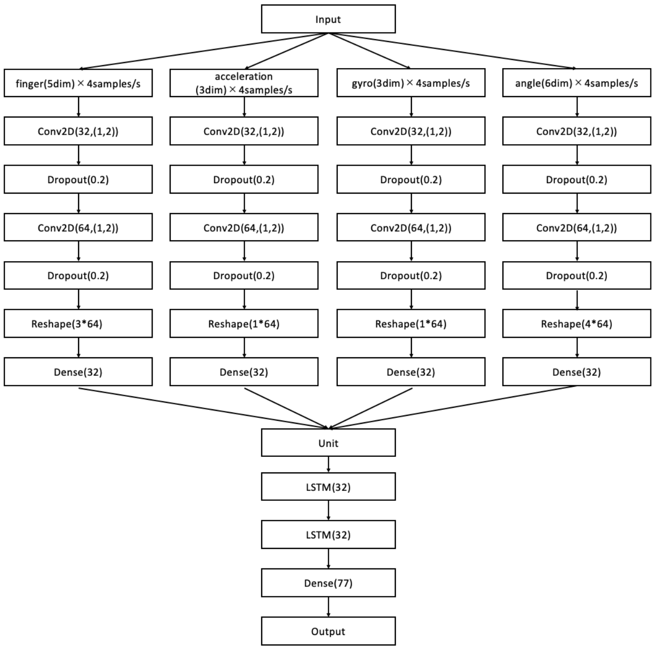


Figure 22. CNN-CNN-unit-LSTM-LSTM neural network.

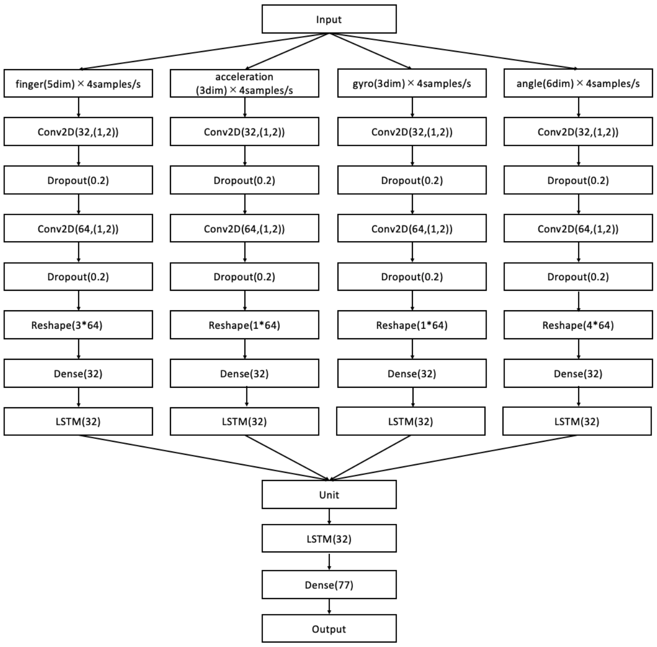


Figure 23. CNN-CNN-LSTM-unit-LSTM neural network.

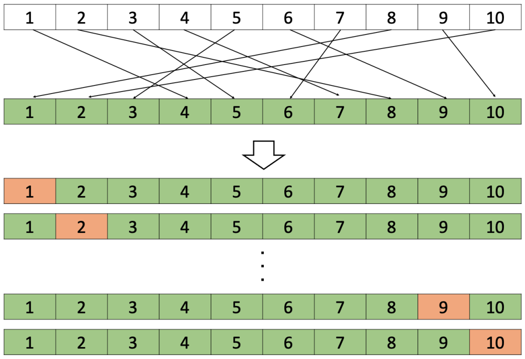


Figure 24. CNN-CNN-LSTM-unit-LSTM neural network.

samples of the five dimensions of the hand (4 sps × 8 s), three of the acceleration (4 sps × 8 s), three of the gyro (4 sps × 8 s), and six of the angle (4 sps × 8 s). These data were transformed after passing through the CNN (32 to 64 filters) to accommodate the Dense operation (32 nodes). Then, after passing through the LSTM (32 nodes), they were combined. Finally, after passing through another LSTM (32 nodes), a Dense operation (77 nodes) corresponding to the number of characters (“ϕ” and the JF characters) was applied to produce the output.

