

Word recognition and automated epenthesis removal for Indonesian sign system sentence gestures

Erdefi Rakun, I Gusti Bagus Hadi Widhinugraha, Noer Fitria Putra Setyono

Faculty of Computer Science, Universitas Indonesia (UI), Depok, Indonesia

Article Info

Article history:

Received Sep 27, 2021

Revised Mar 25, 2022

Accepted Apr 6, 2022

Keywords:

Epenthesis gesture

SIBI

Long short-term memory

Sign language recognition

Threshold conditional random field

ABSTRACT

This research focuses on building a system to translate continuous Indonesian sign system (SIBI) gestures into text. In a continuous gesture, a signer will add an epenthesis (transitional) gesture, which is hand movement with no meaning but needed to connect the hand movement of one word with the next word in a continuous gesture. Reducing the number of irrelevant inputs to the model through automated epenthesis removal can improve the system's ability to recognize the words in continuous gestures. We implemented threshold conditional random fields (TCRF) to identify epenthesis gestures. The dataset consists of 2,255 videos representing 28 common sentences in SIBI. The translation system consists of MobileNetV2 as a feature extraction technique, removing epenthesis gestures found by the TCRF, and a long short-term memory (LSTM) for the classifier. With the MobileNetV2-TCRF-bidirectional LSTM model, the best word error rate (WER) and sentence accuracy (SAcc) were 33.4% and 16.2%, respectively. Intermediate-stage processing steps consisting of sandwiched majority voting of the TCRF and the removal of word labels whose number of frames is less than two frames, along with LSTM output grouping, were able to reduce WER from 33.4% to 3.4% and increase SAcc from 16.2% to 80.2%.

This is an open access article under the [CC BY-SA](https://creativecommons.org/licenses/by-sa/4.0/) license.



Corresponding Author:

Erdefi Rakun

Faculty of Computer Science, University of Indonesia

Kampus UI, Depok 16424, Indonesia

Email: efi@cs.ui.ac.id

1. INTRODUCTION

The 2015 intercensal population survey projects the number of people with hearing impairment in Indonesia to be 3.35 percent of the total population by 2020 [1]. Deaf people are those who have limited verbal communication due to loss of hearing ability [2]. This verbal limitation causes them to adopt non-verbal communication or interaction that utilizes hand and body movements and facial expressions.

Sign language is a non-verbal communication practice representing words such as hand posture and lip movements [3]. Indonesia has an official sign language acknowledged by the government, namely the sign system for Indonesian language (abbreviated as SIBI) listed under ministerial decree Number 0161/U/1994 (Ministry of education and culture of the republic of Indonesia, 1994). The sign language in SIBI consists of standardized movements of fingers and hands to represent a vocabulary. SIBI follows the grammar and structure of the Indonesian language to construct its sign gestures.

In order to communicate well, a deaf person and those involved in the interaction both need to be familiar with the sign language system being used. However, SIBI is only mandated in school for deaf students curricula, so not everyone is familiar with SIBI's gestures. Therefore a sign language translation system is needed to assist communication processes involving SIBI.

Multiple existing studies on this topic have supported the need for a sign language translation system, both internationally and in Indonesia. Takayama *et al.* [4] explained using the hidden Markov model (HMM) method for annotation tasks using Japanese sign language recognition. Rosalina *et al.* [5] employed the artificial neural network (ANN) method as an identifier for Indonesian sign movements but was limited to hand movements as the input. Rakun *et al.* [6] elaborated the recognition process of prefixes, root words, and suffixes in Indonesian inflectional words using HMM method. In another research, Rakun *et al.* [7] used the probabilistic graphical model (PGM) to recognize affix components and root words in Indonesian inflectional word gestures. Models tested in the research were conditional random fields (CRF), hidden Markov model (HMM), long short-term memory neural networks (LSTM), and gated recurrent unit (GRU). In that last paper, the best results are shown by the LSTM model, which turned in a better performance relative to other PGMs for recognizing affix components and root words in Indonesian inflectional word gestures. As mentioned earlier, those studies aimed at identifying gestures to make it easier for everyone to communicate with sign language systems.

There are two types of gestures in a full-fledged video containing sign language sentences: word and epenthesis. A sign movement (word-gesture) is a motion that has a defined meaning in the sign language system, which usually corresponds to a concept in the spoken language that the sign language system represents. In contrast, epenthesis movement (transitional-gesture) is a meaningless motion. Epenthesis movement links a word gesture with the next word gesture or marks the end of a sentence. Besides connecting word gestures in a sentence, epenthesis is also used to connect a root word with its affixes in an inflectional word gesture.

One of the uniqueness of SIBI is the presence of inflectional word gestures. Inflectional words gestures do not have unique gestures but are formed by connecting the component word gestures that make up the inflectional words. Of the various types of SIBI gestures, inflectional word gestures are the most common types of gestures. Inflectional words are root words added with prefixes, suffixes, particles, and with prefix + suffix pairs (confixes). The addition of these affixes serves to give additional meaning to the root word. Figure 1 shows one example of a breakdown of video frames of a sentence. From Figure 1, we can see the Epenthesis in SIBI sentence gestures is located:

- at the beginning of a sentence
- between two consecutive words
- between a prefix and a root word
- between a root word and a suffix
- between a suffix and the next word
- at the end of the sentence

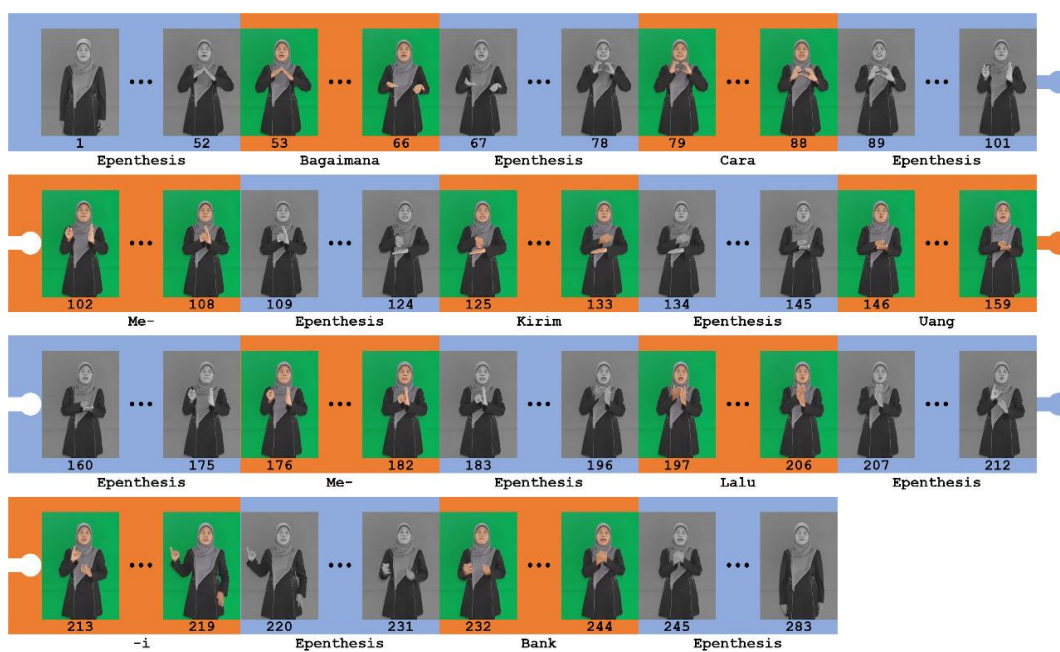


Figure 1. Pieces of SIBI sign movement “Bagaimana cara mengirim uang melalui bank?
(How can I send money through a bank?)”

