A Machine Learning-Based Ambiguous Alphabet Recognition for Indonesian Sign Language System (SIBI)

Yoan Purbolingga[1,2], Ahmad Ridwan[3]*, Dila Marta Putri[1]
[1]Department of Electrical Engineering, Institut Teknologi Bisnis Riau
[2]Department of Computer Engineering, Faculty of Information Technology, Universitas Andalas
[3]Department of Informatics, Faculty of Computer Science, Universitas AMIKOM Yogyakarta
e-mail: [1]ypurbolingga@gmail.com, *[2]ahmadridwan@amikom.ac.id, [3]putri.dilamarta@gmail.com

Abstract

One of the communication problems for deaf people is the inhibition of verbal communication. This is due to the limited hearing function, which has an impact on the imperfection of language sound reception. To communicate with deaf people, extraordinary communication is needed so that the meaning of the conversation can be conveyed properly. Sign language is the main communication medium for deaf people. However, in the use of sign language, there are ambiguous letters, namely "D ", "E", "M", "N", "R", "S", and "U". This research uses the chain code method to identify and reconstruct the shape of hand gesture objects. Then, to solve the problem of the ambiguity of alphabet letters, an artificial intelligence method, namely K-Nearest Neighbors (K-NN), is used. The sample used consists of 350 real-time images with variations in object recognition accuracy. Based on the research using chain code and K-NN classification method, it can be concluded that the recognition of ambiguous letters in sign language has 245 training data for K-NN, which has 88.76% accuracy, and 105 test data with 90% accuracy. This test data is divided into seven letters: "D", "E", "M", "R", and "U" at 100%, and "N" and "S" at 98.88%.

Keywords— Sign Language, Image Processing, SIBI, Machine Learning, Chain Code

1. INTRODUCTION

One form of human deficiency is hearing disability, commonly called deafness, a condition in which a person experiences hearing loss. [1, 2, 3]. This limitation causes deaf people to be unable to communicate appropriately. The existence of this communication disorder indirectly makes it difficult for deaf people to interact with the surrounding community. This can have an impact on the society of deaf people because of the difficulty of interacting with the surrounding environment [4,5]. Therefore, deaf people usually utilize the sense of sight to communicate with others [6,7]. The communication referred to here refers to sign language, an expression of a word that uses hand movements whose form has been approved by the wearer and expressed in spoken language [7, 8, 9].

The sign languages often used in Indonesia are the Indonesian Sign Language System (SIBI) and Indonesian Sign Language (BISINDO) [10]. In the application in daily life, SIBI has not been running optimally because many deaf people do not know Indonesian grammar [10]. In contrast, BISINDO is a sign language that has existed since birth and has been developed scientifically [11]. The official sign language in Indonesia is SIBI. However, using sign language is difficult for the general public, and most normal people find it challenging to interpret the words of these deaf people [12][13][14].

In dealing with these problems, a method is needed to facilitate communication or interaction with deaf people, and this method will produce output in the form of text that is easy to understand and will become a sign language translator for people who communicate directly [12][15][16]. One solution to this problem is to use a digital image processing system to translate the sign language in this study using the chain code method. Chain code is a method that identifies and reconstructs the object's shape. Each point on the contour of the object is represented in the form of a number representing the direction of movement of a moment [4].

Captured hand images from the camera will be recognized as letters of the alphabet using a chain code pattern using a chain code. However, in this study, the use of chain codes can only show the shape of hand gestures that are different in shape, while in SIBI, several hand gestures are similar in shape, including the letters "D," "E," "M," "N," "R," "S," and "U". Therefore, K-Nearest Neighbor (K-NN) is needed to help recognize the shape of the ambiguous letters, where the results of the hand gesture recognition chain code of the sign language alphabet letters are used to make accurate decisions that produce raw data. As well as grouping object chain code patterns using the K-NN intelligent system [17]. Various studies of sign language translator systems have been carried out.

This technology is undoubtedly challenging to design and deploy. Ridwang et al. (2019) did a staged study to obtain excellence. Their study suggested a Naive Bayes classification technique that uses Leap Motion to classify hand movements in the Indonesian Sign Language System (SIBI). This algorithm estimates the probability value for each class of data provided and then predicts the class based on that value. The evaluation findings revealed an average accuracy of 96%. However, there were some prediction errors in the letters "M," "N," and "O," as well as the letters "Z," "U," and "L." The input data generated prediction mistakes in the letters "J" and "L," which were caused by the input data from Leap Motion not being able to recognize hidden fingers [18]. In artificial intelligence, approaches for creating sign languages are constantly evolving. Afifah et al. (2021) did research that demonstrates this. The chain technique is used in this research to discover letter features in the Indonesian Sign Language System (SIBI). The letter detection procedure comprises preprocessing, edge detection, picture extraction, and letter matching. The Manhattan distance approach is utilized during the segmentation phase, and then the RGB image is converted to grayscale. Grayscale values are transferred to the image shape, yielding probabilities 1 and 0. The training data evaluation results reveal an average accuracy of 91% using inputs in the form of photos [19].

