International Journal of Computing and Digital Systems
ISSN (2210-142X)
Int. J. Com. Dig. Sys. 14, No.1 (Sep-2023)

http://dx.doi.org/10.12785/ijcds/140176

Applying Artificial Neural Network Based on Backpropagation Method for Indonesian Sign Language Recognition

Irvanizam Irvanizam[1,*], Ivan Horatius[1] and Hizir Sofyan[2]

[1]*Department of Informatics, Universitas Syiah Kuala, Banda Aceh, Indonesia*
[2]*Department of Statistics, Universitas Syiah Kuala, Banda Aceh, Indonesia*

[*]*Corresponding author: irvanizam.zamanhuri@usk.ac.id*

Received 28 Mar. 2023, Revised 29 Jul. 2023, Accepted 10 Aug. 2023, Published 01 Sep. 2023

Abstract: Sign-language is a nonverbal language utilized by hearing loss and speech disorder people to interact and speak with each other. Sometimes people call it a visual-language utilizing hand gestures and changes in hand shape to express intentions and thoughts. In Indonesia, people use two sign language types commonly. They are Indonesian Signal System (ISS) and Indonesian Sign Language (ISL). In this paper, we investigated research on the recognition of ISL alphabets by proposing an application using the Artificial Neural Network algorithm with the principle of the Backpropagation method for dealing with this research. Firstly, we performed preprocessing the image data sequentially by splitting the dataset, labeling the dataset based on the alphabet types, scaling the images into a size of 150x150 pixels, and segmentizing the datasets to eliminate noise in the image to yield color, grayscale, binary segmentations, and image edge detection. Later on, we split the dataset and distributed them into training and testing sets with two different proportions, 70:30 and 80:20 percent. After applying the algorithm, we obtained the highest accuracy at 80:20 data proportions using an epoch of 700, a learning rate of 0.001, a batch size of 25, and some hidden nodes of 100 was 98.41 percent.

Keywords: Artificial Neural Network, Backpropagation, Indonesia Sign Language, Image Recognition

1. INTRODUCTION

According to data from the Management Information System for Disabilities Persons (MISDP) from the Ministry of Social Affairs on October 8, 2019, among people with disabilities in Indonesia, as many as 7,03% of the Indonesian population are people with hearing disorder disabilities, and 2,57% are people with speech disorder disabilities [1]. In their daily lives, they employ sign language to speak and know each other. Instead of sound, the shape and motion of the fingers and hands, lips, physique, and facial gestures are all used in this non-verbal language to speak the speaker's intentions and thoughts.

In the world, Sign language has various types. Although with the same written language, almost every country has a different sign language type. On the contrary, a few in every country have the same sign language, but the written language is diverse. Indonesian Sign Language (ISL) is a language promoted by the Indonesian Deaf Welfare Organization (IDWO) and developed by their deaf communities. Therefore, the ISL has become a practical and effective communication system for the people in Indonesia because it was born by deaf people themselves.

The most pleasant and natural form of communication for the disability and deaf-mute applied daily by people with hearing and speech disorders is sign-language. It is also a teaching and communication instrument in special education classrooms. It is a natural language form that decodes facial expressions, hand gestures, and shapes to determine meanings. Likely other natural languages, this language features a complete vocabulary system and regulated grammar. Therefore, many people in different nations are proficient in sign language and eager to extend the language to be a common natural language. However, a theoretical study of sign language translation remains in its early stages [2].

One of the contributions that can overcome such conditions is to develop a technology that can understand sign language. In many computer vision applications, including pattern classification or clustering techniques, computer vision in biomedicine, and information extraction based on learning processes, the Artificial Neural Network (ANN) is the most often used image classification technique [3].

ANNs are parallel distributed engines with a built-in propensity to store first-hand experience. In general, nonlinear correlations, high noise levels, complex, imprecise, error-prone sensor data, and the absence of explicitly stated mathematical answers or procedures are challenges that these techniques can effectively solve. An ANN's core benefit is the ability to construct a system model from the data at hand. It utilizes the backpropagation technique to extract texture characteristics in classifying images [4]. Meanwhile, the backpropagation approach is one of the supervised ANN methods examining the error for all neurons after processing a dataset. It aims to change the weights by training the NN efficiently and mapping an input correctly to an output [5]. To perform a sufficient experiment, in this study, we utilize ANN with a multilayer network architecture to identify the image dataset, and the output of this study is the result of the accuracy of ISL image recognition.

The research gap of this study is whether the backpropagation ANN technique has the ability to accurately recognize ISL alphabet images or not. While the contributions of this study are to implement an ANN application system-based backpropagation approach to interpret ISL alphabets images into alphabet letters and analyze the accuracy of this model in predicting those alphabets images.

The remaining paper neatly unfolds as the following. Section 2 re-discusses some related literature utilized in this research. Section 3 explains the study methodology. Section 4 demonstrates the results to confirm the conformity of the ANN based on backpropagation in recognizing alphabet sign images. Finally, Section 5 lists brief conclusions and suggests future potential works.

2. RELATED WORKS

Recently, the study of developing and applying ANN algorithms has mainly emphasized prediction problems. For instance, Zhang *et al.* [6] utilized ANN with a backpropagation approach to forecast the post-blast-re-entry time. Zuraidah *et al.* [7] applied ANN to forecasting nonlinear electrical loads for short-term at a city in Indonesia. Li *et al.* [8] visualized a simulation of tidal flow systems with synthetic wastewater using the ANN model to predict nutrient removal in this system. Al-Bakri and Sazid [9] applied ANN to predict blast events to obtain an optimized blasting process. Maulana *et al.* [10] utilized ANN to predict Kovats retention indices for some chemical compounds. Idroes *et al.* [11] integrated ANN and genetic algorithms to forecast Kovats retention indices.

