Korean Sign Language Recognition Using EMG and IMU Sensors Based on Group-Dependent NN Models

Seongjoo Shin
Department of Information and Communication Engineering
DGIST
Deagu, Korea
sj_shin@dgist.ac.kr

Youngmi Baek
Department of Information and Communication Engineering
DGIST
Deagu, Korea
ymbaek@dgist.ac.kr

Jinhee Lee
Convergence Research Center for Future Automotive Technology
DGIST
Deagu, Korea
jhlee07@dgist.ac.kr

Yongsoon Eun
Department of Information and Communication Engineering
DGIST
Deagu, Korea
yeun@dgist.ac.kr

Sang Hyuk Son
Department of Information and Communication Engineering
DGIST
Deagu, Korea
son@dgist.ac.kr

***Abstract*—Automatic sign language recognition systems can help many hearing and speech-impaired people communicate with the public. To recognize sign language, the system should first determine the shape of the hand and the movement of the arm. Since sign language consists of a sequence of movements, it is difficult to distinguish a certain gesture from gestures (movements). It also has various lengths of gestures. It is effective to make the fixed length input data (gestures) rather than predefine the length of each gesture for recognition. Furthermore, in order to improve recognition accuracy, the effective way is to exploit multiple heterogeneous sensors (both an electromyography (EMG) sensor and an inertial measurement unit (IMU) sensor) which can produce the redundant information to the same physical variable. Specially, we focus on the fact that EMG signals depends on physical features of people because the amount of muscle and the thickness of the fat layer are different for each person. To address these issues, we propose an automatic recognition method for Korean sign language, which based on a sensor fusion technology and group-dependent Neural Network (NN) models. Our approach on group-dependent NN models is to separate the models so that different people can use different models. Finally, the results of recognition show that the proposed method has high accuracy (99.13% of CNN without dropout and 98.1% of CNN with dropout).**

Keywords— sign language; hand gesture; electromyography; sensor fusion; Artificial Neural Network

I. INTRODUCTION

There are 2.5 million people with disabilities in Korea. Hearing-disabled and speaking-disabled persons mainly speak in Korean sign language in order to communicate with the public using it. The percentage of them using Korean sign language in the people with disabilities is currently about 10% [1]. Since the majority of the public still lack knowledge of Korean sign language, speech-impaired people and deaf people suffer from communicating in signs whenever help is needed. In this regard, an automatic sign language recognition (SLR) system helps them communicate with people conveniently. The system is supposed to automatically recognize certain gestures as an input and convert what they speak in sign language to natural language.

Fundamental technologies for accurate sign recognition are revealed in the many recent research related to hand gesture recognition. In order to recognize sign language accurately and efficiently through hand gestures, there are problems to be addressed. First, after performing one gesture of sign language, we should move our hands or arms to do another gesture in sign language. At that time, it is difficult to distinguish a certain gesture from gestures (movements). In order to determine the timing to end the given gesture and the timing to make the next gesture, we define a basic posture as the status of no moving during the period from the end of the current gesture to the beginning of the next gesture. It makes a simple sequence of movements for further effective recognition of gestures in sign language. The second is related to various lengths of gestures representing the sign language. we address this problem by using zero-padding to make fixed length.

There is another issue to be considered. It is related to which sensors are proper to recognize hand gestures. People have a lot of biometric information such as electroencephalography (EEG), electrooculography (EOG), electrocardiography (ECG), electromyography (EMG) [2]. Especially, EMG is an electrical signal that represents the force measured when the muscle moves. Multiple EMG sensors are used to recognize hand shapes and movements. An inertial measurement unit (IMU)

sensor is also used to recognize the movement of the arm. The IMU is an electronic device that measures the angular rate of motion of the body and linear acceleration. This device actually holds a gyroscope and an accelerometer, and sometimes add magnetometer. The gyroscope measures angular acceleration of a rotating object in 3D space. Generally, it has at least one sensor for each of the three axes: pitch (nose up and down), yaw (nose left and right), roll (clockwise or counter-clockwise from the cockpit). There are already many applications using the IMU which is mounted on the wearable device [3] [4] . The redundant information obtained from multiple heterogeneous sensors might contribute to improve recognition accuracy. To take advantages of the sensor fusion technique, we exploit the IMU and EMG sensors in an armband mounted on the forearm. For performing the sensor fusion, pre-processing to measured raw data is required to use both the EMG sensor and the IMU sensor. This pre-processing consists of reducing the effect of noise of the raw sensor data, adjusting the sensor's sampling frequency and normalizing the range of sensor values.