Sign language detection improved in subsequent years. Putri et al. (2022) conducted a real-time Indonesian sign language detection investigation employing long short-term memory (LSTM). The language model utilized is Indonesian Sign Language (BISINDO). The acquired data is classified using LSTM, and movement frames are detected starting with the hands, face, and body, utilizing Media Pipe Holistic. This study employs 30 BISINDO sign vocabularies. The data is labeled with an array and then divided into 95% training data and 5% test data. The study found that testing the first ten classes with bidirectional LSTM with 1000 epochs, 64 hidden layers, and batch size 32 yielded an accuracy of 92%. However, testing 30 classes with 2 LSTM layers trained for 500 epochs, 64 hidden layers, and a batch size of 64 yields a 65% accuracy, indicating a decline in quality [20]. Imam et al. (2022) conducted another study introducing Indonesian Sign Language (SIBI) utilizing machine learning techniques. Multinomial Logistic Regression and Random Forest are the models that are employed. Twenty-four letters are included in this gesture recognition detection. Nevertheless, the letters "J" and "Z" are not included because their hand signals are dynamic. With 750 decision trees (trees), the Random Forest model's average accuracy is 97.19%. However, the letters "K" and "R," which have comparable hand signals, contain several inaccuracies. Furthermore, the Multinomial Logistic Regression model predicts the letters "U" and "V" with an average accuracy of 95.73%, with occasional mistakes [21].

There are still issues with several artificial intelligence development models that still need to be implemented, such as slow computation while operating in real-time and certain alphabetic letters that are difficult to recognize, which results in a less-than-ideal learning process. In light of this, research into sign language sign detection is ongoing, with increasingly sophisticated but portable techniques being developed to produce the optimal model. According to research by Rivan et al. (2020), K-Nearest Neighbors (K-NN) is one technique that has this potential. The K-Nearest Neighbors (K-NN) classification approach demonstrated the highest accuracy of approximately 72% in their study when it came to identifying linguistic signs from American Sign linguistic (ASL) for the letters A through Y, excluding the letters J and Z. The Histogram of Oriented Gradients (HOG) is used by this K-NN model method to determine gradient values in certain regions of an image. After that, a matrix is produced by linearizing the HOG data using Linear Discriminant Analysis (LDA). The maximum precision, recall, and accuracy scores are found at k=3, k=5, and k=11 for each distance in test results employing Euclidean, Manhattan, and Chebyshev distance matrices with k values of 3, 5, 7, 9, and 11 [22]. Hasma et al. also carried out similar studies in 2022. The K-Nearest Neighbors (K-NN) and SURF algorithms are used in this study to identify Indonesian gestures. Numbers and the Indonesian sign system alphabet (SIBI) are included in the detection process. According to the test results and test data, the highest accuracy of 90% is obtained by the value K=7, 88% by the value K=8, 86% by the value K=9, and 87.5% by the value K=10 [23].

The correct action is to identify and detect language indicators using the K-NN approach. Implementing a real-time, accurate, and efficient language sign recognition system that can identify different language signals worldwide is possible. The findings of this study demonstrated good classification and recognition of various hand hues and forms. Furthermore, the K-NN approach to language sign recognition offers a highly accurate model that can be improved [24][25]. Therefore, this study aims to develop an Indonesian language sign recognition model (SIBI) using the Chain Code method to detect hand gestures in sign language. Additionally, to improve the prediction of each ambiguous letter, researchers also use machine learning in the form of the K-Nearest Neighbor (K-NN) algorithm to overcome the ambiguity of the letters "D," "E," "M," "N," "R," "S," and "U".

2. RESEARCH METHODS

This quantitative research focuses on collecting and analyzing numerical data to test the established hypotheses. In the image pre-processing stage, image data is obtained from the camera in real-time. This process improves the quality of the image results and obtains binary images. After the binary image is obtained, the next step is to use edge detection using the chain code algorithm. This algorithm will later detect the results of the edge detection. The line recognition process starts with chain code tracing, chain code smoothing, and chain code modification. This chain code functions as a feature extraction for hand gesture pattern recognition. The results of this chain code detector will later be used by K-Nearest Neighbor (KNN) as training and testing data. Suppose the chain code detects a hand gesture that is similar or ambiguous.

In that case, the method is to classify the object into a category that matches most of its closest neighbors. The KNN algorithm will choose the most common label among the nearest neighbors. To overcome this ambiguity, KNN requires a method for determining the value of k or the number of neighbors to obtain the KNN model with the best accuracy. Therefore, training is carried out on several variations of k values, and this is done to determine the best k value in determining sign language hand gestures. The variation of k values used in this training are k = 1, 2, 3, 4, and 5. Training data on variations of k values uses the k-fold cross-validation method, which aims to provide more optimal and accurate accuracy to model performance during training with variations of k values. The flowchart of this research can be seen in Figure 1.

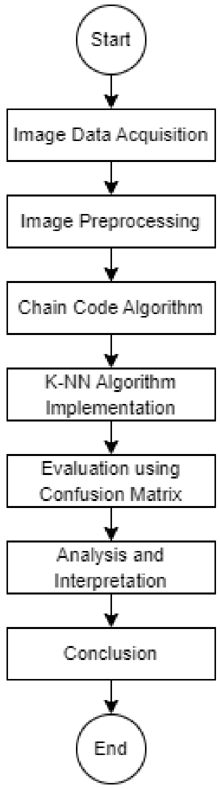


Figure 1. System Research Flowchart

Figure 1 above shows the stages in translating sign language hand gestures. This stage starts with acquiring images by taking pictures with a camera. The image obtained will later be processed in image pre-processing. This process improves the quality of the image results and obtains binary images. After the binary image is obtained, edge or contour detection, such as the Canny algorithm, is used.

The chain code algorithm will later detect the edge detection results. The line recognition process begins with chain code tracing, chain code smoothing, and chain code modification. This chain code functions as a feature extraction for recognizing hand gesture patterns. The results of this chain code detector will later be used for K-NN as training data and testing data. Suppose the chain code of a hand gesture is similar or ambiguous. In that case, the method is to classify the object into a category that matches most of its closest neighbors. KNN will choose the most common label among the nearest neighbors.