In addition, the researchers of ANN have also developed studies related to recognizing objects using some approaches. For instance, Mas'ud *et al.* [12] investigated how to implement an application for detecting partial discharge patterns using ANN. Orazaev *et al.* [13] also demonstrated how to recognize images affected by random-valued impulse noise using ANN. Dieste-Velasco [14] developed an application to detect analog electronic circuit faults using the ANN model. Andrian *et al.* [15] attempted to investigate the batik Lampung motive using ANN Backpropagation. They compared the training and testing datasets containing 70:30 percent and 80:20 percent. The results showed that ANN backpropagation distinguished batik motives between Lampung motives and from outside Lampung with an accuracy rate of 92 percent and the error rate of 8 percent. Hossain *et al.* [16] utilized ANN and developed an intelligent model-based backpropagation to recognize predefined Latin characters subsequently. Bravo-López *et al.* [17] explored an ANN model for landslide susceptibility evaluation at a high land in Ecuador. They implemented a technique consisting of an ANN and multilayer perceptron and generated them with one of the R packages. Quraishi *et al.* [18] provided a new model based on ANN for image recognition. They used a grey-scale image and then added Salt and Peeper noise into the image dataset. Moreover, they implemented an adaptive median filter to eliminate the noise of the image datasets. Here, they regarded the output as filtered.

In the last five years ago, there were some comprehensive studies focused on sign language (SL). For instance, Calado *et al.* [19] used a variety classifiers in machine learning techniques, likely random forest, ANN, SVM, and logistic regression, to detect Italian sign language. ANN displayed the best performance in this investigation, correctly recognizing images of Italian sign language 95.07% of the time.

Yu *et al.* [20] evaluated the use of electrodes on four hands brawns and six instrument measurements on the back of arms to apply a deep belief net to SL. The data was gathered using eight signers who each performed five times each of 150 Chinese SL words. The scenarios with signer dependence and independence obtained accuracy rates of 95.1% and 88.2%, respectively. Gao *et al.* [21] used two Cyber-gloves to record wrist and hand data and three Polhelmus-3SPACE trackers (place) to collect data on the direction and location of the wrist and back of the signer. There were 5113 Chinese SL gestures performed twice by six signers. The outputs exhibited that the average accuracy for classifying the data was 82.9% using a Hidden Markov model.

Wu *et al.* [22] utilized wires on the forearm and nine instrument measurements on the wrist to create a system of American SL. On eighty word symbols, all recounted 77 times by five symbols throughout four sessions, offline classification was performed using SVM, Naive-Bayes, decision trees, and closest k-NN classifiers. While for classifying signer-dependent and signer-independent, an SVM performed with an accuracy of 97.89% and 40%, respectively.

Moreover, many researchers have widely explored ISL for conducting their scientific observations. Daniels *et al.* [23] utilized an object detector technique, You Only Look Once (YOLO), to design an ISL system for recognizing real-time video datasets. The technique predicted ISL alphabets with approximately 93.1% of accuracy. Indra *et* *al.* [24] proposed a system that can read hand-shape ISL letters based on distance calculation between input ISL letters and output letters in the database. The proposed system recognized ISL alphabets properly with about 95% of accuracy. Fadlilah *et al.* [25] developed an Android-based application called BisAndro to facilitate disabled people to communicate with each other. They used Convolutional Neural Network (CNN) approach to perform image recognition.

However, we realize that this study is necessary to investigate using other machine learning techniques for obtaining the diversity of results from other intelligent models. Therefore, this study concentrates on the structured development of the ANN model based on the backpropagation technique for recognizing sign language to help disabled people.

A. *Artificial Neural Network (AI)*

In AI studies, the ANN is an intelligent technique invented using the nervous system of human beings and adopted its principle for yielding decisions. It contains millions of neurons that carry out information processing. As a result, it makes the ANN has some benefits [26]. Adaptive learning is the first benefit. Based on training data, it can learn how to perform a task. The autonomous organization is the second benefit. An ANN can organize the data it gets during learning time and build a representation of it. Thirdly, it operates in real time. It can therefore carry out the calculations in parallel.

In general, there are two forms of ANN architecture: single-layer and multi-layer networks. The single-layer makes up each of one input layer and output layer. Input, hidden, and output layers are the three different sorts of layers that make up a multiple-layer network. To decide the ANN output, each of these architectures needs an activation function or a node at the end of each neural network's output. To obtain the activation function, we can use either the binary sigmoid function or the bipolar sigmoid function. Equations (1) and (2) show the definitions of the binary sigmoid and bipolar sigmoid functions, respectively.

$$h(q) = \frac{1}{1 - e^{-q}} \tag{1}$$

$$h(q) = \frac{2}{1 + e^{-q}} - 1 \tag{2}$$

The activation function (1) is easy to differentiate and has the derivative of the sigmoid (1), as shown in (3). Similarly, by substituting the variable x into (4), we will obtain the derivative of the bipolar sigmoid.

$$h(q) = h(q)(1 - h(q)) \tag{3}$$

$$h(q) = \frac{(1 + h(q))(1 - h(q))}{2} \tag{4}$$

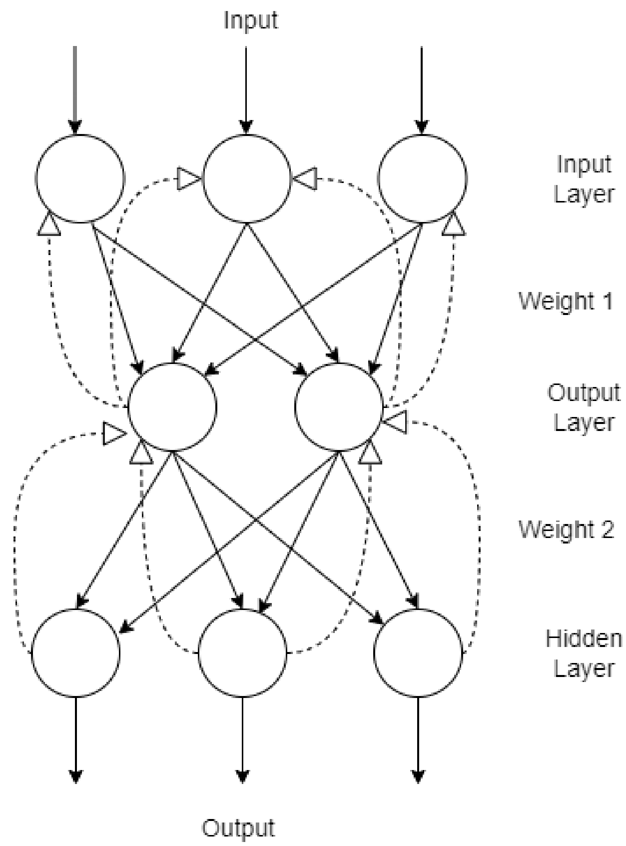


Figure 1. Illustration of Backpropagation Architecture

B. *Backpropagation*

The backpropagation network is a method widely used to deal with challenging problems. Since we used directed learning techniques to teach this algorithm, we can practice it in many case studies [27]. This method uses supervised learning with many layers or multiple-layer and is necessary to compare the targets and network output in utilizing supervised learning. The units of the hidden layer receive signals from the input network before passing them into nodes of the output layer. If the output on the artificial neural network does not achieve the goal, the hidden layer will take a step back before transmitting the input layer [28]. Figure 1 illustrates a backpropagation architecture.