D. Evaluation experiments

We conducted evaluation experiments for each of the seven neural networks constructed in Figures 18a–23. The input data was shuffled and then divided into two parts, i.e., training and test data, using 10-fold cross-validation (see Figure 24).

E. Results and discussion

As shown in Table VI, the accuracy rates of the six neural networks, except for the one with the LSTM (one layer), are above 90%. Comparing the neural network with the two LSTM layers before merging to the neural network with the two LSTM layers after merging, the accuracy of the latter is approximately 1% higher than that of the former. In terms of good fit and recall, and the neural network with both a CNN and LSTM obtains an approximately 90% better performance

TABLE VI. SEVEN NEURAL NETWORKS EXPERIMENT RESULTS.

Neural network	Learning data (%)	Validation data (%)
one LSTM	91.4	90.1
two LSTM	92.5	90.3
branch-CNN-CNN-LSTM-unit	95.1	91.7
branch-CNN-CNN-unit-LSTM	94.0	91.8
branch-CNN-CNN-LSTM-LSTM-unit	96.5	91.3
branch-CNN-CNN-unit-LSTM-LSTM	94.7	92.1
branch-CNN-CNN-LSTM-unit-LSTM	95.2	91.6

TABLE VII. FIVE-FOLD CROSS-VALIDATION RESULTS.

k	Learning data (%)	Validation data (%)
1	95.0	92.4
2	94.7	92.0
3	94.9	92.4
4	94.5	91.8
5	94.4	91.8
Average	94.7	92.1

than the neural network with only one LSTM. That is, the neural network using both a CNN and LSTM has a higher accuracy rate. The higher accuracy may be owing to have been obtained because of the branching of the fingers, acceleration, gyro, and angle datasets and the detection of the feature points from the input data using CNNs. Therefore, we analyze the branch→Conv2D→Conv2D→conjoin→LSTM→LSTM neural network (see Figure 22) with the highest accuracy among the seven neural networks. Table VII summarizes the results of the five-fold cross-validation recognition experiments.

In this experiment, macro-averages are obtained for multi-class classification. Precision and recall are calculated using true positive (TP), false positive (FP), and false negative (FN) as shown below. The F-measure is the harmonic mean of the two values.

$$Precision = \frac{TP}{TP + FP} \tag{4}$$

$$Recall = \frac{TP}{TP + FN} \tag{5}$$

$$F_{measure} = \frac{2 * Precision * Recall}{Precision + Recall} \tag{6}$$

Table VIII shows that the precision, recall, F-measure for macro-averages, and F-measure for micro-averages were 67.8%, 62.3%, 64.7%, and 92.1%, respectively. Table X shows the fingerspellings ranked as ordered from the smallest F-measure. The precision for "ϕ" is 95.6%, the recall is 96.6%, and the F-measure is 96.1%, i.e., relatively high value. "ϕ" may have been misrecognized as "[te]" while the fingers were motionless during data collection. The F value (35.1%) is the smallest for "[te]" among the 77 types of data. The precision is 43.1%, and the recall is 29.6%. We confirmed that "[te]" was mistaken as "[de]," "[wa]," and "ϕ" using a confusion matrix.

The F-measure of "[ho]" is 38.5%. We confirmed that "[ho]" is included in "[bo]" and "[po]" using a confusion matrix. "[ho]" is expressed with the front of the hand forward, and the space between the five fingers close together. For "[bo]", right-handed users move their fingers to the right, and left-handed users to the left after expressing "[ho]." "[po]" is expressed by fingers moving upward after expressing "[ho]." The misrecognition of "[ho]" as two characters, "[bo]" and "[po]," owes to the similarities in hand movement.