The other is related to the versatility of a recognition model. It is a wonder that one user model is only adopted to all of the people who want to use it for sign language recognition. Individual data obtained from people have different features for two reasons: (1) the EMG sensor value depends on the thickness of the fat layer and amount of muscle and (2) the IMU value has different values depending on the length of arm and height. For this reason, one model learned with a small amount of data is less accurate. To address this issue, we create an individual recognition model for each of groups with similarity called *group-dependent model* so that each group of people can use the different user model to increase the accuracy of the user model.

In this paper, we propose an effective way to recognize Korean sign language, which based on a sensor fusion technology and Neural Networks (NN) as a data-based method. The main objective is to provide the optimized and accurate uer model for each group. For the group-dependent model, we find out the tested user model has high performance when the user model learned from data of another person is tested by using data of one.

The remainder of this paper is organized as follows. In the Section II, we give the brief explanation of hand gesture techniques. Section III describes how to recognize the sign language after wearing an armband and performing sign language and propose group-dependent NN models after we evaluate the performance of recognition. Finally, we provide conclusions and future work in Section IV.

II. BACKGROUND

A. Hand Gesture Recognition

Some prior research mainly deals with gesture recognition in terms of a human-machine/service interface [2] [3] [5]. They focus on how to identify the given gestures and understand the meaning of them through various sensors in order to control the electronic devices or provide the information required for applications.

Costanza *et al.* have studied a hand gesture technique to control some electronic devices in the smart home. Although it aims to minimize computational complexity and provide robust recognition without calibration for every user, only pre-defined gestures are detected by using the muscular contraction. Their method is not related to movements at all [5]. In [6], they focus on improving the human acceptance in augmented reality (AR), considering the execution time. Unfortunately, in their recognition, any movement of the hands and arms is not allowed.

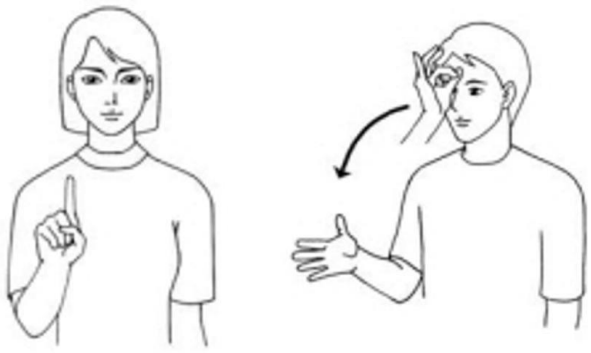
Rahman *et al*. have proposed a recognition scheme by identifying a valid sequence of movements [7]. The implemented system supports a low mobility since their recognition is performed at the certain area where the equipment is stationary. Radkovski and Strizke have proposed a free-hand interaction method between a device and a user in AR [8]. It not only requires many sensors for the landmarks but also is capable of the limited portability.

Recently, many studies have used machine learning algorithms such as support vector machine (SVM) [9], hidden Markov model (HMM) [10] [11], and Neural Network (NN) [12] to classify gesture. NN, called deep learning, is one of the most popular approaches for classifier, including Convolutional Neural Network (CNN) and Long-Short Term Memory (LSTM), in recent years. The CNN architecture is designed to recognize input data with a 2D structure, and many researchers employ CNN to study camera data such as pedestrian detection [13]. This technique requires a large amount of training so that it typically makes one model. The output of the EMG depends on physical characteristics and behavior of individuals such as the amount of the user's muscle and fat, and how the person is moving. If one user model is used, the performance might be low because the physical characteristics are quite different for each person.

To address this problem, one study has have attempted to use sensor values and the estimated physical characteristics as a feature of input data to learn the user model [14]. In this case, overall performance has been improved, but some sign language is not identified. They try to estimate this user-dependent factors by observing one motion, the estimation error under the uncertainty and the noise might affect recognition performance.

B. Sensors for Hand Gestures

Hand gestures are divided into static and dynamic gestures as shown in Fig. 1 [10]. A static gesture is a gesture with no hand or arm movement, and the shape of the hand is only significant.