3. METHODOLOGY

A. *The Procedures of Backpropagation ANN Algorithm*

There are three steps for training the backpropagation algorithm. Firstly, the algorithm uses feedforward and the defined activation function to measure the forward received from input to output. Secondly, it uses backpropagation to measure errors obtained from input and neurons in output to yield modified weights. Thirdly, it changes weights and biases to reduce error values obtained from the previous step. The following steps are detailed ANN procedures.

Step 1. Give weights using small-random values.

Step 2. Perform a looping of the feedforward process from step 3 to 8 until the condition "the error rate is greater than a specified value" is satisfied.

Feedforward process:

Step 3. Receive an input signal x_i for input neuron X_i where $i = 1, 2, ..., n$ and forward them to all nodes at the hidden layer.

Step 4. Sum weights of input signal for hidden layer unit Z_j where $j = 1, 2, ..., p$ using Equation (5).

$$Z_j^{(input)} = v_{oj} + \sum_{i=1}^{n} x_i v_{ij} \tag{5}$$

where v_{oj} is bias of input layer and v_{ij} is the weight of input layer i. It then measures output signal from hidden layers using the activation function, as presented by (6).

$$Z_j = f(Z_j^{(input)}) \tag{6}$$

Then, the output signal is sent to all units at the output layer.

Step 5. Receive a hidden signal z_j for output layer unit Y_k where $k = 1, 2, ..., m$ and sum weights of input signal using Equation (7).

$$Y_k^{(input)} = w_{ok} + \sum_{j=1}^{p} z_j w_{jk} \tag{7}$$

where w_{oj} is bias of hidden layer and w_{jk} is the weight of hidden layer j. It then applies the activation function (8) to calculate output signal.

$$Y_k = f(Y_k^{(input)}) \tag{8}$$

Backpropagation process:

Step 6. Receive target patterns that matches its training input pattern for output layer unit Y_k and calculate the error rate using Equation (9).

$$\delta_k = (t_k - y_k) f' Y_k^{(input)} \tag{9}$$

where t_k is output target. With giving a training parameter value α, it then calculates the weight and bias corrections that will be used later to update the w_{jk} value by using Equations (10) and (11), respectively.

$$\Delta w_{jk} = \alpha \delta_k z_j \tag{10}$$

$$\Delta w_{ok} = \alpha \delta_k \tag{11}$$

After obtaining this, it sends the value of δ_k into units at the below layers.

Step 7. Sum the error rates δ_k from the forward layers for hidden layer unit Z_j for $j = 1, 2, ..., p$ using (12) and multiply them by the derivative of the sigmoid using Equation (13) in order to calculate the error rate.

$$\delta_j^{(input)} = \sum_{k=1}^{m} \delta_k w_{jk} \tag{12}$$

$$\delta_j = \delta_j^{(input)} f'(Z_j^{input}) \tag{13}$$

With utilizing the training parameter value of α, it then calculates the weight and bias corrections that will be used later to update the w_{jk} value by using Equation (14) and (15), respectively.

$$\Delta v_{jk} = \alpha \delta_j x_i \tag{14}$$

$$\Delta v_{ok} = \alpha \delta_j \tag{15}$$

Updating processes of weights and bias:

Step 8. Update bias and its weight $j = 1, 2, ..., p$ for output layer Y_k for $k = 1, 2, ..., m$ using Equation (16).

$$w_{jk}(new) = w_{jk}(old) + \Delta w_{jk} \tag{16}$$

Similarly, update bias and its weight for input $i = 1, 2, ..., n$ and hidden layer Z_j where $j = 1, 2, ..., p$ and $k = 1, 2, ..., m$ using Equation (17).

$$v_{ij}(new) = v_{ij}(old) + \Delta v_{ij} \tag{17}$$

TABLE I. The Detailed Datasets Characteristics

No	Dataset	# of Files	# of File Sizes
1	The alphabet data	2653	3.04 Gigabytes
2	The undefined data	102	116.92 Megabytes

B. Data Collection

In this experimental study, we utilized two datasets, namely the alphabet and the undefined datasets. Firstly, we downloaded the alphabet dataset available at the Bisindo2 Kaggle repository [29] and the undefined dataset available at the Bengali sign Kaggle repository [30]. The alphabet dataset consists of 26 ISL alphabet gestures, while the undefined dataset contains 38 types of non-latin alphabet gestures. Table I presents the detailed dataset information. Secondly, we created 26 folders for the alphabet dataset and a folder for the undefined dataset. Thirdly, we filled 102 digital images for each folder of the 26 alphabets and 102 randomly selected analphabet images for the analphabet folder. Figure 2 is an example of ISL and analphabet gesture images.

C. Data Preprocessing

Before we input the data into the model, there is a data preprocessing that needs to be carried out and includes several stages:

1) *Split Data Training and Data Testing:* This stage divides the dataset randomly into testing and training sets. The ratios of the training and testing sets are 70:30 percent and 80:20 percent, respectively.