There are so many possibilities for epenthesis that make them difficult to recognize by the translation system. Because the epenthesis is challenging to identify and it has no meaning, in this paper, we propose a translation system that excluding all epenthesis found in SIBI sentence gestures. The remainder of this paper is organized as shown in; Section 2 states the proposed method for SIBI gesture recognition system. Section 3 explains the proposed method for automated epenthesis removal, dataset, evaluation metric, and base case experiment results. Section 3 also explains the Post-TCRF and Pre-LSTM Improving techniques. Section 4 analyzes the experiment results. Finally, section 5 closes this paper with the conclusion and future works.

2. PROPOSED METHOD FOR SIBI SENTENCE RECOGNITION SYSTEM

Existing works on sign language recognition systems are summarized in Table 1. From the literature review, other research generally uses isolated words and has not been designed to be implemented as mobile apps. Our research proposes a mobile app for sign language recognition that uses the phone's built-in camera and operates in the continuous-dynamic domain.

One of the main tasks in producing sign language translation system is eliminating epenthesis (transitional-gesture), determining the starting and ending frame of a word-gesture and recognizing those word-gestures. The purpose of eliminating epenthesis frames is to make the word-gesture recognition easier, by reducing the amount of irrelevant inputs to the model. If they are not deleted, these epenthesis frames can reduce the accuracy of gesture recognition.

Table 1. Related works

Author	Specification	Dataset Static, Dynamic, Isolated, Continuous Gesture	Implementation Details	Remarks
Koller <i>et al.</i> [8]	Multiple signers. Uses hand shape + position + movement, inter-hand relation, detailed facial parameters and temporal derivatives. Vision-based.	SIGNUM database, RWTH-PHENIX-weather database Continuous Gesture	N/A (no implementation details)	Early phase, error rate is still high.
Breland <i>et al.</i> [9]	Thermal Imaging, Deep CNN.	Static.	Raspberry Pi 4, Nvidia AGX.	Not usable with a phone, since most phones do not have thermal cameras.
Ferreira <i>et al.</i> [10]	Signer-invariant representations (SI-PSL) + Jochen-Triesch, MKLM datasets.	Static.	Multiple encoder + decoder and a CNN and an MLP. Relatively heavy.	Not suitable for continuous gesture recognition.
Zhou <i>et al.</i> [11]	BERT-based. Hong Kong Sign Language, Chinese sign language, and RWTH-PHOENIX-Weather datasets.	Continuous.	Heavy.	Not usable for Indonesian yet; needs the existence of an Indonesian SignBERT model first.
Xu <i>et al.</i> [12]	Tensor train factorization for reduced parameter count. Chinese sign language.	Dynamic, not continuous (isolated words).	Heavy	Not usable for a video feed (continuous data).
Wei <i>et al.</i> [13]	Aligned video and text. Semantic boundary detection. Reinforcement learning. CSL Split II, RWTH-PHOENIX-Weather.	Spatial + temporal. Continuous (needs both text and video).	CNN-BiLSTM-CTC.	Semantic schema incompatible with Indonesian sign language.
Chaikaw <i>et al.</i> [14]	Thai language, no further details discussed.	Static.	Mobile app implementation was not discussed.	Very limited vocabulary.
Eqab and Shanableh [15]	Arabic. Sensor-based (Leap Motion on Android). Tiny dataset.	Isolated, dynamic.	Android.	Not usable because it's not using the phone's camera.
Mohamed <i>et al.</i> [16]	N/A (review paper)	N/A	N/A	Not many people have done a deep dive into the continuous hand gesture recognition domain.
Neiva and Zanchettin [17]	N/A (review paper)	N/A	N/A	Most mobile apps do not operate in the continuous dynamic domain.
Kudrinko <i>et al.</i> [18]	N/A (review paper)	N/A	N/A	Not many vision-based sign recognition systems are capable of using mobile phone cameras.

The activities reported here are part of the research that has been carried out since 2017. Broadly speaking, the application for translating SIBI gestures into text consists of 7 phases as shown in Figure 2 [19]. The research began with recording SIBI sentence gestures. The dataset used in this study consists of videos of various sentences in SIBI, which are performed by three teachers and two students from the Santi Rama Special School for hearing-impaired students, Jakarta. The gesture recording is done by using a mobile phone camera. The 2nd phase consists of dividing a video into a set of organized frames. The 3rd phase consists of developing the feature extraction process to determine identity of each frame.

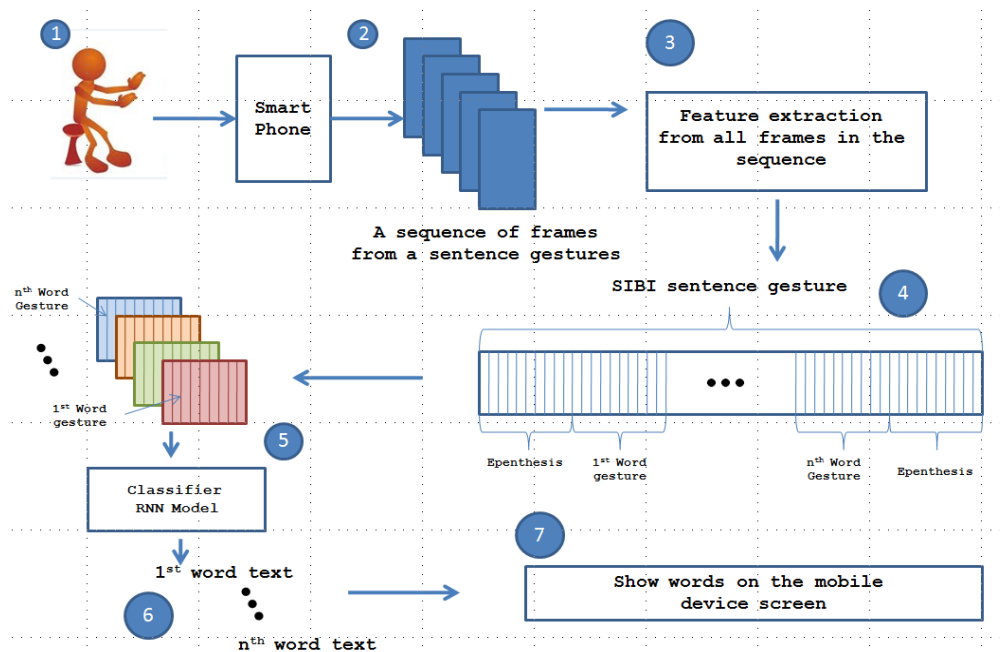


Figure 2. The Indonesian sign movement recognition system development research series [19]

Regarding the 3rd phase, the feature extraction techniques that are examined in this work are skeleton feature extraction [20], hand shape feature extraction [21], and MobileNetV2 [22]. The skeleton and hand shape feature did not utilize deep learning models and experiments were run on another dataset, the inflectional words dataset. The skeleton and hand shape feature extraction methods yielded a best accuracy of 86.6% and 99% respectively [20], [21]. The MobileNet feature extraction [22], involving deep learning models, was running on the sentence dataset, the same dataset used by this paper. This extraction technique can pinpoint the identity for each video frame (separating epenthesis-gestures, suffixes, prefixes and root words), enabling sentence recognition from SIBI gestures with a 99% accuracy rate.