(a) (b)

Fig. 1. Static gesture and Dynamic gesture (a) "One" (b) "Sorry [15]

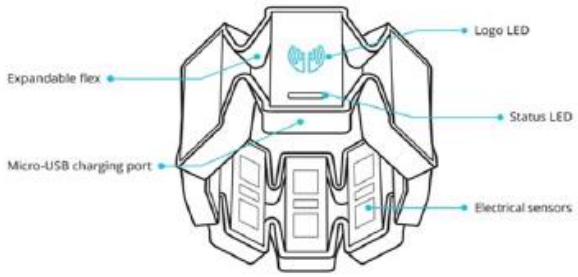


Fig. 2. Armband called MYO

In contrast, a dynamic gesture is a gesture with hand or arm movements. In many research, hand gesture recognition technologies mainly use sensors such as cameras, EMG, glove sensor, and IMU. The camera sensor can understand gestures by recognizing our arms or fingers [16], [17]. The glove sensor includes IMU and flex sensors, which can track hand motions. Camera and glove equipped with bending sensors have a high recognition rate, but there are some limitations and disadvantages. If using a camera sensor, it is difficult to recognize hand gestures outdoors in relation to implementation and weather issues. If using a glove sensor, people have to wear a cumbersome glove. Lightweight wearable devices such as smartwatches and armbands can get rid of this inconvenience and improve mobility and portability [18], [19], [20].

III. Methodology

In this section, we introduce an effective way to recognize Korean sign language. In order to design the user model having better performance in terms of accuracy, we perform a series of operations: (1) feature extraction, (2) pre-processing the values measured from sensors, (3) generation of architectures for the CNN and LSTM models, (4) learning and testing the architectures by using cross-validation.

A. Feature Extraction

We use an armband called MYO to extract the data, which is commonly used to read motions and control gestures. The MYO's SDK (Software Development Kits) allows us to easily retrieve raw data from MYO's sensors. The armband has 8 stainless steel EMG sensors and a nine-axis IMU. The IMU of the MYO contains three-axis gyroscope, three-axis accelerometer, three-axis magnetometer as mentioned above. The armband transmits the sensor value via Bluetooth. Fig. 2 shows the MYO device which is used for the automated Korean sign recognition system called *pocket assistant* in our work. The armband is mounted on the forearm and can recognize five hand shape as shown in Fig. 3 [21].

To effectively recognize gestures corresponding to sign language, we define a basic posture making a simple sequence

Fig. 3. Five hand gestures recognized by MYO

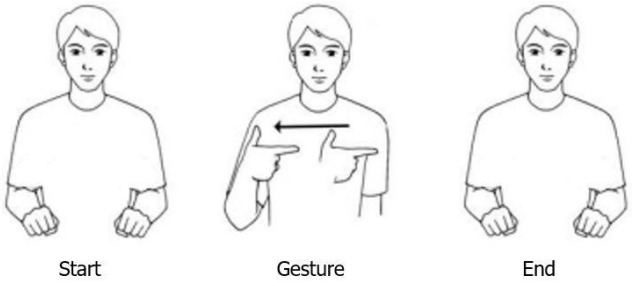


Fig. 4. Predefined sequence (one unit) for understanidng sign language

of movements. The defined basic posture describes the status of no moving in order to distinguish the current gesture from the next gesture. Fig. 4 presents the predefined sequence consisting of three postures (Start, Gesture, and End) to start, execute, and end a certain movement for one sign recognition, respectively. In the basic posture, the Start posture is the same as the End posture. The people to communicate in sign language must first drop their hands making a fist to waist high before speaking in sign language as shown in Fig. 4. While movements representing words in sign language constantly occur during a given time, they have the different lengths. It is important to find out the starting and ending points of a gesture in sign language. In the proposed systems, a gesture for one sign therefore starts from the basic posture and returns to the basic posture after it finishes.

The IMU's accelerometer is used to determine the starting and ending points. To reduce the effect of noise in the accelerometer, we adopt a moving average approach using the last *n* values obtained from each axis of the accelerometer. The moving average, $\overline{ACC}_{\psi}(m)$, is given in equation (1).

$$\overline{ACC}_{\psi}(m) = \frac{1}{n}\sum_{i=0}^{n-1} V_{\psi}(m-i) \quad (1)$$

where $\psi \in \{x, y, z\}$ denotes each axis of the accelerometer, m is the m-th average value to the given axis ψ, and $V_{\psi}(j)$ represents the *j-th* value of the axis ψ. We set *n* for 10 as the length of the moving average period. To minimize the effect of the change as a function of *n*, we approach the determination of *n* empirically by monitoring the result of a change of the *n* value.