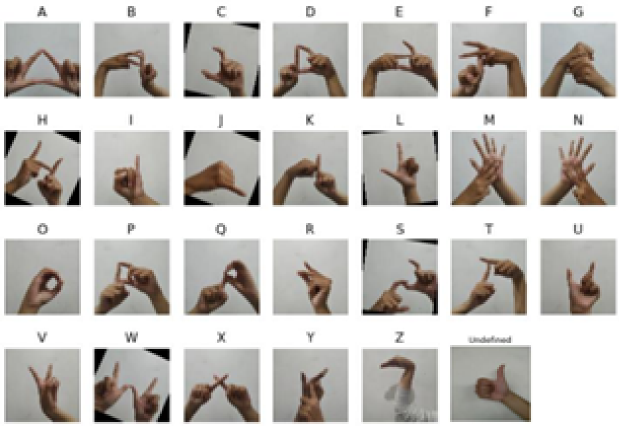


Figure 2. Sample Data of ISL and Undefined Image

2) *Scaling:* The scaling process in this study is a process to change image resolution. It is needed to equalize the size of every image so that it will perform the classification stage effectively. In addition, it scales every single image data into a size of 150x150 pixels.
3) *Segmentation:* There are several stages of segmentation carried out in this study.
   a) *Color image segmentation:* We used color image segmentation to separate and obtain the object in the photos that will be our target. A hand pattern resembling the ISL alphabet is the study's object goal. This procedure uses the color distance measuring algorithm and threshold value to assess how closely skin color resembles the colors of the pixels in an image. Figure 3 displays the outcome of color image segmentation.
   b) *Grayscale segmentation:* Images still consist of three color layers, red, green, and blue (RGB). The three color layers will have carried out three times calculation process too. Therefore, we need to convert it into a grayscale image. A grayscale image has only one color layer. Figure 4 shows the result of grayscale segmentation.
   c) *Binary segmentation:* A binary image is a black-and-white image that can determine which areas include objects and backgrounds. We also use the image to remove noise that still patches in the grayscale image. So, there is a difference in results between grayscale images and binary images when performing edge detection. Figure 5 exhibits the binary segmentation result.
   d) *Edge Detection:* Edge detection aims to mark parts detailed in the image by signing changes in the color intensity that are very high in the image dataset. The algorithm used for this stage is the canny algorithm [31]. Figure 6 presents the edge detection result.

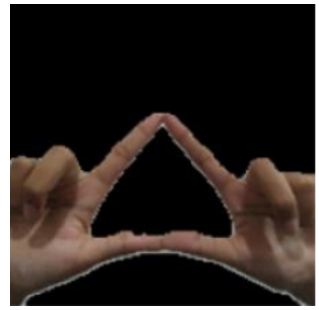


Figure 3. Result of Color Image Segmentation

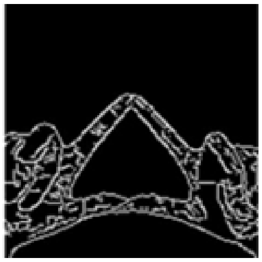
Figure 4. Result of Grayscale Segmentation

D. Training Model

The initial stage in developing the ANN backpropagation model is to determine the initial bias weight and the activation function used in this model. In this study, we utilized a random value to initialize bias weight which is a value ranging between -1 and 1. Then, the activation function used for the hidden and output layers are a binary sigmoid function and the SoftMax function, respectively. These functions are very suitable for multi-classes classification cases [32]. These two-activation functions will give output values within an interval between 0 and 1.

Furthermore, the network architecture used in this study is a multilayer network with three types of layers: input, hidden, and output. The total of nodes in the input layer is 22500 nodes determined based on the resolution size of the

Figure 5. Result of Binary Segmentation



(a) Grayscale Image

(b) Binary Image

Figure 6. Result of Grayscale Segmentation (a) from Grayscale Image and (b) from Binary Image

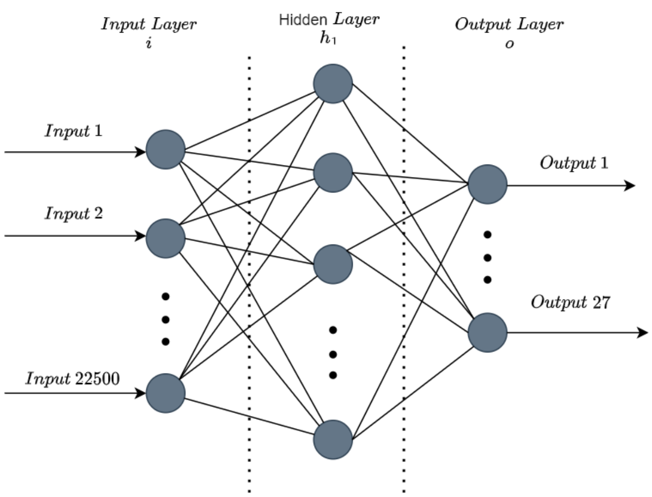


Figure 7. The Proposed ANN Architecture

images that we used, 150x150 pixels. The total of hidden layers in this study is one layer with the number of nodes determined by trial error with values of 25, 50, 75, and 100 nodes. Whereas the nodes in the output layer are 26 + 1 nodes, 26 nodes are for the ISL alphabet letter numbers, and one is for the output analphabet letter. Hence, there are 27 nodes in total at the output layer. Figure 7 is an illustration of the proposed ANN architecture.

After constructing the network, the next step is that we should determine the parameters used during the training process. Epoch is the number of iterations that occur during training, the batch size is the size of the pattern on the network before the weights are updated, and the learning rate is the quantity used as a rule in updating the weights for each batch. In this study, we used the batch size value of 25, the learning rate of 0.001, and epoch values of 300, 500, and 700. In addition, to prevent overfitting of the built model, we utilized L1 and L2 regularization algorithms [33] to reduce the error by fitting a function appropriately on the given training set. To calculate the error value, we used the loss function based on cross-entropy. This function estimates the performances of a classification model whose output is probability. From the experiment process, we will have the number of true-positives (TP), the number of true-negatives (TN), the number of false-positives (FP), and the number of false-negatives (FN).

The TP refers to information that is accurate and projected to be true. For instance, in this study, an image containing the alphabet A and the developed model classifies the image as alphabet A. The FP refers to data with a false value but is assumed to be true. For instance, the developed model predicts the alphabet A image as another alphabet besides the alphabet A. The FN refers to information that is accurate and projected to be false. For instance, an image containing the alphabet B and the developed model classifies the image as an alphabet other than the alphabet B. The TN refers to information that is inaccurate and projected to be false. The image does not contain the alphabet A, and the developed model does not classify the image as alphabet A.