To overcome this ambiguity, K-NN requires a method for determining the value of k, or the number of neighbors, to obtain a K-NN model with the best accuracy. Therefore, training is carried out on several variations of k values and determining the best k value in sign language hand gestures. The variations of k values used in this training are k equal to 1, 2, 3, 4, and 5. Training of training data on variations of k values uses the k-fold cross-validation method. K-fold cross-validation aims to provide more optimal accuracy to model performance during training with variations of k values.

2.1. Image Data Acquisition

This research uses a total of 350 ambiguous letter sample data from the Indonesian Language Sign System (SIBI), including "D," "E," "M," "N," "R," "S," and "U". Where each letter sample consists of 50 data samples taken using a camera with a white background. In this testing process, the number of samples of each alphabet letter is divided into 70:30. This is useful as the beginning of making the K-Nearest Neighbor, where 70% is training data and 30% is testing data. Details of the total data used in this study are divided into training data and testing data, as shown in Table 1.

Table 1. Number of Total Data

Alphabet Letter	Number of Data	Number of Training Data	Number of Testing Data
D	50	35	15
E	50	35	15
M	50	35	15
N	50	35	15
R	50	35	15
S	50	35	15
U	50	35	15
Amount	**350**	**245**	**105**

In making this K-NN model, training data and test data are needed to test the performance of the trained model where the data is obtained from the chain code search, which produces output in the form of frequencies of 9 features including many codes, number of searches in direction 0, number of searches in direction 1, number of directions 2, number of directions 3, number of directions 4, number of directions 5, number of directions 6, and number of directions 7. The training data used was 245 samples, while the test data amounted to 105 samples. This training data is used when training the K-NN model, while the test data will be input when testing the trained K-NN. This frequency count data will later be given its respective class targets, as seen in Table 2.

Table 2. Data Class Target Label

Target Representation	Target
Letter D 	1
Letter E 	2
Letter M 	3
Letter N 	4
Letter R 	5
Letter S 	6
Letter U 	7

2.2. Image Processing

The preprocessing stage converts the original image into an image that the algorithm can process. In this stage, RGB images from the camera will be transformed into HSV images, and the goal is to calculate the color dependence on image intensity. The following process is segmentation, which separates the object from its background and separates the object from objects that obstruct it. The segmented image will be converted into a grayscale image. This process allows for focusing on pixel intensity information without considering color variations, and the chain code can detect object contours on one color channel. Then, the gray image will be given a certain value threshold to convert into a binary image. This process helps separate the object (hand) from the background.

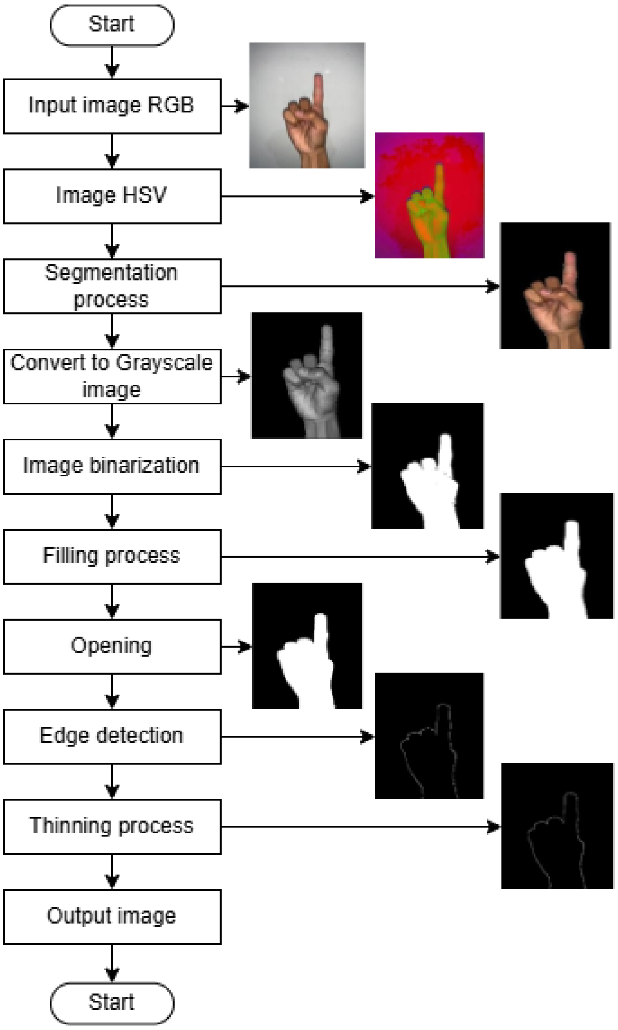


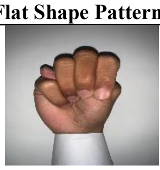
Figure 2. Preprocessing Stages

The next stage is the filling process, which overcomes the areas that still need to be filled due to the change of image from RGB format to a binary image. This phenomenon occurs due to the unevenness of the light effect in the image. The image that has gone through the filling process will continue with the opening process. This aims to eliminate interconnected pixels below 1000 pixels. The next stage goes into edge detection using edge operators to identify sharp changes in the intensity of the binary image. This helps highlight the hand's edges, essential for recognizing gestures. The final stage is thinning, where the results of the edge detection line will be thinned to produce a connected line. The preprocessing stages in this research can be seen in Figure 2.

2.3. Chain Code

The stages of the chain code algorithm in detecting hand gestures in this study use eight chain code directions in a counterclockwise direction. Horizontal and vertical scanning is carried out to determine the search's starting point. This aims to get white color or edge pixels that are white the first time. After all the chain codes of the test image are obtained, the chain code is smoothed to obtain a smoother chain code. The following process is to modify the chain code by replacing the smoothed chain code if an error occurs. Pattern recognition on flat shapes refers to the chain code for hand gestures. Table 3 shows the hand chain code pattern in this research.