Fig. 5. Starting and ending points for one gesture

A gesture can start at any time when the value of $\overline{ACC}_{\psi}(m)$ is greater than the upper bound (a certain positive value) or less than the lower bound (a certain negative value). The ending point for the gesture is the time when all values of $\overline{ACC}_{\psi}(m)$ for all ψ are kept within these boundaries for ten consecutive times. This is because all values of $\overline{ACC}_{\psi}(m)$ derived from any gesture in sign language can be near the boundaries. Fig. 5 shows the starting and ending points for one gesture.

Six healthy men without musculoskeletal involvement participate in speaking in the sign language. They wear the MYO armband on their right arm and make predefined gestures of the hands to generate their sign data we should gather. They watch the sign language video 2 or 3 times first and then speak in sign language without any instruction from the video. 30 different gestures in Korean sign language are identified by the proposed recognition system and are selected from one-handed gestures listed in descending sort order by the number of hits to a search term in the Korean sign language dictionary [15]. The dataset is constructed with training data of 72% (50 samples per class) and testing data of 28% (20 samples per class).

B. Preprocessing and Acquistion

First of all, in the EMG sensor, since the measurement of the raw signal is varied within the range from -128 to 128, it is converted into the absolute value, $E_{c,i} = |E_c^i|$, of the magnitude of the muscle force, where E_c^i is the i-th EMG sensor value from channel c ($c \leq 8$). Second, when we exploit the raw data of the MYO armband with the EMG sensor and the IMU sensor, there are two problems to be addressed. The EMG sensor and the IMU sensor provide the different sampling frequencies of approximately 200 Hz and 50 Hz, respectively. For sensor synchronization between the different sensors, the lowest sampling frequency is selected based on simple calibration to match those of two sensors. It is also necessary to adjust the values of physical variables on a common scale prior to designing a user model since both sensors measure the different physical variables.

In our work, z-scores and min-max normalization are employed and we try to find out how much influence each normalization method has on the performance of recognition. The raw values of sensors are transformed into z-scores $z'_{\psi,i}$, and $z_{c,i}$ by using equation (2).

$$z_{c,i} = \frac{E_{c,i} - \mu}{\sigma}, and \qquad z'_{\psi,i} = \frac{M_{\psi,i} - \mu'}{\sigma'} \tag{2}$$

where $E_{c,i}$ and $M_{\psi,i}$ are the i-th values of each channel ($c \leq 8$ and $\psi \leq 10$) of the EMG and the IMU respectively. In the case of the EMG sensor, μ is the mean of the values obtained from all the channels of the EMG sensor and σ is the standard deviation of those. A mean value μ' and the standard deviation σ' of the IMU sensor are derived from values obtained from gyroscope, accelerometer, and magnetometer, respectively.

Min-max normalization is an alternative method to z-score normalization, in which the raw values of sensors is scaled to a fixed range with 0 to 255. The raw values of both EMG and IMU sensors are adjusted to a common scale by using equation (3), respectively.

$$\widetilde{d_{c,\iota}} = \frac{E_{c,i} - S_{min}}{S_{max} - S_{min}}, and \qquad \widetilde{d'_{\psi,\iota}} = \frac{M_{\psi,i} - S_{min}}{S_{max} - S_{min}} \tag{3}$$

where $E_{c,i}$ and $M_{\psi,i}$ are the i-th values of each channel ($c \leq 8$ and $\psi \leq 10$) of the EMG and the IMU sensor respectively. S_{min} is 0 and S_{max} is 255.

After normalization of sensor values, zero-padding is performed in order to make fixed length data as shown in Fig. 6. Zero padding is a way to fill in the necessary parts with zeros and is often used in image processing and signal processing using filters in order to maintain the size of the image [22], [23]. Recurrent Neural Network (RNN) typically uses zero-padding to equalize input data of different lengths. One study using CNN also used zero-padding for natural language matching with different lengths [24]. In our work, the obtained data corresponding to one gesture varies in length. We observe that the length of the longest sample of the sensor is 192. Therefore, we simply set the fixed length to 200 which is more than long enough to perform max-pooling of the fixed window size as described in the next subsection.