Next, we used the F-measure [34] in this multi-class classification study to observe the model performance using precision and recall. We used precision to define the ability of a model to anticipate positive values accurately. Equation 18 shows how to obtain the *precision* value.

$$precision = \frac{TP}{TP + FP} \tag{18}$$

We used *recall* to define the percentage of successful outcomes that the model accurately predicts. Equation 19 shows how to calculate the recall value.

$$recall = \frac{TP}{TP + FN} \tag{19}$$

We used F-measure to yield values from consideration of both precision and recall. Equation 20 is a mathematical formula to calculate the F-measure value.

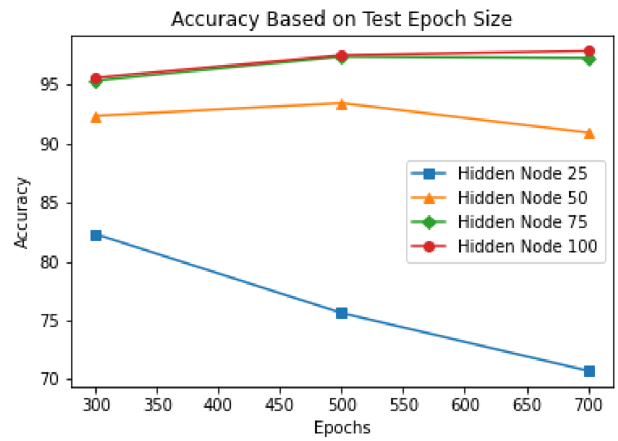
$$F_{measure} = 2 \times \frac{recall \times precision}{recall + precision} \tag{20}$$

Finally, it uses Equation 21 to calculate the accuracy value for determining the degree of similarity between the predicted and the actual values.

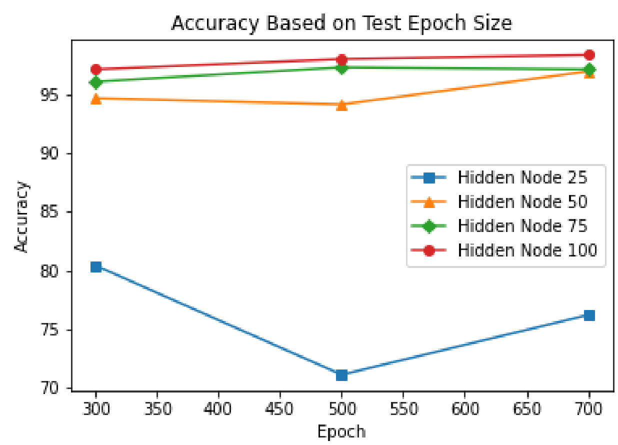
$$accuracy = \frac{TP + TN}{TP + TN + FP + FN} \tag{21}$$

TABLE II. The accuracy comparison results between 80:20% and 70:30% of the testing dataset for some different epochs and hidden nodes

epochs	Hidden Nodes	Accuracy for	
		70:30%	80:20%
300	25	82.32%	80.42%
	50	92.35%	94.71%
	75	95.34%	96.12%
	100	95.58%	97.18%
500	25	75.63%	71.08%
	50	93.43%	94.18%
	75	97.37%	97.35%
	100	97.47%	98.06%
700	25	70.73%	76.19%
	50	90.92%	97.00%
	75	97.25%	97.18%
	100	97.85%	98.41%



(a) 70:30% of the dataset



(b) 80:20% of the dataset

Figure 8. Comparison graph of testing accuracy for (a) 70:30 percent data proportion and (b) 70:30 percent data proportion

4. RESULTS AND DISCUSSION

After training the model, we obtained the highest testing accuracy when we attempted to perform for the number of hidden nodes as many as 100 hidden nodes and epochs of 700. The best result for accuracy came out to 98.41 percent for an 80:20 percent data proportion, while the best value for accuracy reached 97.85 percent for a 70:30 ratio. In addition, the higher number of epochs used on the model does not give better accuracy. When we attempted to analyze the epoch values of 300 and 500 for 25 hidden nodes, we obtained the epoch of 300 impacted the accuracy became decrement. Table II presents the comparison testing accuracy of 70:30 percent and 80:20 percent data proportions.

Figure 8 shows the comparison graph of testing accuracy. It indicates that the average accuracy obtained by each hidden node with different epoch sizes is above 90 percent. However, the higher accuracy for 25 hidden nodes only reaches values in an interval between 70 and 85 percent. Its performance was still not good enough to recognize the ISL image properly. However, when the model utilized 50 and 75 hidden nodes, it started to yield a good performance relatively and identify the ISL alphabet images correctly.

The precision is the level of accuracy between the predicted images to be correct from the total predicted images. Meanwhile, recall is the level of accuracy of the number of correctly predicted images from the total owned mages. According to Table III, we obtained that the recall and precision values of all alphabets are almost 100 percent. The recall value did not get 100 percent for the alphabets of A, I, M, O, V, and Undefined only, and the precision value did not reach 100 percent for the alphabets of J, N, Q, X, and Y. Alphabets having a 100 percent on recall indicates that each image predicted to be correct for that category. Meanwhile, for alphabets having 100 percent on recall, but the obtained precision value does not reach 100 percent, it indicates that the model predicted each of the alphabets belongs to other alphabet categories.

Alphabet images with a recall value of 1.00 indicate that the model has correctly predicted each image of the alphabet belonging to the appropriate category. Whereas alphabet images that obtained a recall value of 1.00 but did not reach the precision value of 1.00 indicate that the model predicted each alphabet as other alphabetic categories. Figure 9 shows an example of how the model predicted alphabet image A as the image category of alphabet J.