TABLE VIII. MACRO-AVERAGE AND MICRO-AVERAGE

macro			micro
‘ Precision(%)	Recall(%)	F-measure(%)	F-measure(%)
67.8	62.3	64.7	92.1

The F-measure of "[chi]" is 41.4%. The confusion matrix confirms that "[chi]" is included in "[di]," "[tsu]," "[du]," and "[xtsu]." The "[chi]" is expressed with the thumb touching the index, middle, and ring fingers and the little finger extended. For "[di]," right-handed users move their hand to the right, and left-handed users to the left, after expressing "[chi]". The differences in hand movement cause misrecognition. The misrecognition of "[tsu]," "[du]," and "[xtsu]" is caused by the ring finger being extended for "[chi]" or not.

The F-measure of "[pe]" is 45.9%. "[pe]" is included in "[he]" and "[be]," as confirmed using a confusion matrix. "[he]" is expressed with fingers pointed downward, with the thumb and little finger extended and the other three fingers flexed. "[pe]" is expressed with the fingers moving upward after expressing "[he]." To express "[be]," right-handed users move their fingers to the right, and left-handed users to the left, after expressing "[he]." "[pe]" is misrecognized as two characters, "[he]" and "[be]", because the first parts of "[pe]" and "[be]" are expressed the same as "[he]," resulting in misrecognition owing to the similarities in hand movements.

The F-measure of "[du]" is 46.2%, and the confusion matrix confirms "[du]" is included in "[tsu]" and "[di]." "[tsu]" is expressed with the thumb touching the index and middle fingers and the ring and little fingers extended. Right-handed users express "[du]" by moving their fingers to the right, and left-handed users to the left, after expressing "[tsu]." "[du]" is misrecognized as "[tsu]" owing to movement similarities.

The F-measure of "[xyo]" is 77.9%. A confusion matrix confirms that "[yo]" is included in "[yo]." For "[yo]," only the thumb is flexed, and the other four fingers are extended. Therefore, the misrecognition of "[xyo]" as "[yo]" is caused by hand movement problems.

The F-measure of "[nn]" is 79.1%. We confirmed that "[nn]" is included in "[so]" and "[-](long vowel)" using a confusion matrix. For "[nn]", the index finger is extended to express a writing image, such as "[nn]" in katakana. "[so]" is expressed with the index finger extended downward and slightly diagonal. The misrecognition of "[nn]" as "[so]" is caused by the similarities in hand movement. T he "[-]" sound is represented by the index finger extending and moving up and down. "[nn]" is misrecognized as a "[-]" because the movement required to express "[nn]" halfway resembles the movement required to express "(-)".

The F-measure of "[mu]" is 80.2%. A confusion matrix confirms that "[mu]" is included in "[ku]," using a confusion matrix. "[mu]" is expressed with the thumb and index finger extended. "[ku]" is expressed with all five fingers extended and in close contact (excepting the thumb). "[mu]" is misrec-