The 4th phase of the research initiative involves the separation of gestures and epenthesis movements. In a previous study, Widhinugraha and Rakun [23] conducted a segmentation process using a TCRF model on SIBI inflectional word gestures only. The TCRF model with a Gaussian variance of 50, over 500 iterations and a threshold of 1.5 achieved an accuracy of 81.5% in identifying epenthesis gestures in an inflectional words-only dataset, using data recorded from a Kinect sensor. Referring to Figure 2, the fifth phase involves the process of recognizing gesture. Halim and Rakun [24] used a long short-term memory (LSTM) as a classifier. Halim's work explored variations in the LSTM's parameters, including hidden layer sizes of 128, 256, and 512, training data batch sizes of 50, 100, 200, and the number of training iterations from 100 to 1000 in increments of 10. The highest accuracy achieved (on the inflectional words dataset) was 96.150% for root words recognition using 512 hidden units and a batch size of 50 at the 400th epoch, and 78.380% for affix words recognition using 512 hidden units and a batch size of 200 at the 600th epoch through removal of epenthesis-gesture labeled movement inputs based on manual annotation (without using a TCRF). The 6th phase is a process of uniting the constituents of the gesture sentences, namely, the gesture representing each word. The final phase is the display of the translated sentence. Sentences that appear as output must be following Indonesian grammar. For example, the use of capital letters in the first letter of a name; the existence of fusion when connecting prefix with root word (prefix "me"+root word "sapu" becomes the inflectional word "menyapu"=to sweep).

The essence of the research in this paper lies in process number 4. The research aims to find features and inputs to TCRF, so that TCRF can precisely distinguish between gesture and epenthesis SIBI movements. To be able to measure TCRF performance, process 1-7 from Figure 2 needs to be carried out entirely. From our previous works [20]-[22], [24], it can be seen that the system we built can implement the overall design of the SIBI gesture-to-text translation system in Figure 2. The best design is achieved with the MobileNetV2 as feature extraction and the bidirectional LSTM as the classifier, resulting in a 99% accuracy rate. The drawback of the system we have built is implementing process number 4 in Figure 2, which separates the frames containing the word gesture from the epenthesis gesture. Because our approach is supervised learning, all of the datasets we have are already labeled by an annotator. The labels used are 0 for epenthesis gesture and non-0 label for word gesture. Our previous work implemented the separation between word and epenthesis gesture (process number 4 in Figure 2) based on the annotated label. After creating features for each frame, frames with epenthesis-gesture label 0 based on annotations from the annotator were discarded. Then, we equalized the number of frames of all the remaining word gestures before we made them as the input to the LSTM.

The system we have built will be used in a mobile application that does not allow manual separation between the frames for epenthesis-gesture and the word-gesture based on labels. For automated epenthesis removal, we propose to use threshold conditional random fields (TCRF) in process number 4 in Figure 2, which will be discussed in detail below.

3. METHOD

The discussion in this section will be divided into two parts. The first part discusses the proposed method for automated epenthesis removal. The second part discusses post-TCRF and pre-LSTM Improving Techniques. This second part aims to improve the performance of the methods proposed in the first part.

3.1. Proposed method for automated epenthesis removal

In this section, we will discuss how we use TCRF to distinguish between frames containing epenthesis-gestures or containing word-gestures. Seeing that previous works in this series have already focused on the coarse-grained optimization approaches on the data gathering and feature engineering stages, this time, we decided to look into optimizing the actual data ingestion part, as in the way the data is fed into the classification model. With the application of TCRF in process number 4 of Figure 2, the entire design in Figure 2 can then be implemented as a mobile application.

3.1.1. Threshold conditional random field

The system described in this work uses a Threshold Conditional Random Field (TCRF) and could be referred to [25]. *Threshold* models with CRF (TCRF) are established by adding labels for the G non-sign pattern in the original CRF, using weights of transition function and features of the original CRF. Therefore, TCRF includes label $S = \{Y_1, \dots, Y_l, G\}$, where $(1, \dots, l)$ signifies the number of CRF label and G symbolizes label for epenthesis-gesture pattern. The weight of state feature function of the G epenthesis-gesture label μ_m is calculated by (1):

$$\mu_m(G) = \bar{\mu}_m + T\sqrt{\sigma_{\mu_m}} \quad (1)$$

where $\bar{\mu}_m = \frac{\sum_{k=1}^l \mu_m(Y_k)}{l}$, k is a number of CRF label, σ_{μ_m} is a variant of weight to the m of state feature function and T (threshold) implies calculated weight to maximize the overall recognition rate through the use of training data. In Chung and Yang's research it was mentioned that the range of likelihood values that are considered not a core movement is between 1.0 to 4.0. In this work, the T (threshold) variable becomes an independent variable that can affect the model's accuracy.

3.1.2. Dataset

The data set examined in this work contains sentences that are used in daily conversation in Indonesian, represented as real SIBI sequences (containing word-gesture and epenthesis-gesture frames) as shown in Table 2. There are 2,275 videos in total, and each sentence is repeated several times in order to maintain the balance of the number of words in the data set. The sentences are essential phrases, encompassing greetings and introductions, public transportation usage, and phrases that would be used in navigating hospitals, markets, banks and cinemas.

Table 2. Sentences in the dataset

No	Sentence	Repetition
1	Siapa namamu? (What's your name?)	50
2	Di mana alamat rumahmu? (Where do you live?)	75
3	Di mana sekolahmu? (Where is your school located?)	75
4	Bolehkah saya minta nomor teleponmu? (May I have your telephone number?)	75
5	Film apa yang sedang diputar? (What movies are playing right now?)	75
6	Jam berapa film ini diputar? (At what times will this movie be shown?)	50
7	Berapa harga karcis film ini? (How much would a ticket for this movie cost?)	75
8	Di mana film ini diputar? (Where is this movie being shown?)	75
9	Apa nama sayuran itu? (What is this vegetable called?)	25
10	Berapa harga sayuran itu? (How much does this vegetable cost?)	50
11	Apakah harga sayuran ini boleh ditawar? (Is the price of this vegetable negotiable?)	75
12	Berapa jumlah yang harus saya bayar? (So how much do I have to pay for all of this?)	75
13	Kami ingin pergi ke kota tua, naik bis apa? (We would like to go to the Old Town, which bus do we have to take?)	100
14	Berapa harga karcis yang harus saya bayar? (How much would the tickets cost?)	75
15	Kami harus turun di mana? (At which station should we get off?)	75
16	Adakah cara lain kita pergi ke kota tua? (Is there another way we can get to the Old Town?)	100
17	Saya ingin membuka tabungan, bagaimana caranya? (How would I go about opening a savings account?)	75
18	Bagaimana cara menabung? (How would I go about saving money?)	50
19	Di mana kami bisa mengambil tabungan? (Where can we withdraw our savings?)	75
20	Bagaimana cara mengirim uang melalui bank? (How can I send money through a bank?)	125
21	Selamat natal dan tahun baru (Merry Christmas and Happy New Year)	125
22	Selamat idul fitri mohon maaf lahir dan batin (Happy Eid ul-Fitr, please forgive me for my mistakes)	125
23	Selamat ulang tahun (Happy Birthday)	125
24	Semoga panjang umur (May you live for as long as you wish)	125
25	Saya sering sakit kepala, saya harus periksa ke bagian mana? (I frequently get headaches, which medical specialty department should I visit?)	100
26	Saya ingin ke dokter umum, siapa nama dokternya? (I want to see a general practitioner, could you put me in touch with one that's available?)	75
27	Jam berapa dokter datang? (At what time is the doctor expected to arrive?)	25
28	Di apotek mana obat ini bisa dibeli? (At which pharmacy can I obtain this medicine?)	75