In Fig. 6, the preprocessed data of individual gestures (corresponding to terms “Bee” and “Suddenly”) are visualized as 2D images. The input data corresponding to one gesture is constructed by stacking the eight channels of the EMG sensor, three channels of the accelerometer, three channels of the gyroscope, three channels and one weight of the magnetometer. Therefore our input data consists of the observed and preprocessed sensor data on the y-axis against the fixed length on the x-axis in the 200 * 18 matrix.

C. Creation of Architectures using Neural Networks

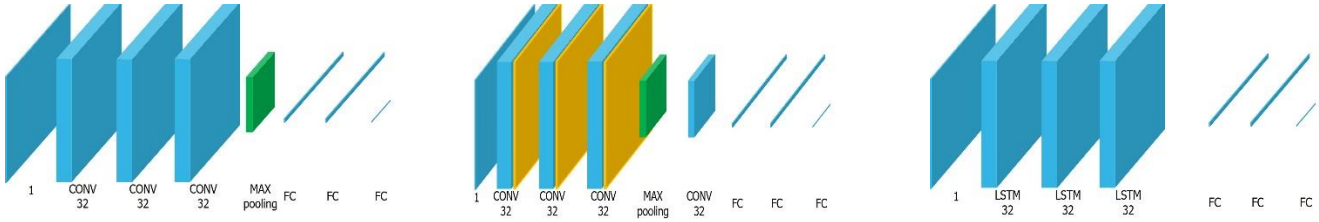
We design three NN architecture to be used for our pocket assistant. Specially, when CNN is applied to 2D image recognition, it achieves better performance. In order to take advantage of it, the best approach to exploit CNN is to express and transform the processed data as images first before CNN is applied to recognize the sign. In addition, as LSTM has shown good performance at speech recognition with variable length data these days, we evaluate the performance of our pocket assistant using CNN, comparing with that using LSTM.

A convolution is an operation that extracts a feature using a filter. CNN is a model that learns these filters. The CNN and

(a)

(b)

Fig. 6. Preprocessed input data of individual gestures as an image. (a) “Bee” (b) “Suddenly”



(a) A CNN architecture without dropout

(b) A CNN architecture with dropout (orange is dropout)

(c) A LSTM architecture
NN architectures for Korean sign recognition

Fig. 7. NN architectures for Korean sign recognition

LSTM architectures can be constructed with a simple solution to prevent it from over-fitting. A dropout solution is typically used to avoid over-fitting. When applying dropout, the architecture has higher performance [25].

Fig. 7 shows three architectures we devise: (a) a CNN architecture without dropout, (b) a CNN architecture with dropout and (c) a LSTM architecture. Dropout is a simple method to change the value to zero with a certain probability when forwarding the value of one neuron to the next layer [25]. Changing the value of a neuron to zero has the same effect as excluding the neuron. The CNN architecture without dropout consists of three convolution layers, one max-pooling, and three fully connected layers. The second is the CNN model with dropout. In the constructed LSTM as shown in Fig. 7(c) as well as two CNNs, the values of the hyper parameters used are specified as follows: 0,001 of the learning rate, 10 of the batch size, Relu for activation function, SGD for the optimizer, and 50 for an epoch. We use both TensorFlow and Keras as an interface to TensorFlow. TensorFlow is an open source library for machine learning that is built on Google and Keras is also a neural network API written in Python.

D. Group-Dependent NN Models

We first created the learning architecture of NNs against one subject's data, and then we have gotten the result of 99.6% by testing his test data. We compare the performance in terms of accuracy for all possible combinations of the two normalization (z-score and min-max) and the three architectures as shown in Fig. 7. Each architecture was trained with total training data of 9,000 and tested with 3,600 test data obtained from six subjects. As shown in Table I, the CNN architecture without dropout after z-score has the best performance in terms of accuracy. It is natural that the results of the architecture being tested for data of various subjects are lower than when only one subject is tested.

TABLE I. THE PERFORMANCE OF LEARNING ATCHITECTURES BASED ON NNS FOR UNDERSTANIDNG SIGN LANGUAGE

Number	Pre-Processing	NN model	Accuracy (%)
1	Z-score	CNN without dropout	64.6
2	Min-Max	CNN without dropout	58.0
3	Z-score	CNN with dropout	63.2
4	Min-Max	CNN with dropout	59.6
5	Z-score	LSTM	60.0
6	Min-Max	LSTM	11.3

This is because the people have the different physical characteristics and the same sign language can be differently expressed depending on personality.

