International Journal on Advanced Science Engineering Information Technology

Vol.7 (2017) No. 5
ISSN: 2088-5334

Recognition of Bisindo Alphabets Based on Chain Code Contour and Similarity of Euclidean Distance

Dolly Indra[#], Sarifuddin Madenda[*], Eri Prasetyo Wibowo[*]

[#]*Faculty of Computer Science, University of Moslem Indonesia, Makassar, 90121, Indonesia*
E-mail: dolly.indra@umi.ac.id

[*]*Faculty of Computer Science, Gunadarma University, Jakarta, 16424, Indonesia*
E-mail: sarif@staff.gunadarma.ac.id, eri@staff.gunadarma.ac.id

Abstract — **In Indonesia, there are two forms of sign language practiced in the community, i.e., Indonesian sign language or known as BISINDO, and Indonesian sign language system or known as SIBI. In this study, we conduct research about recognition of Bisindo alphabets using contour chain code for the method of feature extraction and similarity of Euclidean distance for the method of recognition. The features used are the probability of chain code generated from contour following and the formation of chain code. The proposed method in this study consisted of five section, i.e., input test image, segmentation, edge detection, feature extraction and matching process of the alphabet. In the testing of the proposed method, we used 52 images of hand gestures used as test images. The images are in the form of static images and 26 images of hand gestures used as reference images which represent 26 alphabets BISINDO from A to Z, where the images stored in the database. The test images of different shapes and sizes with image references. For recognition, we do the matching between the probability of the test image chain code with the probability of the reference image chain code using Euclidean distance. The measurement result of Euclidean distance in this study was generated average accuracy rate of similarity above 94%. This indicates that the method proposed in this study was effective and produce the level of similarity of BISINDO alphabets was accurate**

Keywords—**BISINDO; segmentation; morphology; edge detection; contour following; chain code; Euclidean distance**

I. INTRODUCTION

In Indonesia, hearing, and speech impaired people communicate using sign language. The sign language used in this study, i.e., Indonesia sign language (BISINDO). BISINDO was developed by deaf people themselves through Gerkatin (Indonesian Deaf Welfare Movement) whilst SIBI was developed by normal individuals, instead of deaf speakers [1].

On the BISINDO alphabets to represent alphabets A to Z based on the drawing hand gestures [2]. In BISINDO alphabets there are some alphabets that are formed using one hand i.e., C, E, I, J, L, O, R, U, V, Z and some alphabets that are formed using two hands i.e., A, B, D, F, G, H, K, M, N, P,Q, S, T, W, X , Y. Pattern identification result of BISINDO alphabets is not uniform or heterogeneous. Some scientific studies about the identification of hand shape that representing a letter [3], [4], [5], [6].

In this study, the image used in the database is as much as 26 images, i.e., BISINDO alphabets A to Z which are used as reference images while the test images used as much as 52 images in which the images of different shapes or sizes with images in the database or reference image. The test images used are static images.

In this study, we use feature extraction method using contour following and formation of chain code to identify patterns in BISINDO alphabets which they are not uniform or heterogeneous. Some studies by using chain code [7], [8], [9], [10]. This is because the BISINDO alphabets can be formed with one hand and two hands. The object of hand image size sometimes smaller or bigger. The size change caused the different length of the chain code even though the shape of the BISINDO alphabets were still similar to solve the problems then we do the probability calculation of each chain code. Features identification results in features extraction process used in the next stage, i.e., recognition process. The recognition process that we are using the distance calculation.

In this study, the distance calculation between two images based on the Euclidean distance which aim to obtain the level of similarity between the probability values from chain code of test image with the probability values from chain code of the reference image found in the database. The process was conducted to determine the percentage of similarity probability between test image with reference image contained in the database. Some studies by using Euclidean distance [11], [12], [13]. This became the philosophical foundation of the method we proposed in our study which aims to obtain a feature extraction method was very effective so that it can produce the recognition of BISINDO alphabets with similarity level was accurate.

II. MATERIAL AND METHOD

In this section, we propose a recognition system for BISINDO alphabets. In the method, there are 5 processes, as shown in Fig. 1, i.e.