3.1.3. Evaluation metrics

TCRF model evaluation is done using accuracy as a metric. Accuracy is calculated relative to manually labeled data (each video frame has a epenthesis-gesture label or a word label associated to it), according to (2):

$$Accuracy = \frac{1}{N} \sum_{n=1}^N \frac{\text{Number of correctly identified frames}}{\text{Total number of frames}} \quad (2)$$

where N is the number of samples.

LSTM model evaluation is done using word error rate (WER) and sentence accuracy (SAcc) as a metric. The formula for calculating WER is:

$$WER = \frac{S+D+I}{N} = \frac{S+D+I}{S+D+C} \quad (3)$$

where:

- S is the number of substitutions (a word in the sentence was replaced)
- D is the number of deletions (words that are in the original sentence but does not appear in the predictions)
- I is the number of insertions (words that are not in the original sentence but appears in the prediction).
- C is the number of correct words,
- N is the number of words in the reference ($N = S + D + C$)

Another metric for the LSTM is Sentence Accuracy (SAcc). The original sentence is compared with the sentence resulting from LSTM prediction. If the original sentence does not match with the prediction one, then the sentence accuracy is 0%. The formula to calculate average SAcc is (4).

$$Sentence Accuracy = \frac{\text{Number of correct sentences}}{\text{Total number of sentences}} \quad (4)$$

3.1.4. Experiment phases

The overall system designed in this work contains two models for the two different subtasks, namely the separation of word-gesture and epenthesis-gesture frames (the gesture separation/sequence truncation

task) and the classification of the SIBI gesture frame sequences into Indonesian words (the word classifier task).

The first phase aims to find the best way to label and the best TCRF threshold for the gesture separation task. The second phase combines TCRF with the best setting obtained from the first phase with a bidirectional LSTM for the gesture classification task. The second phase produces the baseline experiment results. The improvement techniques discussed in section 3.2 will be compared to this baseline result.

a). First phase experiments

For the gesture separation task using MobileNetV2 for feature extraction, experiments were carried out on three label combinations, the 2-label (word or epenthesis gesture), 4-label, 84-label, and using threshold 1.5 for TCRF. With the 4-class label set, each frame could be one of four classes of gestures: prefix, suffix, root word, or epenthesis-gesture. For example, in the prefix group, there are seven prefixes ("be," "di," "ke," "me," "pe," "se," and "te"). TCRF does not need to distinguish one prefix from another because TCRF only needs to categorize each frame as non-epenthesis or epenthesis. With the 84-class labels set, each frame can be identified as any labels available in the dataset (83 labels for non-epenthesis and one for epenthesis). Experiment results show that the TCRF output accuracy for 2-, 4- and 84-label are 88.18%, 88.50%, and 89.9% respectively. Accuracy is measured by comparing the TCRF predicted result with the actual label (2). It can be seen that model trained with 84-label achieved the best results with an accuracy of 89.9%.

Figure 3 shows a frame-by-frame 2-class probability plot (epenthesis gesture or word gesture) generated by the best model of MobileNetV2 and TCRF to represent "Siapa namamu?" ("What is your name?"), which is the first sentence in this data set. Note that the word gestures are on frames 78-106, 113-134, and 141-162. All other frames represented in that diagram represent epenthesis gestures.

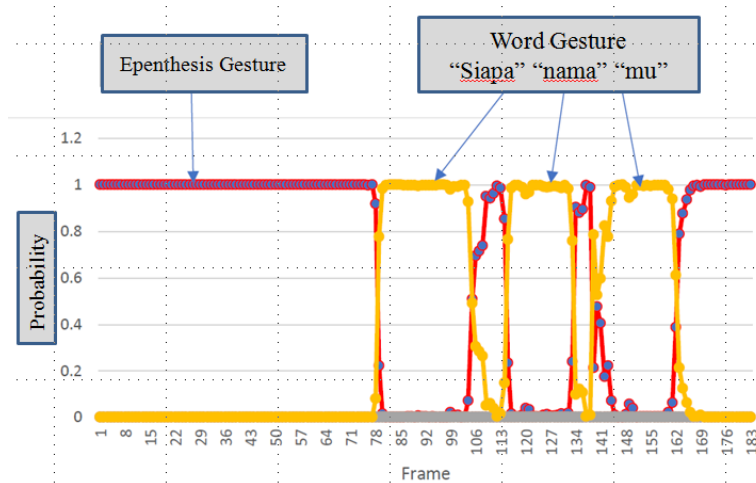


Figure 3. Probability chart 2-Label for "Siapa namamu? (What is your name?)"

b). Second phase experiments

The next experiment is to remove all frames predicted as epenthesis (transition) gesture. The remaining frames are frames that contain the gesture of the words in the sentence. Furthermore, the frames of each word will be used as the input to the 2-layer bidirectional LSTM model. The word error rate (WER) and Sentence Accuracy (SAcc) from the LSTM prediction results can be seen in Table 3. This setting is used in the second phase. In order to make a reliable SIBI gesture recognition system, of course, this result needs to be improved. The improvement technique will be discussed in the following section 3.2, and the results in Table 3 will be used as a baseline.

Table 3. T-CRF - Bidirectional LSTM output accuracy

Label	2-layer Bidirectional LSTM Output	
	WER	SAcc
2 - label	45.749%	7.658%
4 - label	48.003%	8.108%
84 - label	33.407%	16.216%

3.2. Post-TCRF and Pre-LSTM improving techniques

The purpose of this research phase is to improve the WER and SAcc of the LSTM by using additional processing techniques post-TCRF (Process 5 and 6 in Figure 4) and post-LSTM (Process 7). The workflow of this stage can be seen in Figure 4. The LSTM results seen in Table 3 are obtained by running processes number 1, 2, 3 and 4 from Figure 4. The improvements consisted of: removing 1 to 4 frames of transition gesture in between 2-word gestures (sandwiched majority voting, Process 5); Discarding gesture with very few frames (short word-frame sequence relabeling, Process 6); Combining LSTM output/ grouping the same LSTM prediction results (LSTM output grouping, Process 7).

To simplify the discussion and illustrate the flow of the experiment more clearly, the LSTM results in Table 3 are referred to as the baseline case by following processing path (1-2-3-4). Three variations will be done. The first variation will follow processing path (1-5-2-3-4), the second variation will use processing path (1-5-6-2-3-4), and the third follows the (1-5-6-2-3-4-7) processing path, all from Figure 4.