Furthermore, we compared the accuracy result obtained from our model with three existing classifiers, such as Chain Code Contour (CCC) [35], k-NN [36], and CNN [37]. The result showed that the accuracy of our model is superior compared with those three classifiers. The accuracy performance of CCC reached 94%. The k-NN classifier demonstrated accuracy performance relatively far below our

TABLE III. The Accuracy Result of Classification Based on F-Measure Performance for 80:20% of the Dataset

Alphabets	Recall	Precision	F-Measure	Support
A	95.45%	100%	97.67%	21
B	100%	100%	100%	21
C	100%	100%	100%	21
D	100%	100%	100%	21
E	100%	100%	100%	21
F	100%	100%	100%	21
G	100%	100%	100%	21
H	100%	100%	100%	21
I	91.30%	100%	95.45%	21
J	100%	95.24%	97.56%	21
K	100%	100%	100%	21
L	100%	100%	100%	21
M	91.30%	100%	95.45%	21
N	100%	90.48%	95%	21
O	91.30%	100%	95.45%	21
P	100%	100%	100%	21
Q	100%	90.48%	95%	21
R	100%	100%	100%	21
S	100%	100%	100%	21
T	100%	100%	100%	21
U	100%	100%	100%	21
V	95.45%	100%	97.67%	21
W	100%	100%	100%	21
X	100%	95.24%	97.56%	21
Y	100%	85.71%	92.31%	21
Z	100%	100%	100%	21
Undefined	95.45%	100%	97.67%	21
Accuracy			98.41%	567
Macro Avg	98.53%	98.41%	98.40%	567
Weight Avg	98.53%	98.41%	98.40%	567

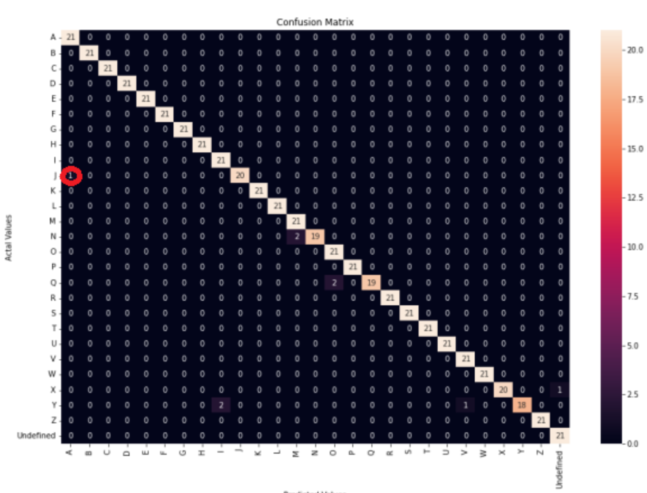


Figure 9. Confusion Matrix at the Accuracy of 98.41%

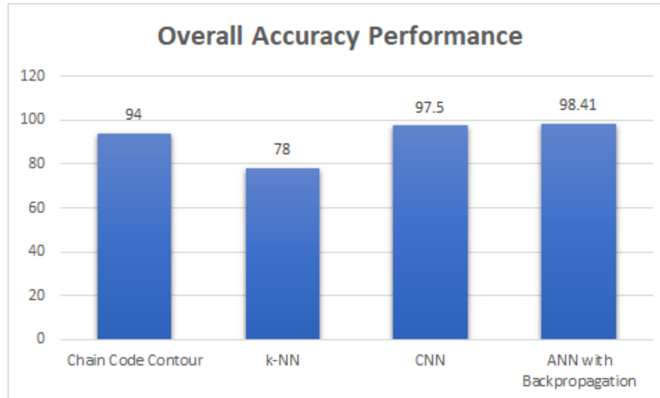


Figure 10. The Accuracy Performance Comparison Results

model, which is only 78% of accuracy. However, the CNN model shows that its accuracy performance in classifying ISL alphabets is slightly smaller than our model, which is 97.5% of accuracy. Figure 10 presents the accuracy comparison results of the four classifiers.

5. CONCLUSIONS AND FUTURE STUDIES

According to findings in this study's experimental, the ANN based on the backpropagation model demonstrated the ability to recognize ISL alphabets. The results showed that the maximum accuracy achieved utilizing 80:20 data proportions, with 100 hidden nodes and 700 epochs, was 98.41%. The experiment attempt to substitute different epochs and hidden nodes into the model and the results suggest varying accuracy levels. It is evident that when we want to generate a better accuracy result, we should consider choosing the optimal number of epochs and hidden nodes to apply to our model.

Although the accuracy of our developed framework methodology is higher than the other three classifiers, we realize that this algorithm still has weaknesses in predicting images that rotate slightly in their gesture positions. Therefore, we need a lot of training data with a variety of gesture positions to be applied the prediction process to this algorithm. Additionally, the model built in this study was only limited to accuracy values, and the dataset was limited to the ISL alphabet (A-Z). Therefore, ISL sign language recognition still has much room for improvement and future studies, such as recognition based on sign language words or an application for real-time recognition. In addition, improving weights in backpropagation iterations can be effectively constructed using decision-making methods such as an integrated objective weighting [38], a hybrid bipolar neutrosophic-EDAS approach [39], and improved trapezoidal fuzzy neutrosophic sets [40][41][42].

References

[1] R. I. Ministry of Social Affairs, "Ministry of social affairs prepares dtks as data source for social assistance for people with disabilities in 2021," 2021, last accessed 15 June 2023. [Online]. Available: https://rb.gy/sxqol

[2] S. He, “Research of a sign language translation system based on deep learning,” in *2019 International Conference on Artificial Intelligence and Advanced Manufacturing (AIAM)*, 2019, pp. 392–396.

[3] T. Witchuda, A. Wiranata, S. Maeda, and C. Premachandra, “Reservoir computing model for human hand locomotion signal classification,” *IEEE Access*, vol. 11, pp. 19 591–19 601, 2023.

[4] B. I. Bitachon, M. Eppenberger, B. Baeuerle, and J. Leuthold, “Reducing training time of deep learning based digital backpropagation by stacking,” *IEEE Photonics Technology Letters*, vol. 34, no. 7, pp. 387–390, 2022.

[5] M. Gong, J. Liu, A. K. Qin, K. Zhao, and K. C. Tan, “Evolving deep neural networks via cooperative coevolution with backpropagation,” *IEEE Transactions on Neural Networks and Learning Systems*, vol. 32, no. 1, pp. 420–434, 2021.

[6] J. Zhang, C. Li, and T. Zhang, “Application of back-propagation neural network in the post-blast re-entry time prediction,” *Knowledge*, vol. 3, no. 2, pp. 128–148, 2023. [Online]. Available: https://www.mdpi.com/2673-9585/3/2/10

[7] A. H. Zuraidah, N. A. Windarko, and R. P. Eviningsih, “Short-term electrical load prediction using ann-backpropagation,” in *2021 International Conference on Artificial Intelligence and Computer Science Technology (ICAICST)*, 2021, pp. 13–18.