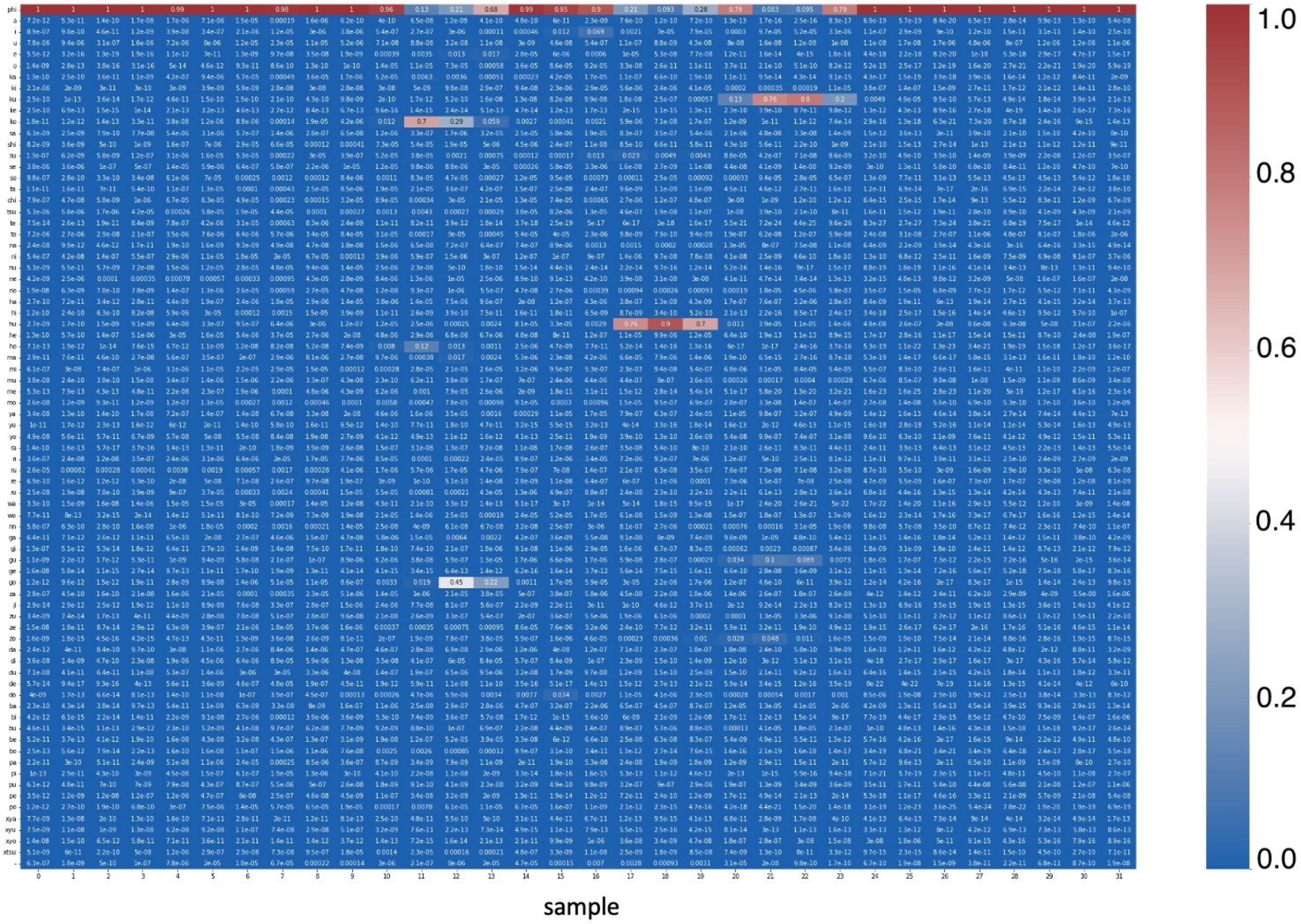


Figure 25. Recognition time series of “gohuku.”

TABLE IX. “gohuku” accuracy(%)

sample	9	10	11	14	15	16	18	19	20
ko	69.0	13.0	1.1	0.0	0.0	0.0	0.0	0.0	0.0
go	3.9	77.0	83.0	0.0	0.0	0.0	0.0	0.0	0.0
hu	0.0	0.0	0.0	19.0	84.0	1.8	0.0	0.0	0.0
bu	0.0	0.0	0.0	1.0	1.0	22.0	0.0	0.0	0.0
ku	0.0	0.0	0.0	0.0	0.0	0.0	0.0	67.0	90.0
phi	19.0	7.5	14.0	79.0	12.0	76.0	95.0	32.0	9.3

ognized as “[ku]” owing to the positions of the middle, ring, and little fingers.

The F-measure of “[mi]” is 82.0%, i.e., the highest among the 76 characters other than “ϕ”. The confusion matrix confirms that “[mi]” is included in “[shi].” The “[mi]” character is expressed with the index, middle, and ring fingers extended to the left for right-handed users and to the right for left-handed users. The index and middle fingers are extended to express “[shi]” with the right-handed fingers pointing left and left-handed fingers right. The misrecognition of “[mi]” as “[shi]” is caused by the position of the thumb and ring finger.