- Input test image
- Segmentation
- Edge detection
- Feature extraction
- Matching process of the alphabet.

The result of this process in the form of alphabet display from test image and the reference image in database and similarity of probability in the form of a percentage value.

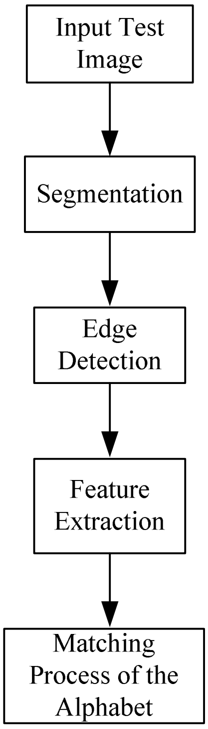


Fig. 1 Diagram of the method for recognition system of BISINDO alphabets

A. *Test Images*

The test image is the image that will be done testing with a reference image contained in the database. In this study, test image used as much as 52 images with the PNG format or Portable Network Graphics where the test images are named “bisindo_n. png”. The test images are made different from reference image contained in the database, in the form of shapes or sizes of different with the reference image. The test images can be smaller or larger than the reference image.

B. *Segmentation*

This process aims to represent an area in an image into a number of segments that have a meaning, and it is easier to be analyzed in the next process. In this segmentation process consists of three stages:

- Segmentation of color image
- Conversion of image
- Morphology process

The first stage, namely the process of color image segmentation. This process is done by using the algorithm of color distance measurement and the threshold value which aims to determine the similarity of skin color with pixels color contained in an image [14], [15], [16]. Referring to the image used in this process in the form of RGB color image, then the distance formula used in this study is city block distance or Manhattan distance, given by equation (1) and (2).

$$\Delta E = \sqrt{(R_1 - R_2)^2 + (G_1 - G_2)^2 + (B_1 - B_2)^2} \quad (1)$$

$$\Delta E = |R_1 - R_2| + |G_1 - G_2| + |B_1 - B_2| \quad (2)$$

Where R_1, G_1, B_1 respectively are the color component of red, green and blue from skin color and R_2, G_2, B_2 respectively are the color component of red, green and blue from the color of each pixels P in the image. If the Value $\Delta E > Th$ then the pixel P is part of the object of interest to be searched.

The second stage, i.e., the conversion of images, where the first image conversion performs the conversion of the image from the result of color image segmentation to the grayscale image and then the second image conversion perform the conversion of the grayscale image to a binary image.

The final stage of the process is a morphological closing. The basic operation of mathematical morphology, i.e., dilation and erosion. Dilation is a morphological process to expand the area or size of an object. Dilation image of f by the structure of element B is shown in Equation (3). Erosion is a morphological process to erode or reduce the width of the surface size of an area or object. The erosion image of f by the element structure of matrix B shown in the equation (4). Morphological closing is dilation operation on an image followed by erosion operation shown in equation (5).

$$f \oplus B = \{s | (B)_s \cap f \neq 0\} \quad (3)$$

$$f \ominus B = \{s | (B)_s \subseteq f\} \quad (4)$$

$$f \cdot B = (f \oplus B) \ominus B \quad (5)$$

C. *Edge Detection*

In the image processing, edge on an object provides a very significant role to perform analysis of an image. Visually, the edge of the object is the boundary between one object with background or another object.

This boundary can be seen through the intensity difference or color one pixel by pixel closest neighbors at the boundary. The process of edge detection simply uses the operator of Roberts.

D. *Feature Extraction*

The stages are performed in the feature extraction process there are two stages, i.e.,

- Contour following and Chain code.
- Calculate the probability of occurrence of each chain code.

The first stage in this section, i.e., contour following. It is a method to find out the start point of the object edge and then follow it until the end of the edge [17], [18], [19]. The following forward technique to determine the direction of contour movement is conducted using eight neighbor pixels.

In this study, all of the contour object represented by white color pixels with thickness contour of one pixel. Contour following algorithm is as follows:

- Check each pixel in each row and in a column by column. Either it was the part of the contour or not. If the start point of the contour was found, then the second stage was taken however if it was not found then the fourth stage was taken.
- Do the following forward trace by using eight neighbor pixels to determine the next contour pixel. Do it continuously to the last pixel from this contour.
- Return to the first stage to find out the other contour.
- The following completed.