We hypothesize that the two people have similar physical conditions, if the accuracy of the tested user model is high when the user model learned from data of one person is tested by using data of another person. We perform experiments to further analyze the performance (accuracy) of individual models learned from using data of six subjects which participate in training. The individual learned model from data of a certain person is called *user model*. The result is shown in Table II(a), (c) and (e) are processed by normalization of z-score, and the normalization used in Table II(b), (d), and (f) is min-max. Also, the user models used in Table II(a) and (b) CNN without dropout, the user models used in Table II(c) and (d) are CNN with dropout, and the user model used in Table II(e) and (f) is based on LSTM.

Accuracy with more than 70% achieved by each user model is marked in bold in Table II. From the results of Table II, it turns out that many user models using the CNN architecture with the z-transformation values achieve higher accuracy (in more than 70% accuracy) than those of it with the min-max normalization. In the cases of Table II(a), the results of applying the first-person data (Label Person 1) to the second user model and the third person data (Label Person 3) to the fourth user model are high (88.0% and 80.3% respectively). In the cases of Table II(c), the result of applying the first-person data (Label Person 1) to the second user model and the third person data (Label Person 3) to the fourth user model and the fifth person data to the sixth user model are high (88%, 84.3%, and 96% respectively).

These results support our view that the learned user model will support the wider representation of users if the user model achieves higher accuracy than another when individual user models are tested by using data of different people matched to each user model in a pairwise manner. In this regard, six subjects that participate in this experiment can be divided into four groups from the results of Table II. Label Person 1 and 2 belong to Group A. Label Person 3 and 4 belong to Group B. Label Person 5 belongs to Group C. Label Person 6 belongs to Group D. We perform the re-learning of the CNN architecture with z-score for each group because it shows many well-trained user models at the previous test. After that, we conduct a test on the user models learned from using data of each group. Form the results of Table III, we see that the individual user models for each group are more accurate than the user models for all people data such as results in row 1 and row 3 of Table I. Our experimental results are shown in Table III. The average

TABLE II. RESULTS OF THE TEST ON EACH USER MODEL LEARNED FROM ONE PERSON'S DATA.

(a) z-transformation

	User Model number	Person 1	2	3	4	5	6
CNN without dropout	1	**99.7**	30.7	14.3	24.8	19.8	20.5
	2	**88.0**	**99.3**	20.2	26.7	38.0	15.5
	3	62.8	**80.3**	**98.7**	53.3	37.3	16.2
	4	58.2	**76.3**	**80.3**	**99.3**	24.7	14.3
	5	58.7	**72.3**	64.2	69.8	**99.3**	26.3
	6	36.0	22.7	15.3	23.7	27.5	**98.5**

(b) Normalization from 0 to 255

User Model number	Person 1	2	3	4	5	6
1	**99.0**	16.2	5.7	17.5	16.5	7.7
2	48.2	**99.0**	15.8	25.3	20.2	17.2
3	18.2	45.8	**98.3**	31.5	23.7	9.7
4	41.5	56.3	51.5	**99.2**	17.5	9.8
5	53.2	45.8	41.3	55.5	**99.2**	26.3
6	33.2	23.0	20.2	27.5	69.2	**99.0**

(c)

CNN with dropout	1	**99.7**	28.0	17.8	22.3	24.7	16.3
	2	**88.0**	**99.5**	18.3	29.8	33.7	23.8
	3	60.7	**75.0**	**99.0**	46.5	41.3	18.5
	4	57.7	**73.5**	**84.3**	**99.5**	31.5	15.3
	5	66.3	**72.3**	62.0	74.3	**99.7**	33.3
	6	65.8	53.0	62.5	53.7	**96.0**	**99.0**

(d)

1	**99.8**	17.0	3.5	15.7	13.8	7.5
2	42.0	**99.2**	16.5	28.2	17.0	15.5
3	23.5	38.2	**99.0**	32.8	24.3	12.8
4	30.2	49.2	62.7	**99.3**	18.5	13.8
5	47.2	50.5	53.2	53.0	**98.7**	29.5
6	26.8	39.3	34.0	24.7	62.8	98.2

(e)

LSTM	1	**99.0**	20.3	8.0	22.8	22.3	18.0
	2	**73.7**	**99.5**	21.3	20.2	25.7	9.8
	3	20.2	24.5	98.7	33.8	24.3	18.5
	4	41.3	52.8	69.7	99.5	24.2	15.5
	5	53.2	52.7	48.5	55.5	99.8	41.3
	6	53.5	39.0	36.8	31.3	79.7	98.8

(f)

1	15.2	10.3	8.3	9.2	7.5	11.7
2	7.2	16.2	8.2	7.8	7.2	8.5
3	6.3	7.5	17.8	8.8	6.0	8.7
4	8.8	10.8	8.8	12.5	5.8	8.7
5	8.0	6.8	6.8	5.0	14.3	9.3
6	8.8	5.0	9.3	8.0	10.0	12.5

accuracy of our CNN model without dropout is 99.13% and the average accuracy of our CNN model with dropout is 98.1%.