As an example of a problem in hand movement, we take the word “[go][hu][ku].” When users attempt to express “[go],” they may first express “[ko],” and we assume that it is recognized correctly. As an example, the recognition result of “[go][hu][ku]” is shown in the Figure 25. In addition, Table IX shows accuracy rate (unit: %) for each sample of “[go][hu][ku].” In this case, the user expression of the word “[go]” first expresses “[ko]” using static fingerspelling, and then performs an action. That is, the first movement of “[go]” is regarded as a static fingerspelling and become “[ko][go].” To recognize “[go]” clearly, specifying the range of time from when the finger begins moving after expressing “[ko]” to when the movement ends is necessary. In addition, it is necessary to insert a new transition movement between “[ko]” and “[go].”

V. CONCLUSIONS AND FUTURE WORK

In this study, to realize smooth communication between DHH and hearing people, we adapted a lightweight sensor

TABLE X. FINGERSPELLING RANK FROM THE LEAST F-MEASURE

Rank	JF	Data	Precision(%)	Recall(%)	F-measure(%)
1	te	159	43.1	29.6	35.1
2	ho	182	41.7	35.7	38.5
3	chi	676	43.9	39.2	41.4
4	pe	245	58.1	38.0	45.9
5	du	255	47.3	45.1	46.2
6	ko	360	55.5	44.7	49.5
7	di	259	53.0	47.1	49.9
8	pi	241	63.8	42.3	50.9
9	ji	256	65.1	42.2	51.2
10	so	338	55.7	47.3	51.2
11	ga	261	54.7	48.7	51.5
12	bo	252	60.6	48.8	54.1
13	ka	555	57.0	52.3	54.5
14	tsu	1801	58.7	51.5	54.8
15	ne	356	60.8	50.6	55.1
16	hu	367	56.1	55.3	55.7
17	ni	546	54.7	58.4	56.5
18	ro	509	61.9	52.7	56.9
19	ha	639	60.6	53.7	56.9
20	he	394	56.0	58.1	57.0
21	pu	242	62.1	52.9	57.1
22	hi	553	57.2	57.7	57.4
23	me	328	63.5	53.7	58.2
24	pa	437	65.9	52.6	58.5
25	xtsu	701	65.2	53.4	58.7
26	po	485	63.8	51.3	58.8
27	wo	273	63.6	56.4	59.8
28	wa	382	61.7	60.2	60.9
29	o	264	66.8	58.0	62.1
30	na	154	60.5	65.6	62.9
31	ra	189	70.6	57.1	63.2
32	su	1082	63.6	63.1	63.4
33	nu	201	70.3	57.7	63.4
34	ma	543	62.6	65.7	64.2
35	shi	863	67.2	61.4	64.2
36	de	222	70.9	60.4	65.2
37	bi	244	70.0	61.1	65.2
38	zo	248	66.0	64.9	65.4
39	go	516	67.1	64.5	65.8
40	ru	1305	65.2	67.1	66.1
41	a	170	70.4	62.9	66.5
42	ta	606	72.9	61.6	66.7
43	xya	268	69.5	64.6	66.9
44	sa	387	68.8	65.4	67.0
45	se	563	68.0	67.3	67.7
46	yu	195	70.6	65.1	67.7
47	e	365	68.3	67.4	67.9
48	ke	168	67.8	69.0	68.4
49	ba	260	68.5	68.5	68.5
50	gu	284	72.8	66.9	69.7
51	bu	260	77.5	63.5	69.8
52	yo	188	73.2	69.7	71.4
53	da	761	72.5	70.8	71.7
54	be	207	76.5	67.6	71.8
55	xyu	488	75.3	69.5	72.3
56	mo	847	75.3	69.8	72.4
57	ze	260	75.8	70.0	72.8
58	ya	1195	74.8	73.8	74.3
59	no	976	75.9	72.8	74.3
60	ki	1374	75.6	73.1	74.3
61	zu	275	76.5	72.4	74.4
62	u	1884	75.8	73.0	74.4
63	re	353	74.3	75.4	74.8
64	za	473	79.5	71.5	75.3
65	do	259	78.9	72.2	75.4
66	ge	228	80.8	71.9	76.1
67	to	1738	78.2	74.3	76.2
68	gi	550	76.2	76.7	76.4
69	i	1428	77.3	76.3	76.8
70	-(long vowel)	2115	75.4	79.0	77.1
71	ri	1890	79.3	76.3	77.8
72	xyo	259	782.2	77.6	77.9
73	ku	1263	79.4	77.7	78.5
74	nn	2878	81.7	76.6	79.1
75	mu	494	80.7	79.8	80.2
76	mi	1021	82.7	81.6	82.0
77	phi	252947	95.6	96.6	96.1