Chain code in this study was used to represent the contour shape of objects using the movement direction code of the contour. Chain code was used to represent the feature contour of objects using the movement direction code of the contour. As mentioned above that the contour movement was searched with the approach direction of eight neighbor pixels [20], [21] as shown in Fig. 2. This method is known as chain code or Freeman code.

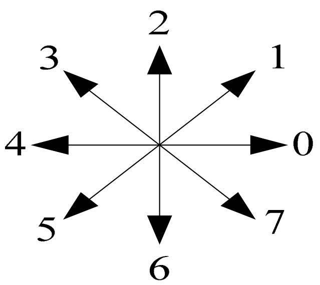


Fig. 2 Chain code

The final stage in the feature extraction process is counting the probability of occurrence of each code chain. The algorithm for calculating the probability of chain codes is as follows:

- Probability value initialization of chain code = 0.
- Initialization k value = 1 dan count = 0.
- Calculate the length of chain code that is Num = code chain. If Num < 8 do step 4 and if not do step 5.
- Probability of (k, Num + 1) = probability of (k, Num + 1) +1 and count value = count +1.
- The probability of (k, 1: 8) = probability of (k, 1: 8) / count. The value of count = 0, k = k + 1 and probability of (k, 1: 8) = 0.
- Calculate the probability value of (k, 1: 8) = probability of (k, 1: 8) / count.

E. *Matching Process of Alphabet*

This stage is used to BISINDO alphabets recognition where the probability of test image will be compared with the probability of reference image based on Euclidean distance. Euclidean distance is given in equation 6.

$$j(v_1, v_2) = \sqrt{\sum_{k=1}^{N}(v_1(k) - v_2(k))^2} \qquad (6)$$

v_1 and v_2 were a vector. In this case, v_1 is the probability of the chain code for the reference image, and v_2 is the probability of the chain code for the test image where the values of v_1 and v_2 will be calculated using Euclidean distance.

The algorithm for the BISINDO alphabet recognition is as follows:

- Read the test image.
- Perform calling a database, i.e., the reference image of BISINDO alphabets.
- Perform segmentation process of the test image.
- Perform the edge detection process of the test image.
- Perform a chain code extraction process of the test image.
- Declare the variables for percentage similarity and largest alphabets.
- Perform database looping as much as 26 times (reference image of A to Z).
- Perform matching between the probability of chain code for test image with the probability of chain code for reference image based on Euclidean distance.
- Display the test image and the reference image along with the similarity percentage.

The algorithm for calculating Euclidean distance is as follows:

- Change the shape of the matrix on the reference image i.e., A2=reshape(probA',prod(size(probA)),1).
- Change the shape of the matrix on the test image i.e., B2=reshape(probB',prod(size(probB)),1).
- Calculate the euclidean distance i.e., dist = sqrt(dot(A2-B2,A2-B2)).
- Calculate the accuracy rate i.e., accuracy = 1/(1+dist)*100.

The algorithm for matching similarity of the BISINDO alphabet is as follows:

- Find the matrix size for the probability of chain code for test image and the reference image.
- Perform equalization of the number of rows to the probability of the chain code for the test image and the probability of the chain code for the reference image in the database by adding a zero row.
- Perform permutation process as much as the number of rows.
- Perform the looping process as much as the permutation of rows.
- Perform computation the Euclidean distance.

III. Results and Discussion

Method and algorithm for segmentation process that we are proposed using color distance measurement algorithm and the threshold value, conversion to the grayscale image, conversion to the binary image and perform morphological closing process will be used in the next step i.e., edge detection to produce the image edge of BISINDO alphabet very clearly.

Fig. 3 shows an original image of the BISINDO alphabet that represents the form of the alphabet A. Fig. 4 result of segmentation from Fig. 3 by using the color distance measurement and threshold value i.e., city block distance or Manhattan distance. Fig. 5 is the shape of a grayscale image from the process of image conversion on Fig. 4.