IV. CONCULSION

In this paper, we present the effective way to recognize Korean sign language using EMG sensor and IMU sensor. In order to recognize sign language accurately and efficiently, there are several problems to be addressed. The first is to be difficult to distinguish a certain gesture from gestures (movements) representing sign language, and the second is related to various lengths of gestures representing the sign language. Therefore, we define the basic posture and use the moving average in order to determine the starting and ending points by using accelerometer values. In addition, to fix the length of the input data, zero padding is applied. The values of each sensor have been preprocessed for the fusion of EMG sensor and IMU sensor. The EMG sensor value has been converted to the absolute value. Sensing data of the EMG was down-sampled to the reading frequency of the IMU sensor. We apply each of z-score or min-max normalization to process the raw sensor values in order to increase overall accuracy. We also create recognition models based on each of CNN and LSTM. The learning architecture has three of NN layers and three fully connected layers.

Our results show that in terms of the accuracy, the performance of user models with z-score is higher than min-max normalization of the user models based on the same architecture. It turns out that many user models based on the CNN architecture show better performance than those based on the LSTM architecture for Korean sign language recognition. We think that the preprocessing method contributes to CNN much rather than LSTM. In addition, we see that the individual user models for each group are more accurate than user models for all people data when recognizing the Korean sign language. Therefore, we contend that to create the individual user model for each of groups with similarity is more effective than that of one user model learnt from data of all available people.

We plan to collect more data to learn user models and to be tested on them since the number of participants is too small to support our view. We also plan to compare results with other experiments. We expect that the individual user models for each of groups will have better performance. After that work, we plan to implement the pocket assistant as an automated Korean sign language recognition system in mobile devices such as smartphones and optimize it to be used outdoors in real time. It is worthwhile to provide our packet assistant for the people who want to communicate with or help the speech-impaired people but they have never learned how to speak sign language.

TABLE III. ACCURACY OF GROUP-DEPENDENT MODELS.

(a) CNN model without dropout

	Groups A	B	C	D
Accuracy(%)	99.83	98.91	99.3	98.5

(b) CNN model with dropout

	Groups 1	2	3
Accuracy(%)	98.8	98.6	97.1

ACKNOWLEDGMENT

This research was supported in part by the Global Research Laboratory Program (2013K1A1A2A02078326) through NRF, the ICT R&D program of MSIP/IITP (2014-0-00065, Resilient Cyber-Physical Systems Research) and the DGIST Research and Development Program (CPS Global Center) funded by the Ministry of Science, ICT & Future Planning.

REFERENCES

[1] "2016 Disablility Statistics," Korea Employment Agency for the Disabled, [Online]. Available: https://www.kead.or.kr/webzine/ibook/ttsbook/WEB/KEAD.html. [Accessed 9 2017].

[2] R. B. Reilly and T. C. Lee, “Electrograms (ecg, eeg, emg, eog),” *Technology and Health Care*, pp. 443-458, 2010.

[3] S. Wan and E. Foxlin, "Improved pedestrian navigation based on drift-reduced MEMS IMU chip," in *Proceedings of the 2010 International Technical Meeting of the The Institute of Navigation, San Diego, CA, USA*, 2010, pp. 220-229.

[4] J. Wu, L. Sun and R. Jafari, “A Wearable System for Recognizing American Sign Language in Real-Time Using IMU and Surface EMG Sensors,” *IEEE Journal of Biomedical and Health Informatics*, pp. 1281-1290, 2016.

[5] A. M. Rahman, M. A. Hossain and J. Parra, “Motion-path based gesture interaction with smart home services,” *ACM international conference on Multimedia*, pp. 761-764, 2009.

[6] S. Reifinger, F. Wallhoff, M. Ablassmeier, T. Poitschke and G. Rigoll, “Static and Dynamic Hand-Gesture Recognition for Augmented Reality Applications,” *International Conference on Human-Computer Interaction*, pp. 728-737, 2007.