Table 3. Hand Chain Code

Flat Patterns	Flat Shape Patterns	Chain Code
Letter M		(4, 5, 4, 6, 4, 6, 4, 6, 7, 0, 2, 1, 1, 2, 3, 2, 3, 4, 3, 4)
Letter N		(4, 4, 6, 5, 4, 5, 6, 7, 0, 2, 1, 2, 2, 3, 2, 3, 4, 4)
Letter S		(4, 4, 6, 4, 5, 6, 4, 4, 6, 7, 0, 1, 1, 1, 0, 2, 2, 2, 3, 4, 3, 4)
Letter E		(4, 5, 4, 5, 6, 4, 6, 6, 7, 0, 1, 0, 1, 2, 4, 1, 2, 3, 4, 3, 4)
Letter D		(6, 3, 4, 4, 6, 5, 6, 7, 6, 7, 0, 2, 1, 1, 2, 3, 3, 2, 4)
Letter R		(6, 4, 6, 4, 5, 6, 4, 6, 7, 0, 1, 2, 3, 2, 3, 2, 3, 4)

2.4 K-NN Algorithm Implementation

The K-Nearest Neighbors (K-NN) method is utilized in image classification, relying on feature extraction results during the training stage. The classification process focuses on determining the class label of new images based on their similarity to images in the training dataset. KNN can assist the chain code algorithm in translating sign language hand gestures, where the frequency of chain codes that have been affected by similar or ambiguous letters from the Indonesian language sign system (SIBI) by conducting several tests for one letter will be used as a data sheet and will later set by the KNN algorithm itself.

2.5 Evaluation using the Confusion Matrix

Evaluation of the KNN algorithm using a confusion matrix, aiming to map the algorithm's performance in tabular form. The confusion matrix shows the relationship between whether or not the data is categorized correctly. The confusion matrix consists of True Positive (TP), False Positive (FP), False Negative (FN), and True Negative (TN). True positive represents data in the positive class correctly predicted by the algorithm. False Positive represents data that should be in the positive class and expected to be in the negative class by the algorithm. False Negative is data that should be in the negative class, predicted to be the positive class by the algorithm. True Negative is data in the hostile class and correctly predicted by the algorithm.

Four evaluation method parameters are used to measure the performance of the K-NN algorithm: accuracy, precision, recall, and F1-Score. Accuracy describes the percentage of data records correctly classified by the system using Equation (1).

$$Accuracy = \frac{Number\ of\ Correct}{Total\ Inputs} \times 100\% \quad (1)$$

Precision describes the percentage of accuracy between the requested data and the K-NN classification result using Equation (2).

$$Precision = \frac{TP}{TP+FP} \quad (2)$$

True Positive Rate (TPR) describes the percentage of accuracy between the requested data and the K-NN result using Equation (3).

$$True\ Positive\ Rate\ (TPR) = \frac{TP}{TP+FN} \quad (3)$$

False Negative Rate (FNR) is the probability that a true positive will be missed by the test, using Equation (4).

$$False\ Negative\ Rate\ (FNR) = \frac{FN}{TP+FN} \quad (4)$$

2.6 Analysis and Interpretation

An analysis is required by calculating the accuracy of sign language hand gesture detection based on the letter variations of the Indonesian Sign Language System (SIBI), namely the letters "D," "E," "M," "N," "R," "S," and "U". This analysis is carried out to see the computation time and memory usage in sign language hand gesture recognition.

3. RESULT AND DISCUSSION

The data used in this study were taken from the camera in real-time, with the help of the Scikit-learn library. The data used were 350 samples, divided into 245 training data samples and 105 testing data samples. The testing phase was carried out to determine the accuracy of the method used in this study, namely the chain code algorithm, which functions to detect the results of the edge detection. The results of this chain code detector will later be used for K Nearest Neighbor (K-NN) as training data and test data. Training data on variations in k values using the k-fold cross-validation method. In this research, five folds were used. The following Table 3 shows the results of training k values with k-fold cross-validation.

3.1 Accuracy Results of Training Data from K-NN Algorithm

Table 4. Classification Results of Training Data Using K-NN Classifier

Number of neighbors (k)	Accuracy (%)	Precision (%)	TPR %	FNR %
1	85.47	88.15	85.44	14.51
2	86.54	81.69	86.34	13.45
3	84.96	85.5	84.76	15.03
4	88.76	83.54	88.60	11.23
5	87.45	88.12	87.53	12.54

Table 4 shows the number of errors, the percentage of errors, and the percentage of accuracy against the value of k. It can be seen in the table that when the value of k = 4, the highest rate of accuracy is 88.76% and the smallest percentage of error is 11.23%. Evaluation of this training can be seen using a valuable confusion matrix for determining accuracy, precision, TPR (recall), and FNR. To determine the best k value, one must consider precision, recall, TPR, and

FNR. Based on the confusion matrix table, it can be seen the percentage of True Positive Rate (TPR) where the highest when the value of k is 4 is 88.60%, and the False Negative Rate (FNR) is at the smallest value of 11.23%, and the Precision highest obtained when k shows 1 is 88.15%. In Figure 3, the confusion matrix test results are displayed.