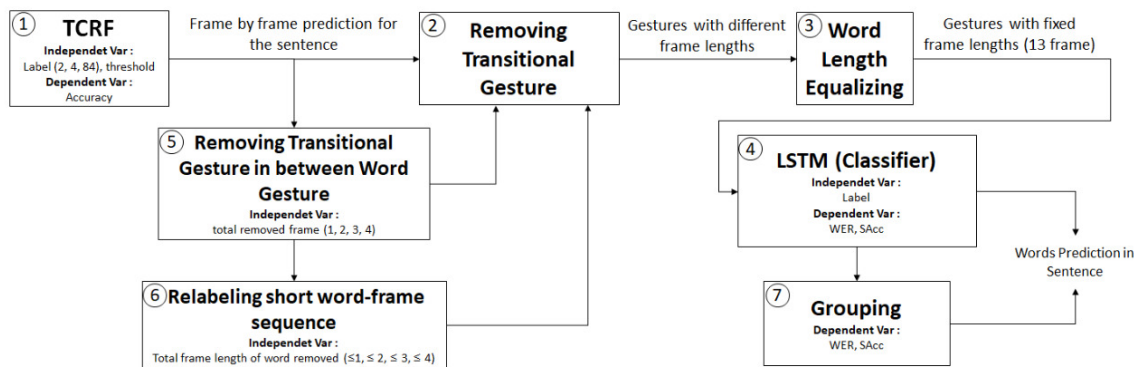


Figure 4. Post-TCRF and Pre-LSTM Improving techniques

3.2.1. Correcting Ambiguous sandwiched word gesture labels (gesture frames classified as transition frames within the gesture sequence of the same word)

When analyzing the output of the TCRF, we found a recurring pattern of prediction errors made by the TCRF. A small number of frames were predicted as transition gestures (labeled as 0) sandwiched between frames of a word gesture, whereby the frames before and after the sandwiched gesture frames are of the same word label. The prediction error can happen because the word gesture feature in that frame is very similar to the epenthesis gesture feature. Figure 5 shows an example of this finding. This prediction error must be corrected by replacing the transition gesture label (0 in this dataset) with the proper word gesture. Figure 5a shows the sequence of frames before the correction. In contrast, Figure 5b shows the sequence of frames after the correction. We refer to this label replacement technique as a sandwiched majority voting process, denoted as Process 5 on Figure 4.

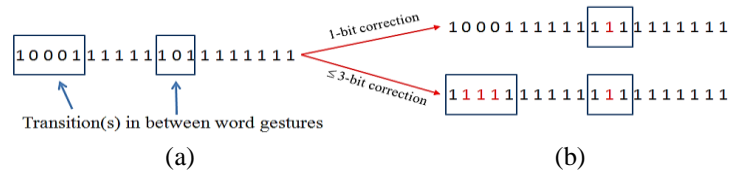


Figure 5. Correcting transition frames in between word gestures based on their surrounding frames, (a) sequence of frames before correction and (b) sequence of frames after correction

Before correcting the label 0, we need to set the threshold, namely how many "0" labeled frames have to happen in order for them to be categorized as a TCRF misclassification. We denote this threshold as the sandwiched majority voting sequence length threshold. If the sandwiched majority voting sequence length threshold is set at 1, the result will look like the above path of Figure 5(b). If the threshold is set to be less than or equal to 3 frames, the result will look like the bottom path of Figure 5(b). To obtain the optimal sandwiched majority voting sequence length threshold, we tested thresholds of 1, 2, 3, and 4 frames while

measuring the resulting accuracy of TCRF predictions as shown in (2). The experimental results of the sandwiched majority voting technique for the dataset using 2, 4, and 84 label sets can be seen in Table 4.

Table 4. TCRF Accuracy with sandwiched majority voting Process

Label Correction	TCRF 2 Label	TCRF 4 Label	TCRF 84 Label
	Average Accuracy (%)	Average Accuracy (%)	Average Accuracy (%)
Baseline	88.1763	88.5032	89.9037
1 frame	88.3816	88.7142	90.0379
2 frames	88.5729	88.9426	90.1490
3 frames	88.4872	88.9349	90.2028
4 frames	88.1939	88.8000	90.2571

To see the impact of the TCRF prediction label correction by sandwiched majority voting on the LSTM prediction results, an experiment was carried out by following the processing path (1-5-2-3-4) from Figure 4. The independent variable used in this experiment was the sandwiched majority voting sequence length threshold (1, 2, 3 or 4) and the number of labels used (2, 4 or 84 labels). The dependent variables are WER and SAcc from the LSTM prediction results as shown in (3) and (4). The experiments result can be seen at Table 5.

Table 5. Output of LSTM after implementing the majority voting process

Label Correction	LSTM (TCRF 2 Label)		LSTM (TCRF 4 Label)		LSTM (TCRF 84 Label)	
	WER (%)	SAcc (%)	WER (%)	SAcc (%)	WER (%)	SAcc (%)
Baseline	45.749	7.658	48.003	8.108	33.407	16.216
1 frame	30.539	15.766	30.969	19.595	21.962	31.757
2 frames	18.586	32.207	19.885	34.685	15.472	45.946
3 frames	14.484	40.315	14.793	43.694	12.905	51.126
4 frames	14.309	43.243	13.298	47.072	11.347	55.405

3.2.2. Short word-frame sequence relabeling

We examined the TCRF prediction results after completing the sandwiched majority voting process in detail, and there are some very short word-frame sequences consisting of only 1-4 frames. An example of this case can be seen in Figure 6. Generally, a word gesture sequence consists of tens of frames, so we decided to re-label the short word frame sequences as epenthesis sequences (label 0). This process is denoted as Process 6 in Figure 4. If this short word-frame re-labeling sequence length threshold is one frame, then the original sequence (Figure 6(a)) will be corrected as the sequence shown in Figure 6(b). Furthermore, If the threshold is 2, then the output of process 6 will look like Figure 6(c). The next experiment is to find the optimal combination of sandwiched majority voting sequence length threshold and short word-frame relabeling sequence length threshold that will produce the best WER and SAcc. This experiment follows the processing path (1-5-6-2-3-4) of Figure 4. The experimental results can be seen in the Table 6. The best results are achieved when TCRF uses 84 labels, sandwiched majority voting sequence length threshold is set at 4, and the short word-frame relabeling sequence length threshold is set at 2 frames. The WER was 5.165% and SAcc was 70.946% in this configuration.