glove, developed an effective CNN model, implemented a JF recognition system, and evaluated the performance of the developed system. JF data collection experiments with 20 participants and 76 target JF characters were repeated five times. Data were acquired at 200 sps for 11 input dimensions. Angle data were transformed by applying a Madgwick filter to gyro readings and were converted into sine and cosine spaces, thereby increasing the total number of input dimensions to 17. However, the data acquired at 200 sps contained various issues, such as noisy signals. To solve this problem, we calculated the moving averages to reduce the frequency to 4 sps. Finally, a 20-fold cross-validation evaluation was conducted. The average recognition rate was approximately 70.0%, and the maximum recognition rate was approximately 75.5%. We determined that the variance in the flexion and direction of the fingers was a significant cause of misrecognition.

We then described the results of the continuous fingerspelling recognition experiment. In daily life, finger flexion, extensions, hand directions, and movements vary considerably among people learning sign language, people using ESJ, and people using JSL. Therefore, we proposed a dataset to exploit the characteristics of JF and selected 64 words. Then, we conducted a data collection experiment. For each of the 64 words, 11 dimensions (hand: five dimensions, acceleration: three dimensions, gyro: three dimensions) were input for eight s (120 sps × 8 s = 960 samples). The data and video collections were repeated five times. Then, using the acceleration and gyro data, the angles (three dimensions) were calculated using the Madgwick filter and converted to sine and cosine values. Six dimensions were added, bringing the total number of dimensions to 17, including those of the fingers (five dimensions) and accelerations and gyro (six dimensions). Next, the data was reduced by setting the average to 32 samples (4 sps × 8 s). Finally, a discrimination experiment was conducted. We compared two neural networks, one with only an LSTM and another with both CNN and LSTM. For the neural network using both CNN and LSTM, the evaluation experiment was conducted by splitting the hand, acceleration, gyro, and angle data, and passing each through the neural network. Consequently, micro F-measure of 92.1% was obtained for the neural network using the CNN and LSTM. Although we solved the calibration problem, a hand adhesion issue remained. Furthermore, distinguishing between static and dynamic fingerspelling based on hand motion became difficult.

Thus, this system had two main problems: hand-finger adhesion and distinguishing between static and dynamic fingerspellings. In the continuous fingerspelling recognition experiment, the accuracy rate of the finger characters decreased owing to the large number of instances "ϕ". To obtain a high discrimination rate in fingerspelling recognition, we must expand the data on fingerspellings and collect more data. The amount of data for "ϕ" is considerably larger than that of the 76 characters of JF; more data facilitate distinguishing between static and dynamic fingerspelling. Distinguishing between "[ko]" and "[go]" became particularly difficult; thus, we must contemplate constructing a system that considers any hand movement as a dynamic fingerspelling. The issue is also occurred in the single fingerspelling recognition experiments, e.g., distinguishing "[te]" from "[u]," "[te]" from "[tsu]," "[te]" from "[ru]," and "[te]" from "[wa]."