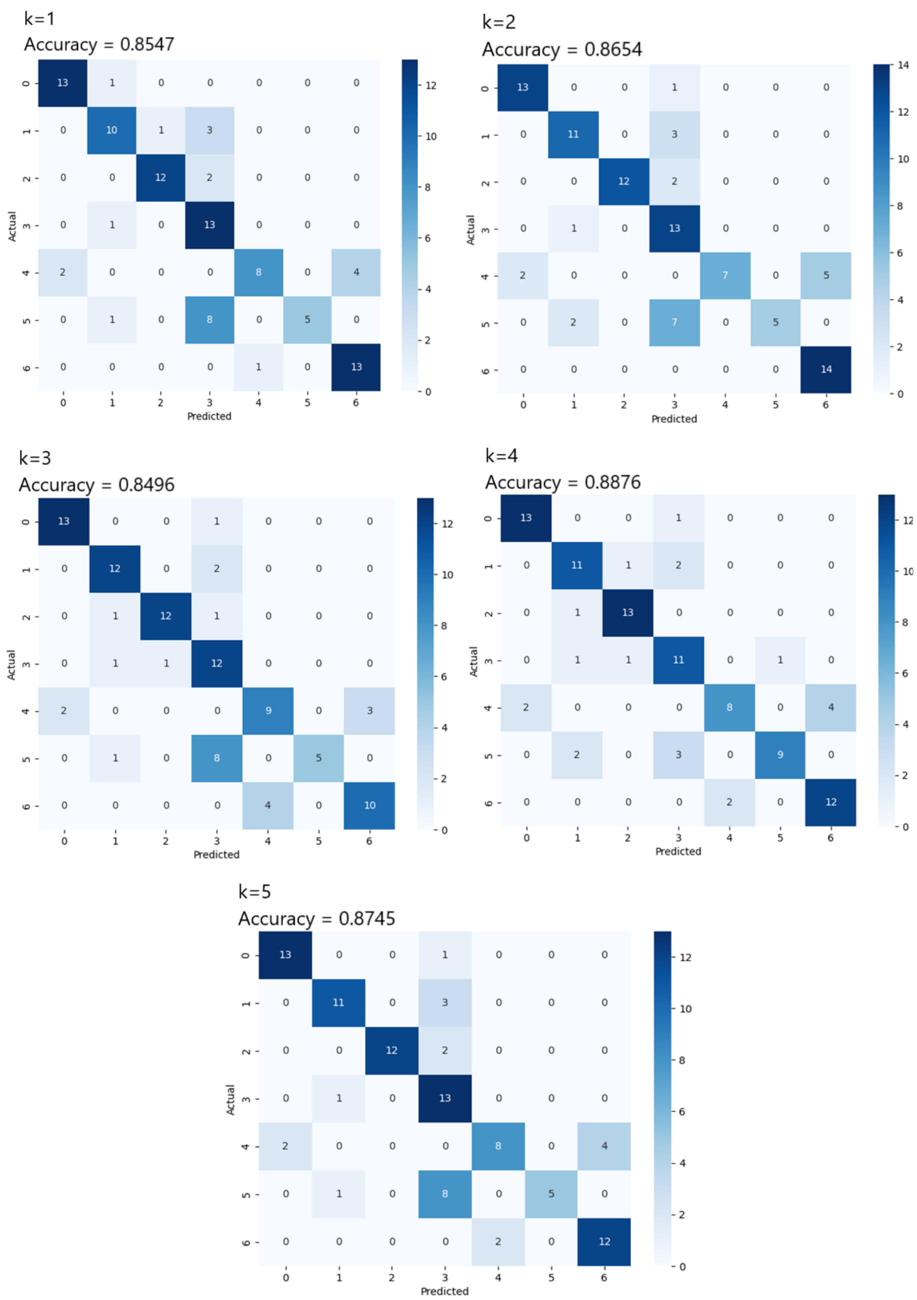


Figure 3. The Results of Training Data Confusion Matrix

3.2 K-Nearest Neighbor Model Testing

This test helps evaluate performance and whether this K-NN model can determine letters that have not been trained before. This test can also determine which k value is the best, as shown in Table 5 for test results based on the K value.

Table 5. Classification Results of Testing Data Using K-NN Classifier

Number of neighbors (k)	Accuracy (%)	Precision (%)	TPR %	FNR %
1	87.1	87.88	87.65	12.9
2	87.1	87.88	87.65	12.9
3	87.1	87.75	87.65	12.9
4	90.1	89.88	89.78	9.9
5	89.04	89.45	88.98	10.96

The table above shows that when k shows a value of 4, the accuracy percentage is 90.1%, with an error percentage of 9.9%. To further prove the best k value, we can evaluate the precision, True Positive Rate (TPR), and False Negative Rate (FNR). Based on the TPR, FNR, accuracy, and precision results, it can be seen that when k is 5, the percentage of the results is more than the other k. It can be concluded that five are taken in this research. In Figure 4, the confusion matrix test results are displayed.