[8] W. Li, L. Cui, Y. Zhang, Z. Cai, M. Zhang, W. Xu, X. Zhao, Y. Lei, X. Pan, J. Li, and Z. Dou, “Using a backpropagation artificial neural network to predict nutrient removal in tidal flow constructed wetlands,” *Water*, vol. 10, no. 1, 2018. [Online]. Available: https://www.mdpi.com/2073-4441/10/1/83

[9] A. Y. Al-Bakri and M. Sazid, “Application of artificial neural network (ann) for prediction and optimization of blast-induced impacts,” *Mining*, vol. 1, no. 3, pp. 315–334, 2021. [Online]. Available: https://www.mdpi.com/2673-6489/1/3/20

[10] A. Maulana, T. R. Noviandy, R. Idroes, N. R. Sasmita, R. Suhendra, and I. Irvanizam, “Prediction of kovats retention indices for fragrance and flavor using artificial neural network,” in *2020 International Conference on Electrical Engineering and Informatics (ICELTICs)*, 2020, pp. 1–5.

[11] R. Idroes, T. Noviandy, A. Maulana, R. Suhendra, N. Sasmita, M. Muslem, G. Idroes, P. Kemala, and I. Irvanizam, “Application of genetic algorithm-multiple linear regression and artificial neural network determinations for prediction of kovats retention index,” *International Review on Modelling and Simulations (IREMOS)*, vol. 14, no. 2, 2021.

[12] A. A. Mas’ud, R. Albarracín, J. A. Ardila-Rey, F. Muhammad-Sukki, H. A. Illias, N. A. Bani, and A. B. Munir, “Artificial neural network application for partial discharge recognition: Survey and future directions,” *Energies*, vol. 9, no. 8, 2016. [Online]. Available: https://www.mdpi.com/1996-1073/9/8/574

[13] A. Orazaev, P. Lyakhov, V. Baboshina, and D. Kalita, “Neural network system for recognizing images affected by random-valued impulse noise,” *Applied Sciences*, vol. 13, no. 3, 2023. [Online]. Available: https://www.mdpi.com/2076-3417/13/3/1585

[14] M. I. Dieste-Velasco, “Application of a pattern-recognition neural network for detecting analog electronic circuit faults,” *Mathematics*, vol. 9, no. 24, 2021. [Online]. Available: https://www.mdpi.com/2227-7390/9/24/3247

[15] R. Andrian, B. Hermanto, and R. Kamil, “The implementation of backpropagation artificial neural network for recognition of batik lampung motive,” *Journal of Physics: Conference Series*, vol. 1338, no. 1, p. 012062, oct 2019. [Online]. Available: https://dx.doi.org/10.1088/1742-6596/1338/1/012062

[16] M. Hossain, M. S. Haque, M. Arifuzzaman, and S. M. Zakir Hossain, “Artificial neural network based system for distorted image recognition,” in *3rd Smart Cities Symposium (SCS 2020)*, vol. 2020, 2020, pp. 503–508.

[17] E. Bravo-López, T. Fernández Del Castillo, C. Sellers, and J. Delgado-García, “Landslide susceptibility mapping of landslides with artificial neural networks: Multi-approach analysis of backpropagation algorithm applying the neuralnet package in cuenca, ecuador,” *Remote Sensing*, vol. 14, no. 14, 2022. [Online]. Available: https://www.mdpi.com/2072-4292/14/14/3495

[18] M. Iqbal Quraishi, J. Pal Choudhury, and M. De, “Image recognition and processing using artificial neural network,” in *2012 1st International Conference on Recent Advances in Information Technology (RAIT)*, March 2012, pp. 95–100.

[19] A. Calado, V. Errico, and G. Saggio, “Toward the minimum number of wearables to recognize signer-independent italian sign language with machine-learning algorithms,” *IEEE Transactions on Instrumentation and Measurement*, vol. 70, pp. 1–9, 2021.

[20] Y. Yu, X. Chen, S. Cao, X. Zhang, and X. Chen, “Exploration of chinese sign language recognition using wearable sensors based on deep belief net,” *IEEE Journal of Biomedical and Health Informatics*, vol. 24, no. 5, pp. 1310–1320, 2020.

[21] W. Gao, G. Fang, D. Zhao, and Y. Chen, “A chinese sign language recognition system based on sofm/srn/hmm,” *Pattern Recognition*, vol. 37, pp. 2389–2402, 12 2004.

[22] 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*, vol. 20, no. 5, pp. 1281–1290, 2016.

[23] S. Daniels, N. Suciati, and C. Fathichah, “Indonesian sign language recognition using yolo method,” *IOP Conference Series: Materials Science and Engineering*, vol. 1077, no. 1, p. 012029, feb 2021. [Online]. Available: https://dx.doi.org/10.1088/1757-899X/1077/1/012029

[24] D. Indra, Purnawansyah, S. Madenda, and E. P. Wibowo, “Indonesian sign language recognition based on shape of hand gesture,” *Procedia Computer Science*, vol. 161, pp. 74–81, 2019, the Fifth Information Systems International Conference, 23-24 July 2019, Surabaya, Indonesia. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S1877050919318113

[25] U. Fadlilah, A. K. Mahamad, and B. Handaga, “The development of android for indonesian sign language using tensorflow lite and cnn: An initial study,” *Journal of Physics: Conference Series*, vol. 1858, no. 1, p. 012085, apr 2021. [Online]. Available: https://dx.doi.org/10.1088/1742-6596/1858/1/012085

[26] S. P. Mahasagara, A. Alamsyah, and B. Rikumahu, “Indonesia infrastructure and consumer stock portfolio prediction using artificial neural network backpropagation,” in *2017 5th International Con-*

ference on Information and Communication Technology (ICoIC7), 2017, pp. 1–4.

[27] S. Amalia, "Identification of number using artificial neural network backpropagation," *MATEC Web of Conferences*, vol. 215, p. 01011, 01 2018.

[28] N. Salman, A. Lawi, and S. Syarif, "Artificial neural network backpropagation with particle swarm optimization for crude palm oil price prediction," *Journal of Physics: Conference Series*, vol. 1114, p. 012088, 11 2018.

[29] Z. Riestiya, "Bisindo2 dataset," 2020, last accessed 15 June 2023. [Online]. Available: https://www.kaggle.com/datasets/riestiyazain/bisindo2?select=Bisindo

[30] A. M. Rafi and N. Nawal, "Image-based bengali sign language alphabet recognition for deaf and dumb community," 10 2019.