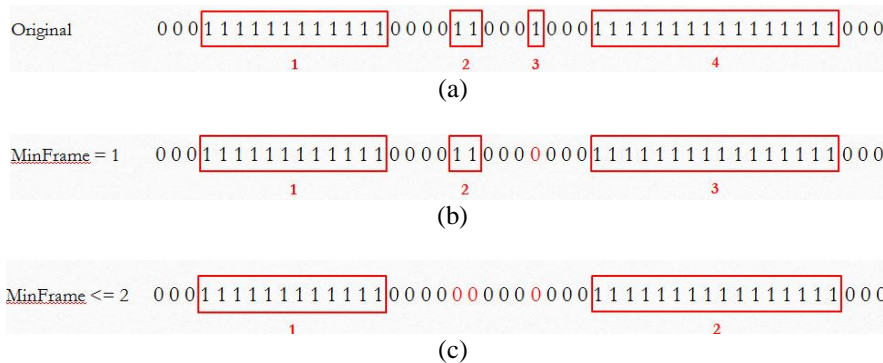


Figure 6. Removing word gesture with a small number of frames (a) original sequence of frames, (b) sequence of frames after removing label with one frame length, and (c) sequence of frames after removing label with less than equal two frames

Table 6. Experiment results for path 1-5-6-2-3-4

Label Correction	LSTM (TCRF 84 Label)							
	Frame_Min <= 1		Frame_Min <= 2		Frame_Min <= 3		Frame_Min <= 4	
	WER	SAcc	WER	SAcc	WER	SAcc	WER	SAcc
Baseline	19.265%	28.604%	12.675%	43.018%	10.839%	47.748%	10.729%	48.198%
1 frame	12.121%	47.748%	8.743%	57.207%	8.082%	59.009%	8.932%	55.405%
2 frames	8.259%	61.036%	6.403%	65.991%	6.580%	65.090%	7.754%	60.586%
3 frames	7.192%	65.090%	5.723%	68.919%	6.102%	66.441%	7.326%	61.486%
4 frames	6.313%	67.793%	5.165%	70.946%	5.622%	68.243%	7.009%	62.613%

3.2.3. LSTM output label grouping

When analyzing the LSTM prediction results, it is often found a label that appears repeatedly, as in the example shown in Figure 7. In Indonesian, the word reduplication is used to express plurality (for example, anak=child, anak-anak=children); increase in intensity (for example mahal=expensive, mahal-mahal=very expensive). In the case of translating gestures into text, word repetition often occurs not because of the reduplication of the Indonesian word, but because the sequence of frames of a word gesture are divided into more than one sequence by TCRF, and each chunk is predicted to be the same word. To solve the repeated occurrence of a word, we try to group the same labels that appear sequentially. The grouping process is carried out by process number 7 in Figure 4. An example of grouping process can be seen in Figure 7. The experiment carried out the grouping process following the processing path (1-5-6-2-3-4-7) from Figure 4. The results of the experiment can be seen in Table 7. The best results are achieved when TCRF uses 84 labels, sandwiched majority voting sequence length threshold is set at 4, and the short word-frame relabeling sequence length threshold is set at 1 frame. The best WER is 3.367% and SAcc 80.180%.



Figure 7. An example of LSTM Output LSTM output grouping process

Table 7. Experiment Results for path 1-5-6-2-3-4-7

Label Correction	LSTM (TCRF 84 Label)							
	Frame_Min <= 1		Frame_Min <= 2		Frame_Min <= 3		Frame_Min <= 4	
	WER	SAcc	WER	SAcc	WER	SAcc	WER	SAcc
Baseline	4.327%	75.676%	4.580%	73.198%	5.970%	67.117%	8.063%	60.135%
1 frame	4.001%	77.027%	4.196%	75.225%	5.540%	69.144%	7.481%	61.712%
2 frames	3.688%	78.829%	3.941%	76.802%	5.167%	70.946%	6.951%	63.964%
3 frames	3.421%	79.730%	3.770%	77.477%	4.915%	71.622%	6.637%	64.640%
4 frames	3.367%	80.180%	3.700%	77.928%	4.808%	72.297%	6.529%	65.315%

4. RESULT AND DISCUSSION

This section will discuss the impact of the extra processing steps (tested in Section 3.2) upon WER and SAcc by conducting a significance test. Table 8, which summarizes the best WER and SAcc results for each case, depicts the decline in the WER value and incline in the SAcc after extra processing steps were applied. Path 1-2-3-4 is the baseline case. The observations in Table 8 were carried out for 3 cases: 2-, 4- and 84-label. The use of 84-labels gives the best results for any extra processing steps, while 2 and 4 labels give almost the same results. When using 2- or 4-labels, many different word gestures have the same label. On the other hand, when using 84-label, each word gesture has a unique label.

A two-way ANOVA was conducted to compare the main effects of the extra processing steps and number of labels being used effect on word error rate (WER). The extra processing steps (path 1-2-5-3-4, 1-2-5-6-3-4 and 1-2-5-6-3-4-7) effects was statistically significant on WER. It was explained by the F-value = 82.52 (> Fcritical 4.47), and P-value = 2.87E-05 (<0.05). The number of label (2, 4 or 84) effects was statistically not significant on WER as the F-value = 3.08 (< Fcritical 5.14) and the P-value = 0.11 (>0.05).

A two-way ANOVA was also conducted to compare the main effects of extra processing steps and number of labels effect on Sentence Accuracy (SAcc). Both the extra processing steps (path 1-2-5-3-4, 1-2-5-6-3-4 and 1-2-5-6-3-4-7) and number of labels (2, 4 or 84) were statistically significant at p < 0.05. The main effect of extra processing steps was statistically significant on SAcc. It was explained by the F-value =

1055.89 ($>F_{critical} 4.47$), and P-value = $1.48E-08$ (< 0.05). The number of label effects was also statistically significant on SAcc as the F-value = 45.87 ($>F_{critical} 5.14$) and the P-value = 0.0002 (< 0.05).

Table 8. The impact of the extra processing steps performed on WER

Path	WER				SAcc			
	1-2-3-4 (baseline)	1-5-2-3-4	1-5-6-2-3-4	1-5-6-2-3-4-7	1-2-3-4 (baseline)	1-5-2-3-4	1-5-6-2-3-4	1-5-6-2-3-4-7
2-label	45.749%	14.309%	8.448%	4.903%	7.658%	43.243%	60.811%	74.324%
4-label	48.003%	13.298%	7.451%	4.980%	8.108%	47.072%	61.486%	72.072%
84-label	33.407%	11.347%	5.165%	3.367%	16.216%	55.405%	70.946%	80.180%