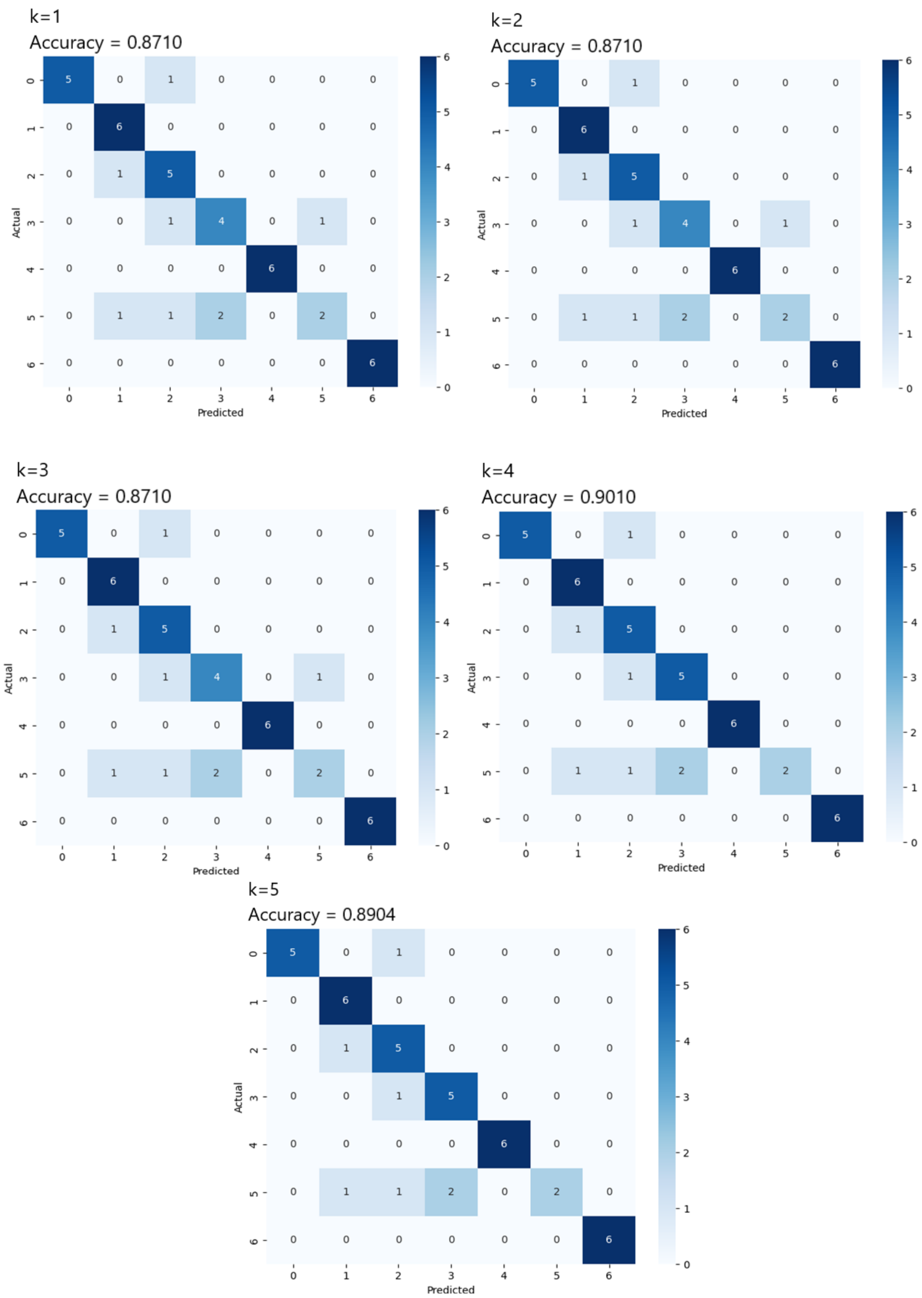


Figure 4. The Results of Testing Data Confusion Matrix

3.3 System Testing Results

The results of system testing in this study were carried out by training KNN by inputting 245 training data that will be learned by KNN, which consists of real-time images of ambiguous letters of sign language "D," "E," "M," "N," "R," "S," and "U". After training, the next researcher prepares 105 test data on real-time images of ambiguous letters in sign language where each letter has 15 data to be tested. Here, researchers use a value of k equal to 5. This test is carried out to determine the accuracy, memory usage, and computing time needed to process the data.

The accuracy of the recognition results of the letters "D," "E," "M," "R," and "U" is 100%. It can be concluded that the use of the K-NN method on this letter successfully recognises the letter pattern. While in the recognition of the letters ‘N’ and ‘S’, the results reached 98.88%. In the pattern recognition of these two letters, there is one error because one of the samples has a frequency similar to other letters. This similarity is also influenced by the results of the chain code on the similarity of the average hand. Because of this similarity, the K-NN algorithm becomes complicated in getting the shape of the pattern of this letter, as shown in Table 6.

Table 6. Results of The Sample Testing

Alphabet Letter	Sample Detected	Total Samples	Accuracy (%)
D	15	15	100
E	15	15	100
M	15	15	100
N	14	15	98.88
R	15	15	100
S	14	15	98.88
U	15	15	100

The accuracy of testing the letters "D," "E," "M," "R," and "U" has a percentage of 100%. It can be concluded that the use of the K-NN method proves that K-NN can perform pattern recognition on these letters. While testing the letters "N" and "S," the accuracy presentation results reached 98.88%. However, there is one error in the recognition of the letters N and S. This is certainly due to one of these samples having a frequency similar to other letters. This similarity is also influenced by the identical chain code results on average hand, because of this similarity, KNN has difficulty getting the shape of the pattern of this letter. The results of the comparison of the accuracy of sign language hand gestures on ambiguous letters can be seen in Figure 5.