[7] E. Costanza, S. A. Inverso and R. Allen, “Toward subtle intimate interfaces for mobile devices using an EMG controller,” *SIGCHI conference on Human factors in computing systems*, pp. 481-489, 2005.

[8] R. Radkowski and C. Stritzke, “Interactive Hand Gesture-based Assembly for Augmented Reality Applications,” 2012.

[9] M. Yoshikawa, M. Mikawa and K. Tanaka, "Real-time hand motion estimation using EMG signals with support vector machines," in *SICE-ICASE, 2006. International Joint Conference*, IEEE, 2006, pp. 593-598.

[10] X. Zhang, X. Chen, Y. Li, V. Lantz, K. Wang and J. Yang, "A framework for hand gesture recognition based on accelerometer and EMG sensors," *IEEE Transactions on Systems, Man, and Cybernetics-Part A: Systems and Humans*, vol. 41, no. 6, pp. 1064-1076, 2011.

[11] Z. Yang, Y. a. C. W. Li and Y. Zheng, "Dynamic hand gesture recognition using hidden Markov models," in *Computer Science & Education (ICCSE), 2012 7th International Conference on*, IEEE, 2012, pp. 360-365.

[12] M. R. Ahsan, M. I. Ibrahimy and O. O. Khalifa, "Electromygraphy (EMG) signal based hand gesture recognition using artificial neural network (ANN)," in *Mechatronics (ICOM), 2011 4th International Conference On*, IEEE, 2011, pp. 1-6.

[13] Y. Tian, P. Luo, X. Wang and X. Tang, "Deep learning strong parts for pedestrian detection," in *Proceedings of the IEEE International Conference on Computer Vision*, 2015, pp. 1904-1912.

[14] T. Matsubara and J. Morimoto, "Bilinear modeling of EMG signals to extract user-independent features for multiuser myoelectric interface," *IEEE Transactions on Biomedical Engineering*, vol. 60, no. 8, pp. 2205-2213, 2013.

[15] "Korean numeral words dictionary," National Institue of Korean Language, [Online]. Available: http://sldict.korean.go.kr/signhand/hand/main.do. [Accessed 9 2017].

[16] R. Hartanto, A. Susanto and P. I. Santosa, "Real time hand gesture movements tracking and recognizing system," in *Electrical Power, Electronics, Communications, Controls and Informatics Seminar (EECCIS), 2014*, IEEE, 2014, pp. 137-141.

[17] Z. Ren, J. Yuan and Z. Zhang, “Robust hand gesture recognition based on finger-earth mover's distance with a commodity depth camera,” *19th ACM international conference on Multimedia*, 2011.

[18] C. Xu, P. H. Pathak and P. Mohapatra, "Finger-writing with smartwatch: A case for finger and hand gesture recognition using smartwatch," in *Proceedings of the 16th International Workshop on Mobile Computing Systems and Applications*, ACM, 2015, pp. 9-14.

[19] Z. Lu, X. Chen, Q. Li, X. Zhang and P. Zhou, "A hand gesture recognition framework and wearable gesture-based interaction prototype for mobile devices," *IEEE transactions on human-machine systems*, vol. 44, no. 2, pp. 293-299, 2014.

[20] M. Sathiyanarayanan and S. Rajan, "MYO Armband for physiotherapy healthcare: A case study using gesture recognition application," in *Communication Systems and Networks (COMSNETS), 2016 8th International Conference on*, IEEE, 2016, pp. 1-6.

[21] “Myo,” [Online]. Available: https://www.myo.com/.

[22] D. Wang, N. Canagarajah, D. Redmill and D. Bull, “Multiple description video coding based on zero padding. In Circuits and Systems,” *IEEE International Symposium on Circuits and Systems*, 2004.

[23] J. Borkowski and J. Mroczka, “LIDFT method with classic data windows and zero padding in multifrequency signal analysis,” *Measurement*, pp. 1595-1602, 2010.

[24] B. Hu, Z. Lu, H. Li and Q. Chen, “Convolutional neural network architectures for matching natural language sentences,” *Advances in neural information processing systems*, pp. 2042-2050, 2014.

[25] H. Wu and X. Gu, "Towards dropout training for convolutional neural networks," *Neural Networks*, vol. 71, pp. 1-10, 2015.