To further develop sign language recognition systems, three issues must be addressed. First, a large amount of speech data exists but with insufficient sign language data. To improve the accuracy rates using deep learning, collecting more sign language data is necessary. Second, preparing data on JSL is also necessary. In daily life, variations in finger flexion, hand directions, and hand movements occur among people. In addition, we must develop a learning model suitable for JSL. A multimodal approach, concerning JSL, addresses the first issue by inputting different types of information such as finger flexion, hand directions, hand movements, and facial expressions to improve recognition. JSL uses fingers, hand directions, and hand movements as well as the upper body, head, face, and mouth to express. Therefore, constructing a suitable language model for JSL is necessary. We plan to development of a sign language recognition system able to address these three issues.

VI. Acknowledgment

This work was supported by JSPS KAKENHI Grant Number #19K11411 and the Promotional Projects for Advanced Education and Research in NTUT. We would like to thank Editage (www.editage.com) for English language editing.

References

[1] T. Tsuchiya, A. Shitara, F. Yoneyama, N. Kato, and Y. Shiraishi, “Sensor glove approach for japanese fingerspelling recognition system using convolutional neural networks,” in Proceedings of The Thirteenth International Conference on Advances in Computer-Human Interactions (ACHI 2020), 2020, pp. 152–157.

[2] “UDtalk,” 2015, URL: https://udtalk.jp/ [retrieved: December, 2022].

[3] “KoeTra,” 2015, URL: https://www.koetra.jp/en/ [retrieved: December, 2022].

[4] “speech-to-text,” 2019, URL: https://cloud.google.com/speech-to-text [retrieved: December, 2022].

[5] S. Gattupalli, A. Ghaderi, and V. Athitsos, “Evaluation of deep learning based pose estimation for sign language recognition,” in Proceedings of the 9th ACM International Conference on PErvasive Technologies Related to Assistive Environments, 2016, pp. 1–7.

[6] M. A. Jalal, R. Chen, R. K. Moore, and L. Mihaylova, “American sign language posture understanding with deep neural networks,” in 2018 21st International Conference on Information Fusion (FUSION). New York, NY, USA: IEEE (Institute of Electrical and Electronics Engineers), July 2018, pp. 573–579.

[7] G. Luzhnica, J. Simon, E. Lex, and V. Pammer, “A sliding window approach to natural hand gesture recognition using a custom data glove,” in 2016 IEEE Symposium on 3D User Interfaces (3DUI). New York, NY, USA: IEEE (Institute of Electrical and Electronics Engineers), March 2016, pp. 81–90.

[8] K. Murakami and H. Taguchi, “Gesture recognition using recurrent neural networks,” in Proceedings of the SIGCHI conference on Human factors in computing systems, 1991, pp. 237–242.

[9] N. Mukai, N. Harada, and Y. Chang, “Japanese fingerspelling recognition based on classification tree and machine learning,” in 2017 Nicograph International (NicoInt). New York, NY, USA: IEEE (Institute of Electrical and Electronics Engineers), June 2017, pp. 19–24.

[10] H. Hosoe, S. Sako, and B. Kwolek, “Recognition of jsl finger spelling using convolutional neural networks,” 05 2017, pp. 85–88.

[11] M. E. Cabrera, J. M. Bogado, L. Fermin, R. Acuna, and D. Ralev, “Glove-based gesture recognition system,” in Adaptive Mobile Robotics. World Scientific, 2012, pp. 747–753.

[12] C. K. Mummadi, F. P. P. Leo, K. D. Verma, S. Kasireddy, P. M. Scholl, and K. Van Laerhoven, “Real-time embedded recognition of sign language alphabet fingerspelling in an imu-based glove,” in Proceedings of the 4th International Workshop on Sensor-Based Activity Recognition and Interaction, ser. iWOAR ’17. New York, NY, USA: Association for Computing Machinery, 2017, pp. 1–6. [Online]. Available: https://doi.org/10.1145/3134230.3134236

[13] R. Takada, J. Kadomoto, and B. Shizuki, “A sensing technique for data glove using conductive fiber,” in Extended Abstracts of the 2019 CHI Conference on Human Factors in Computing Systems, ser. CHI EA ’19. New York, NY, USA: Association for Computing Machinery, 2019, pp. 1–4. [Online]. Available: https://doi.org/10.1145/3290607.3313260

[14] M. Kondo, N. Kato, K. Fukui, and A. Okazaki, “Development and evaluation of an interactive training system for both static and dynamic fingerspelling using depth image,” IEICE technical report, vol. 114, no. 512, 2015, pp. 23–28, (in Japanese).