5. CONCLUSION

In order to build a reliable sign system for the Indonesian language (SIBI) sentence recognition system, the research aims to find the best way to recognize each word in SIBI sentence gestures and remove epenthesis gestures automatically. The purpose of eliminating epenthesis frames is to simplify word-gesture recognition by reducing the number of irrelevant inputs to the model. This research also tested three methods of labeling the dataset. The first method was to distinguish the gestures in the dataset into two groups, namely word gestures and epenthesis (transition) gestures. The second was to divide the gestures into four groups: root word, prefix, suffix, and epenthesis. The third way was to use 84 labels from the number of words in the dataset (83 words) and epenthesis gestures. The best result was achieved with the MobileNetV2-extracted feature set being fed into a TCRF, using the 84-labels dataset (third labeling method). This configuration recognized word-gesture and epenthesis-gestures with an accuracy of 89.9037%. Using the MobileNetV2-TCRF-BiLSTM model, the best word error rate (WER) and sentence accuracy (SAcc) were 33.407% and 16.216%, respectively. Improving the WER and SAcc is done through extra processing of the TCRF outputs. TCRF post-processing consists of using sandwiched majority voting (Section 4.1) and short word-frame relabeling (Section 4.2), and LSTM output grouping (Section 4.3). The sandwiched majority voting technique replaced the epenthesis gesture that lies in between a word gesture with the corresponding word gesture. The application of this technique reduced WER from 33.407% to 11.347% and increased SAcc from 16.216% to 55.405% if sandwiched majority voting sequence length threshold is equal to 4. The next improvement technique is to discard the predicted word gestures that are less than the minimum number of frames allowed for a word gesture. The experimental results show that the minimum number of frames for a word gesture is two frames (usually gestures representing the alphabet and numbers). By removing the word prediction with the number of frames ≤ 1 , WER can be reduced to 6.313%, and SAcc can be increased to 67.793%. The last improvement is the grouping of prediction results that appeared more than once in a row. WER could be lowered again to 3.367% and increase SAcc to 80.18%. A two-way ANOVA was conducted to compare the effects of the sandwiched majority voting, short word-frame sequence relabeling, and output grouping on both WER and SAcc. The test shows that they can improve WER and SAcc, and are statistically significant at $p < 0.05$. The impact of using 2, 4, or 84 labels did not give significantly different results on WER, but was statistically significant on SAcc as the F-value = 45.87 ($>F_{critical} 5.14$) and the P-value = 0.0002 (< 0.05). Although in general the use of 84-labels gives the best results. We can conclude that MobileNetV2, T-CRF, and LSTM can recognize word gestures in the dataset better when using 84 labels.

ACKNOWLEDGEMENTS

This work is supported by the computing facilities at the Tokopedia-UI AI Center of Excellence. This work is supported by Universitas Indonesia's Research Grant PUTI Q3 NKB-1832/UN2.RST/HKP.05.00/2020. This support is gratefully received and acknowledged. The authors also wish to thank Lim Yohanes Stefanus PhD and M. I. Mas M.Kom for the final proofreading.

REFERENCES




- [1] Badan Pusat Statistik, "Profile of Indonesian Population from SUPAS 2015 (In Indonesia: Profil Penduduk Indonesia hasil SUPAS 2015)," 2015. Accessed: Apr. 11, 2019. Accessed: 11 Apr 2019. [Online]. Available: <https://www.bps.go.id/publication/2016/11/30/63daa471092bb2cb7c1fada6/profil-penduduk-indonesia-hasil-supas-2015.html>
- [2] I. Wardani, D. Tarsidi, T. Hernawati, Astati, and Z. Alimin, "Introduction to education for children with special needs (In Indonesia: Pengantar pendidikan anak berkebutuhan khusus)," *Journal of Chemical Information and Modeling*, vol. 53, no. 9, pp. 1689–1699, 2013.
- [3] U. Von Agris, J. Zieren, U. Canzler, B. Bauer, and K. F. Kraiss, "Recent developments in visual sign language recognition," *Universal Access in the Information Society*, vol. 6, no. 4, pp. 323–362, Feb. 2008, doi: 10.1007/s10209-007-0104-x.