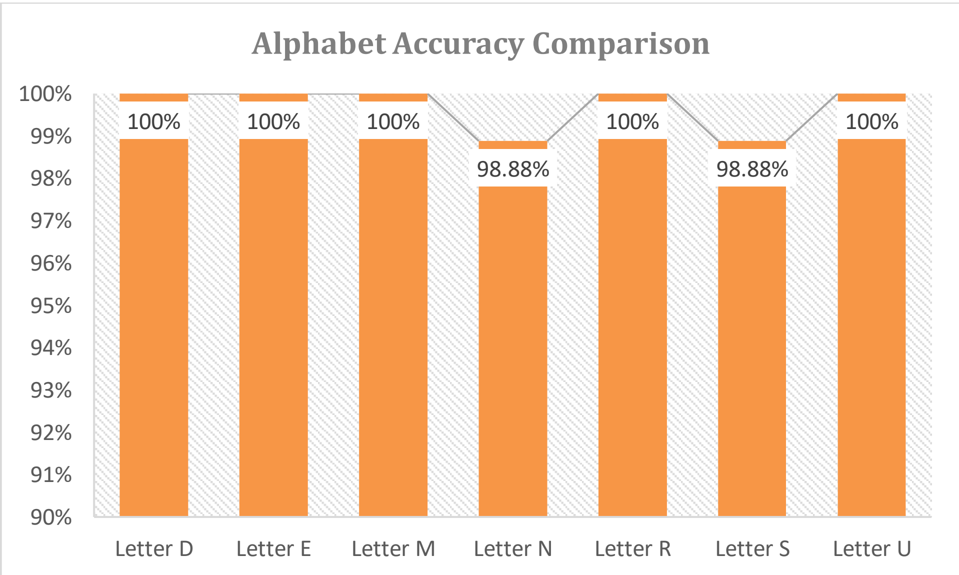


Figure 5. Comparison of Ambiguous Letter Accuracy

3.4 Discussions

Based on the research results, the highest k value from k = 1 to k = 5, the highest accuracy is k = 4, which is 90.1%. With an average k value reaching 87%, this is certainly very good, because it proves that K-NN has succeeded in recognizing patterns of ambiguous letters in the Indonesian sign language system (SIBI). The factors that cause accuracy not to be optimal are influenced by errors in the image preprocessing stage. In addition, errors also occur due to shadow noise factors, different light intensities, objects that reflect colors so that they look dark or light, and hand movements that can be more shaped. In this research, we used Canny edge detection, a technique that produces hand movement lines. The shape of the Indonesian sign language system (SIBI) discussed here can be clearer, with letters such as "D," "E," "M," "N," "R," "S," and "U" that are similar and unclear. When using real-time images, the edge detection results may be less than perfect due to the rough contours of the objects and the edge pixels that change during color segmentation. The results of this hand gesture edge detection will be searched for the starting point to get the results of the chain code, whose cardinal direction is searched counterclockwise. Then the output is obtained in the form of the number of directions or frequencies of the hand gesture series code. This frequency will be used as a dataset, which is divided into test data and training data, where K-NN will learn from the training data.

4. CONCLUSION

Based on the results and discussions that have been carried out related to the recognition of ambiguous letters in sign language using chain code and K-NN algorithm, the algorithm used can recognise the extraction of chain code features in the form of frequency against test data. The real-time dataset of 350 data is divided into two datasets, namely 70% training data and 30% test data. The highest accuracy result on training data reached 88.76%, while on test data the highest accuracy reached 90.1%. In this study, the letters "D," "E," "M," "R," and "U" have 100% accuracy. While the letters "N" and "S" have a slightly lower accuracy of 98.88%. This is because the contour of the letter is very similar to the tested letter.

The author realises that the analysis carried out by researchers still has many shortcomings. Therefore, there are several things that can be considered to develop this research to be even better in the future, namely: 1) Increase the amount of training data used in the classification process. 2) It is recommended to use different machine learning algorithms to classify ambiguous letters in sign language. 3) Future research is expected to use animated images for maximum results.

5. ACKNOWLEDGMENTS

Acknowledgments are given to the DRTPM KEMENRISTEKDIKTI which has funded this research and Lembaga Penelitian dan Pengabdian Masyarakat Institut Teknologi Bisnis Riau (ITBR) which has facilitated so that this research activity can be carried out. (112/E5/PG.02.00.PL/2024, 028/LL10/PG. AK/2024).

REFERENCES

[1] F. Fitriyani, L. Q. Ainii, R. Jannah, and S. Maryam, “Analysis of Sign Language Skills in Improving Communication and Learning for Deaf Children,” *Continous Educ. J. Sci. Res.*, vol. 5, no. 1, pp. 30–39, 2024, doi: 10.51178/ce.v5i1.1757.

[2] K. Wangchuk, P. Riyamongkol, and R. Waranusast, “Real-time Bhutanese Sign Language digits recognition system using Convolutional Neural Network,” *ICT Express*, vol. 7, no. 2, pp. 215–220, 2021, doi: 10.1016/j.icte.2020.08.002.

[3] M. J. Cheok, Z. Omar, and M. H. Jaward, “A review of hand gesture and sign language recognition techniques,” *Int. J. Mach. Learn. Cybern.*, vol. 10, no. 1, pp. 131–153, 2019,

doi: 10.1007/s13042-017-0705-5.

[4] S. Maullasari, "The role of social support to increase the Confidence of the Deaf in the Difabel Study Council," *J. Adv. Guid. Couns.*, vol. 3, no. 2, pp. 94–105, 2022, doi: 10.21580/jagc.0.0.0.5896.

[5] A. Harditya, "Indonesian Sign Language (BISINDO) As Means to Visualize Basic Graphic Shapes Using Teachable Machine," in *Proceedings of the International Conference of Innovation in Media and Visual Design (IMDES 2020)*, 2020, pp. 1–7. doi: 10.2991/assehr.k.201202.045.

[6] S. Syamsuddin, T. Pristiwaluyo, W. A. Saleh, and Z. Zulfitrah, "Development of Indonesian Sign Language System (SIBI) Dictionary Application for Students with Special Needs," *Al-Ishlah J. Pendidik.*, vol. 15, no. 3, pp. 3132–3143, 2023, doi: 10.35445/alishlah.v15i3.3529.

[7] G. Latif, J. Alghazo, N. Mohammad, and R. Alghazo, "Communicating with the Deaf and Hard of Hearing through Automatic Arabic Sign Language Translator," in *Journal of Physics: Conference Series*, 2021. doi: 10.1088/1742-6596/1962/1/012055.

[8] S. Shivashankara and S. Srinath, "American Sign Language Recognition System: An Optimal Approach," *Int. J. Image, Graph. Signal Process.*, vol. 10, no. 8, pp. 18–30, 2018, doi: 10.5815/ijigsp.2018.08.03.

[9] N. B. Ibrahim, M. M. Selim, and H. H. Zayed, "An Automatic Arabic Sign Language Recognition System (ArSLRS)," *J. King Saud Univ. - Comput. Inf. Sci.*, vol. 30, no. 4, pp. 470–477, 2018, doi: 10.1016/j.jksuci.2017.09.007.