[15] “5DT Data Glove 5 Ultra,” 2019, URL: https://5dt.com/ [retrieved: December, 2022].

[16] N. M. Kakoty and M. D. Sharma, “Recognition of sign language alphabets and numbers based on hand kinematics using a data glove,” Procedia Computer Science, vol. 133, 2018, pp. 55–62.

[17] T.-W. Chong and B.-J. Kim, “American sign language recognition system using wearable sensors with deep learning approach,” The Journal of the Korea Institute of Electronic Communication Sciences, vol. 15, no. 2, 2020, pp. 291–298.

[18] X. Yu, S. Liu, W. Fang, and Y. Zhang, “Research and discovery of smart dumb gloves,” in Journal of Physics: Conference Series, vol. 1865, no. 4. IOP Publishing, 2021, p. 042054.

[19] O. Glauser, S. Wu, D. Panozzo, O. Hilliges, and O. Sorkine-Hornung, “Interactive hand pose estimation using a stretch-sensing soft glove,” ACM Transactions on Graphics (TOG), vol. 38, no. 4, 2019, pp. 1–15.

[20] H. Zhou, W. Zhou, Y. Zhou, and H. Li, “Spatial-temporal multi-cue network for continuous sign language recognition,” in Proceedings of the AAAI Conference on Artificial Intelligence, vol. 34, no. 07, 2020, pp. 13 009–13 016.

[21] “PWTH-PHOENIX-Weather 2014-T),” 2019, URL: https://www-i6.informatik.rwth-aachen.de/ koller/RWTH-PHOENIX-2014-T/ [retrieved: December, 2022].

[22] B. Shi, A. M. D. Rio, J. Keane, D. Brentari, G. Shakhnarovich, and K. Livescu, “Fingerspelling recognition in the wild with iterative visual attention,” in Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV), October 2019.

[23] “Chicago Fingerspelling in the Wild Data Sets (ChicagoFSWild, ChicagoFSWild+),” 2019, URL: https://home.ttic.edu/ klivescu/ChicagoFSWild [retrieved: December, 2022].

[24] D. Bragg, O. Koller, N. Caselli, and W. Thies, “Exploring collection of sign language datasets: Privacy, participation, and model performance,” in The 22nd International ACM SIGACCESS Conference on Computers and Accessibility, ser. ASSETS ’20. New York, NY, USA: Association for Computing Machinery, 2020. [Online]. Available: https://doi.org/10.1145/3373625.3417024

[25] “TensorFlow,” 2019, URL: https://www.tensorflow.org [retrieved: December, 2022].

[26] “scikit-learn,” 2019, URL: https://scikit-learn.org/stable/index.html [retrieved: December, 2022].

[27] G. Hinton, N. Srivastava, and K. Swersky, “Lecture 6e-rmsprop: Divide the gradient by a running average of its recent magnitude. cousera neural networks machine learning, 2012.”

[28] S. O. H. Madgwick, A. J. L. Harrison, and R. Vaidyanathan, “Estimation of imu and marg orientation using a gradient descent algorithm,” in 2011 IEEE International Conference on Rehabilitation Robotics. New York, NY, USA: IEEE (Institute of Electrical and Electronics Engineers), June 2011, pp. 1–7.

[29] E. Bisong, “Google colaboratory,” in Building Machine Learning and Deep Learning Models on Google Cloud Platform. Springer, 2019, pp. 59–64.

[30] K. Kazama, Y. Horiuchi, S. Masayoshi, and S. Kuroiwa, “Continuous finger spelling recognition using kinect based on linguistic information,” IEICE technical report, vol. 117, no. 502, 2018, pp. 83–88, (in Japanese).

[31] S. Hochreiter and J. Schmidhuber, “Long short-term memory,” Neural computation, vol. 9, no. 8, 1997, pp. 1735–1780.