[31] J. Canny, "A computational approach to edge detection," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. PAMI-8, no. 6, pp. 679–698, 1986.

[32] M. Kaloev and G. Krastev, "Comparative analysis of activation functions used in the hidden layers of deep neural networks," in *2021 3rd International Congress on Human-Computer Interaction, Optimization and Robotic Applications (HORA)*, 2021, pp. 1–5.

[33] R. Tibshirani, "Regression shrinkage and selection via the lasso," *Journal of the Royal Statistical Society. Series B (Methodological)*, vol. 58, no. 1, pp. 267–288, 1996. [Online]. Available: http://www.jstor.org/stable/2346178

[34] D. J. Hand, P. Christen, and N. Kirielle, "F*: an interpretable transformation of the f-measure," *Machine Learning*, vol. 110, no. 3, pp. 451–456, Mar 2021. [Online]. Available: https://doi.org/10.1007/s10994-021-05964-1

[35] D. Indra, S. Madenda, and E. Prasetyo, "Recognition of bisindo alphabets based on chain code contour and similarity of euclidean distance," *International Journal on Advanced Science, Engineering and Information Technology*, vol. 7, p. 1644, 10 2017.

[36] F. Humaira, S. Supria, D. Herumurti, and K. Widarsono, "Real time sibi sign language recognition based on k-nearest neighbor," *Proceeding of the Electrical Engineering Computer Science and Informatics*, vol. 5, 11 2018.

[37] S. Suharjito, N. Thiracitta, and H. Gunawan, "Sibi sign language recognition using convolutional neural network combined with transfer learning and non-trainable parameters," *Procedia Computer Science*, vol. 179, pp. 72–80, 01 2021.

[38] I. Irvanizam, Z. Zulfan, P. F. Nasir, M. Marzuki, S. Rusdiana, and N. Salwa, "An extended multimoora based on trapezoidal fuzzy neutrosophic sets and objective weighting method in group decision-making," *IEEE Access*, vol. 10, pp. 47 476–47 498, 2022.

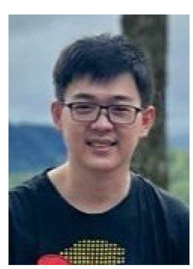
[39] I. Irvanizam, I. Syahrini, N. Zi, N. Azzahra, M. Iqbal, M. Marzuki, and M. Subianto, "An improved edas method based on bipolar neutrosophic set and its application in group decision-making," *Applied Computational Intelligence and Soft Computing*, vol. 2021, 10 2021.

[40] I. Irvanizam, M. Siraturahmi, A. M. B. Irwansyah, P. F. Nasir, Z. Zulfan, and N. Salwa, "A hybrid dematel-edas based on multi-criteria decision-making for a social aid distribution problem," in *2022 IEEE International Conference on Cybernetics and Computational Intelligence (CyberneticsCom)*, 2022, pp. 341–346.

[41] I. Irvanizam, N. Azzahra, I. Nadhira, Z. Zulfan, M. Subianto, and I. Syahrini, "Multiple criteria decision making based on vikor for productive economic endeavors distribution problem," in *2021 Sixth International Conference on Informatics and Computing (ICIC)*, 2021, pp. 1–6.

[42] I. Irvanizam, N. Nazaruddin, and I. Syahrini, "Solving decent home distribution problem using electre method with triangular fuzzy number," in *2018 International Conference on Applied Information Technology and Innovation (ICAITI)*, 2018, pp. 139–144.

Irvanizam Irvanizam was born in Teupin Raya, Aceh, Indonesia in 1981. He is currently an Associate Professor in applied informatics or computer science at the Department of Informatics, Universitas Syiah Kuala, Banda Aceh, Indonesia. He obtained the B.Sc. degree in mathematics science majoring informatics from Universitas Syiah Kuala, Banda Aceh, Indonesia, in 2002 and the M.Sc. degree in computer science from Free University of Bozen-Bolzano, Bolzano, Italy, in 2010. He is currently a PhD student in Computer Science from Universitas Sumatera Utara, Medan, Indonesia. He is also the author of publications in international journals and conference proceedings, many of which are indexed by Scopus and Web of Science such as *IEEE Access*, *Applied Computational Intelligence and Soft Computing*, *Axioms*, *Advances in Fuzzy Systems*, and some famous IEEE proceedings. He regularly serves as a reviewer for some prestigious technical and international journals such as *International Journal of Computing and Digital Systems (IJCDS)*, *Applied Soft Computing*, *Expert Systems with Applications*, *Artificial Intelligence Review*, *Soft Computing*, *Case Studies in Thermal Engineering*, *International Journal of Information Technology and Web Engineering (IJITWE)*, etc. His researches focus on neutrosophic and fuzzy sets, soft computing, group decision-making, and image processing.



Ivan Horatius was born in Banda Aceh, Aceh, Indonesia in 2000. He obtained his B.Sc in computer science (informatics) from Universitas Syiah Kuala, Banda Aceh, Indonesia in 2021. His research interests include intelligence systems, big data, data mining, image-processing, and information retrieval. From 2019-2021, he was also a researcher at the laboratory of Data Science and Artificial Intelligence, Department of Informatics, Universitas Syiah Kuala. He is working on data analytics and image processing.

Hizir Sofyan was born in Banda Aceh, Aceh, Indonesia in 1968. He is a Professor in Applied Statistics at Universitas Syiah Kuala, Banda Aceh, Indonesia. Moreover, he serves as the chairman of the International Statistical Institute (ISI) South East Asia Outreach Committee. In 1992, he graduated with a B.Sc. in Statistics from Indonesia's Sepuluh Nopember Institute of Technology in Surabaya. In 2003, he obtained his PhD degree from the Institute of Statistics and Econometrics, School of Business and Economics, Humboldt-Universitat zu Berlin, Germany. His supervisors were Prof. Dr. Wolfgang Haerdle from Statistics Department, Humboldt Universitaet zu Berlin, Germany, and Prof. Oliver Guenther, PhD (President of Potsdam University, Germany). He is currently the Head of Graduate School, Universitas Syiah Kuala, Banda Aceh, Indonesia. His current research interests include computational statistics, soft computing, fuzzy modelling, artificial-intelligence, advanced data-analysis, statistical geo-spatial, and deep learning.