[10] I. D. M. B. A. Darmawan *et al.*, "Advancing Total Communication in SIBI: A Proposed Conceptual Framework for Sign Language Translation," in *2023 International Conference on Smart-Green Technology in Electrical and Information Systems (ICSGTEIS)*, 2023, pp. 23–28. doi: 10.1109/ICSGTEIS60500.2023.10424020.

[11] A. Aljabar and Suharjito, "BISINDO (Bahasa isyarat indonesia) sign language recognition using CNN and LSTM," in *Advances in Science, Technology and Engineering Systems*, 2020, pp. 282–287. doi: 10.25046/AJ050535.

[12] A. D. Saputra, J. Jayanta, and A. B. Pangaribuan, "Klasifikasi Alfabet Bahasa Isyarat Indonesia (BISINDO) Dengan Metode Template Matching dan K-Nearest Neighbors (KNN)," in *Seminar Nasional Mahasiswa Ilmu Komputer dan Aplikasinya (SENAMIKA)*, 2020.

[13] R. K. Pathan, M. Biswas, S. Yasmin, M. U. Khandaker, M. Salman, and A. A. F. Youssef, "Sign language recognition using the fusion of image and hand landmarks through multi-headed convolutional neural network," *Sci. Rep.*, vol. 13, no. 1, pp. 1–11, 2023, doi: 10.1038/s41598-023-43852-x.

[14] P. W. Aditama, P. S. U. Putra, I. M. M. Yusa, and I. N. T. A. Putra, "Designing augmented reality sibi sign language as a learning media," in *Journal of Physics: Conference Series*, 2021. doi: 10.1088/1742-6596/1810/1/012038.

[15] K. Bantupalli and Y. Xie, "American Sign Language Recognition using Deep Learning and Computer Vision," in *Proceedings - 2018 IEEE International Conference on Big Data, Big Data 2018*, IEEE, 2018, pp. 4896–4899. doi: 10.1109/BigData.2018.8622141.

[16] A. Ridwan, Y. Purbolingga, and H. Hanisah, "Utilizing Convolutional Neural Network for Learning Web-Based Braille Letter Classification System," *J. Comput. Networks, Archit. High Perform. Comput.*, vol. 6, no. 1, Jan. 2024, doi: 10.47709/cnahpc.v6i1.3386.

[17] P. Yugopuspito, I. M. Murwantara, and J. Sean, "Mobile Sign Language Recognition for

Bahasa Indonesia using Convolutional Neural Network," in *Proceedings of the 16th International Conference on Advances in Mobile Computing and Multimedia*, in MoMM2018. New York, NY, USA: Association for Computing Machinery, 2018, pp. 84–91. doi: 10.1145/3282353.3282356.

[18] R. Ridwang, S. Syafaruddin, A. A. Ilham, and I. Nurtanio, "Indonesian Sign Language Letter Interpreter Application Using Leap Motion Control based on Naïve Bayes Classifier," in *IOP Conference Series: Materials Science and Engineering*, 2019. doi: 10.1088/1757-899X/676/1/012012.

[19] A. Nur and H. Nugroho, "Deteksi Fitur Huruf Sistem Isyarat Bahasa Indonesia menggunakan Metode Chain Code," *POSITIF J. Sist. dan Teknol. Inf.*, vol. 8, no. 1, pp. 36–40, 2022, doi: 10.31961/positif.v8i1.1133.

[20] H. M. Putri, F. Fadlisyah, and W. Fuadi, "Pendeteksian Bahasa Isyarat Indonesia Secara Real-Time Menggunakan Long Short-Term Memory (LSTM)," *J. Teknol. Terap. Sains 4.0*, vol. 3, no. 1, p. 663, 2022, doi: 10.29103/tts.v3i1.6853.

[21] I. Suyudi, S. Sudadio, and S. Suherman, "Pengenalan Bahasa Isyarat Indonesia menggunakan Mediapipe dengan Model Random Forest dan Multinomial Logistic Regression," *J. Ilmu Siber dan Teknol. Digit.*, vol. 1, no. 1, pp. 65–80, 2023, doi: 10.35912/jisted.v1i1.1899.

[22] N. A. Hasma, F. Arnia, R. Muharar, and M. K. Muchamad, "Pengenalan Gerakan Isyarat Bahasa Indonesia Menggunakan Algoritma Surf Dan K-Nearest Neighbor," *J. Komputer, Inf. Teknol. dan Elektro*, vol. 7, no. 1, p. 2022, 2022, [Online]. Available: https://jurnal.unsyiah.ac.id/kitektro/article/view/23262

[23] M. E. Al Rivan, H. Irsyad, K. Kevin, and A. T. Narta, "Implementasi LDA pada fitur HOG untuk Klasifikasi ASL Menggunakan K-NN," *JATISI (Jurnal Tek. Inform. dan Sist. Informasi)*, vol. 7, no. 2, pp. 214–225, 2020, doi: 10.35957/jatisi.v7i2.286.

[24] S. V Sonekar, H. Dhoke, V. Mate, S. Dhewle, and M. Patil, "Real-Time Sign Language Identification using KNN: A Machine Learning Approach," in *2023 11th International Conference on Emerging Trends in Engineering & Technology - Signal and Information Processing (ICETET - SIP)*, 2023, pp. 1–4. doi: 10.1109/ICETET-SIP58143.2023.10151523.

[25] A. Sultan, W. Makram, M. Kayed, and A. A. Ali, "Sign language identification and recognition : A comparative study," *Open Comput. Sci.*, vol. 12, no. 1, pp. 191–210, 2022.