- [4] N. Takayama and H. Takahashi, "Sign words annotation assistance using Japanese sign language words recognition," in *Proceedings - 2018 International Conference on Cyberworlds, CW 2018*, Oct. 2018, pp. 221–228, doi: 10.1109/CW.2018.00048.
- [5] Rosalina, L. Yusnita, N. Hadisukmana, R. B. Wahyu, R. Roestam, and Y. Wahyu, "Implementation of real-time static hand gesture recognition using artificial neural network," in *Proceedings of the 2017 4th International Conference on Computer Applications and Information Processing Technology, CAIPT 2017*, Aug. 2018, vol. 2018-January, pp. 1–6, doi: 10.1109/CAIPT.2017.8320692.
- [6] E. Rakun, M. I. Fanany, I. W. W. Wisesa, and A. Tjandra, "A heuristic hidden Markov model to recognize inflectional words in sign system for Indonesian language known as SIBI (sistem isyarat Bahasa Indonesia)," in *Proceedings of the 2015 International Conference on Technology, Informatics, Management, Engineering and Environment, TIME-E 2015*, Sep. 2016, pp. 53–58, doi: 10.1109/TIME-E.2015.7389747.
- [7] E. Rakun, A. M. Arymurthy, L. Y. Stefanus, A. F. Wicaksono, and I. W. W. Wisesa, "Recognition of sign language system for Indonesian language using long short-term memory neural networks," *Advanced Science Letters*, vol. 24, no. 2, pp. 999–1004, Feb. 2018, doi: 10.1166/asl.2018.10675.
- [8] O. Koller, J. Forster, and H. Ney, "Continuous sign language recognition: Towards large vocabulary statistical recognition systems handling multiple signers," *Computer Vision and Image Understanding*, vol. 141, pp. 108–125, Dec. 2015, doi: 10.1016/j.cviu.2015.09.013.
- [9] D. S. Breland, A. Dayal, A. Jha, P. K. Yalavarthy, O. J. Pandey, and L. R. Cenkeramaddi, "Robust hand gestures recognition using a deep CNN and thermal images," *IEEE Sensors Journal*, vol. 21, no. 23, pp. 26602–26614, Dec. 2021, doi: 10.1109/JSEN.2021.3119977.
- [10] P. M. Ferreira, D. Pernes, A. Rebelo, and J. S. Cardoso, "DeSIRE: Deep signer-invariant representations for sign language recognition," *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, vol. 51, no. 9, pp. 5830–5845, Sep. 2021, doi: 10.1109/TSMC.2019.2957347.
- [11] Z. Zhou, V. W. L. Tam, and E. Y. Lam, "SignBERT: A BERT-based deep learning framework for continuous sign language recognition," *IEEE Access*, vol. 9, pp. 161669–161682, 2021, doi: 10.1109/ACCESS.2021.3132668.
- [12] B. Xu, S. Huang, and Z. Ye, "Application of tensor train decomposition in S2VT model for sign language recognition," *IEEE Access*, vol. 9, pp. 35646–35653, 2021, doi: 10.1109/ACCESS.2021.3059660.
- [13] C. Wei, J. Zhao, W. Zhou, and H. Li, "Semantic boundary detection with reinforcement learning for continuous sign language recognition," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 31, no. 3, pp. 1138–1149, Mar. 2021, doi: 10.1109/TCSVT.2020.2999384.
- [14] A. Chaikaew, K. Somkuan, and P. Sarapee, "Mobile application for Thai sign language," in *2018 22nd International Computer Science and Engineering Conference, ICSEC 2018*, Nov. 2018, pp. 1–4, doi: 10.1109/ICSEC.2018.8712709.
- [15] A. Eqab and T. Shanableh, "Android mobile app for real-time bilateral Arabic sign language translation using leap motion controller," in *2017 International Conference on Electrical and Computing Technologies and Applications, ICECTA 2017*, Nov. 2017, vol. 2018-January, pp. 1–5, doi: 10.1109/ICECTA.2017.8251936.
- [16] N. Mohamed, M. B. Mustafa, and N. Jomhari, "A review of the hand gesture recognition system: Current progress and future directions," *IEEE Access*, vol. 9, pp. 157422–157436, 2021, doi: 10.1109/ACCESS.2021.3129650.
- [17] D. Hirafuji Neiva and C. Zanchettin, "Gesture recognition: A review focusing on sign language in a mobile context," *Expert Systems with Applications*, vol. 103, pp. 159–183, Aug. 2018, doi: 10.1016/j.eswa.2018.01.051.
- [18] K. Kudrinko, E. Flavin, X. Zhu, and Q. Li, "Wearable sensor-based sign language recognition: A comprehensive review," *IEEE Reviews in Biomedical Engineering*, vol. 14, pp. 82–97, 2021, doi: 10.1109/RBME.2020.3019769.
- [19] A. A. Pratama, E. Rakun, and D. Hardianto, "Human skeleton feature extraction from 2-dimensional video of Indonesian language sign system (SIBI [sistem isyarat Bahasa Indonesia]) gestures," in *ACM International Conference Proceeding Series*, 2019, pp. 100–105, doi: 10.1145/3330482.3330484.
- [20] M. Elpeltagy, M. Abdelwahab, M. E. Hussein, A. Shoukry, A. Shoala, and M. Galal, "Multi-modality based Arabic sign language recognition," *Inst. Eng. Technol. Comput. Vis.*, vol. 12, 2018, doi: 10.1049/iet-cvi.2017.0598.
- [21] E. Escobedo and G. Camara, "A new approach for dynamic gesture recognition using skeleton trajectory representation and histograms of cumulative magnitudes," *29th SIBGRAPI Conf. Graph. Patterns Images*, pp. 209–216, 2016, doi: 10.1109/SIBGRAPI.2016.037.
- [22] M. A. Bencherif, M. Algabri, M. A. Mekhtiche, M. Faisal, and M. Alsulaiman, "Arabic sign language recognition system using 2d hands and body skeleton data," *IEEE Access*, vol. 9, pp. 59612–59627, 2021, doi: 10.1109/ACCESS.2021.3069714.
- [23] M. H. Nur Fauzan, E. Rakun, and D. Hardianto, "Feature extraction from smartphone images by using elliptical fourier descriptor, centroid and area for recognizing Indonesian sign language SIBI (Sistem isyarat Bahasa Indonesia)," in *Proceedings - 2019 2nd International Conference on Intelligent Autonomous Systems, ICoIAS 2019*, Feb. 2019, pp. 8–14, doi: 10.1109/ICoIAS.2019.00008.
- [24] S. Pramada and A. Vaidya, "Intelligent sign language recognition using image processing," *IOSR J. Eng.*, vol. 3, pp. 45–51, 2013, doi: 10.9790/3021-03224551.
- [25] I. A. Adeyanju, O. O. Bello, and M. A. Adegbeye, "Machine learning methods for sign language recognition: A critical review and analysis," *Intell. Syst. with Appl.*, vol. 12, p. 200056, 2021, doi: 10.1016/j.iswa.2021.200056.
- [26] Q. Xiao, M. Qin, P. Guo, and Y. Zhao, "Multimodal fusion based on LSTM and a couple conditional hidden markov model for Chinese sign language recognition," *IEEE Access*, vol. 7, pp. 112258–112268, 2019, doi: 10.1109/ACCESS.2019.2925654.
- [27] N. C. Camgoz, O. Koller, S. Hadfield, and R. Bowden, "Multi-channel Transformers for Multi-articulatory Sign Language Translation," in *16th European Conference on Computer Vision*, 2020, doi: 10.1007/978-3-030-66823-5_18.
- [28] N. F. P. Setyono and E. Rakun, "Recognizing word gesture in sign system for Indonesian language (SIBI) Sentences using DeepCNN and BiLSTM," in *2019 International Conference on Advanced Computer Science and Information Systems, ICACSIS 2019*, Oct. 2019, pp. 199–204, doi: 10.1109/ICACSIS47736.2019.8979772.
- [29] M. Al-Qurishi, T. Khalid, and R. Souissi, "Deep learning for sign language recognition: current techniques, benchmarks, and open issues," *IEEE Access*, vol. 9, pp. 126917–126951, 2021, doi: 10.1109/ACCESS.2021.3110912.
- [30] N. Adaloglou, T. Chatzis, I. Papastratis, and A. Stergioulas, "A comprehensive study on deep learning-based methods for sign language recognition," *IEEE Trans. Multimed.*, vol. 1, no. 1, 2021, doi: 10.1109/TMM.2021.3070438.
- [31] J. Huang, W. Zhou, H. Li, and W. Li, "Sign language recognition using 3D convolutional neural networks," *IEEE Int. Conf. Multimed. Expo*, pp. 1–6, 2015, doi: 10.1109/ICME.2015.7177428.
- [32] M. Parelli, K. Papadimitriou, G. Potamianos, G. Pavlakos, and P. Maragos, "Exploiting 3D hand pose estimation in deep learning-based sign language recognition from rgb videos," in *Computer Vision – ECCV 2020 Workshops*, 2020, doi: 10.1007/978-3-030-66096-3_18.




- [33] I. G. B. H. Widhinugraha and E. Rakun, "Indonesian language sign system (SIBI) recognition using threshold conditional random fields," in *ACM International Conference Proceeding Series*, Oct. 2019, pp. 380–384, doi: 10.1145/3373509.3373591.
- [34] K. Halim and E. Rakun, "Sign language system for Bahasa Indonesia (Known as SIBI) recognizer using TensorFlow and long short-term memory," in *2018 International Conference on Advanced Computer Science and Information Systems, ICACSIS 2018*, Oct. 2019, pp. 403–407, doi: 10.1109/ICACSIS.2018.8618134.
- [35] H. Chung and H.-D. Yang, "Conditional random field-based gesture recognition with depth information," *Optical Engineering*, vol. 52, no. 1, p. 017201, Jan. 2013, doi: 10.1117/1.oe.52.1.017201.

BIOGRAPHIES OF AUTHORS






Erdefi Rakun    received her bachelor degree in Electrical Engineering from the University of Indonesia, in Jakarta, Indonesia in 1982. She received her M. Sc. in Computer Science from University of Minnesota, USA, 1988. She received her Ph.D. in Computer Science from University of Indonesia, in 2017. From 1986 until now, she is a full-time lecturer in the faculty of Computer Science in University of Indonesia, holding an Academic rank of Associate Professor. Her research interests include, machine learning, deep learning, image processing for indonesian sign language recognition systems. She can be contacted at email: efi@cs.ui.ac.id.



I Gusti Bagus Hadi Widhinugraha    received his B.Sc. degree in Computer Science from the University of Udayana and M.Sc. degree in Computer Science from the University of Indonesia in 2017 and 2020, respectively. From 2019 to 2020, he was a research assistant at the University of Indonesia in the faculty of Computer Science, and has worked on the Indonesian Sign Language Translation Project. His research interest include, deep learning, data analysis, computer vision and image processing. He can be contacted at email: i.gusti711@ui.ac.id.



Noer Fitria Putra Setyono    received his B.Sc. in Computer Science from IPB University, Indonesia, in 2013 and M.Sc. in Computer Science from the University of Indonesia in 2020. From 2013 to 2016, he worked as an IT Consultant in sev-eral companies in Jakarta. In 2016, he also worked in the Sharia Division of May-bank Indonesia as a Sharia Business Support Analyst. Currently, he is working as a Research Assistant at Computer Science Faculty in University of Indonesia. His research interests include computer vision, artificial intelligence, and deep learning. He can be contacted at email: noer.fitria@ui.ac.id.