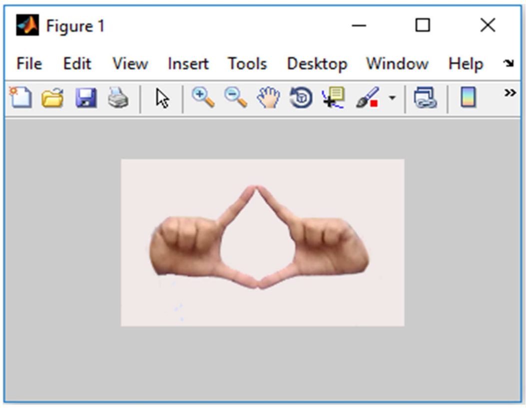


Fig. 3 Original image

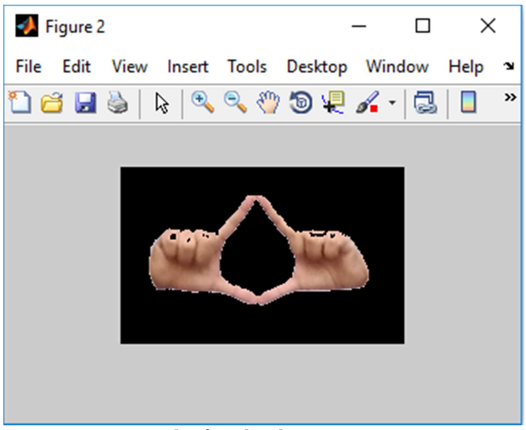


Fig. 4 Result of Color distance measurement

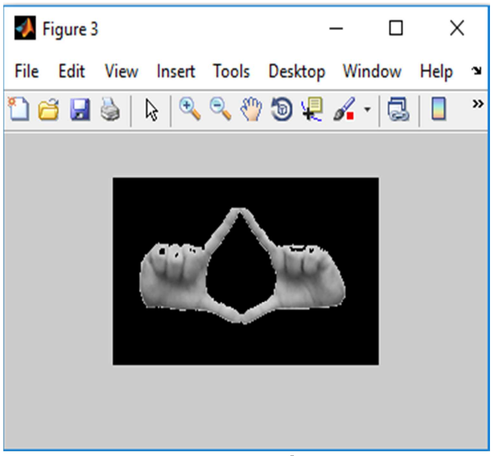


Fig. 5 Grayscale image

Fig. 6 shows the shape of the binary image which is conversion process of the grayscale image in Fig., where the image is still noise so that the next process is needed by performing morphology process i.e., closing operation. This process aims to cover the hole contained in an object shown in Fig. 7.

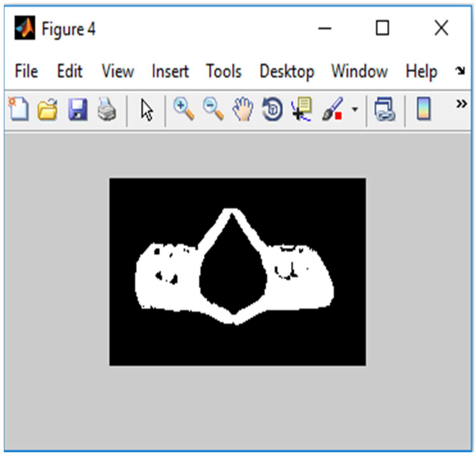


Fig. 6 Binary image

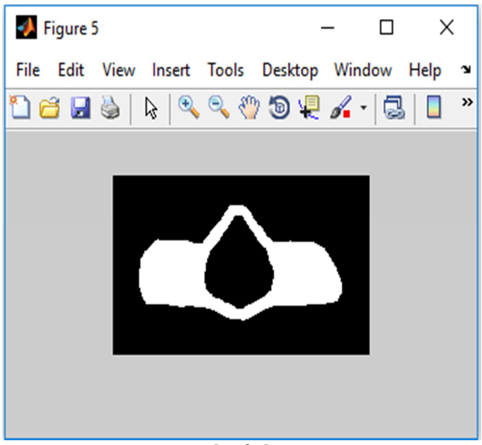


Fig. 7 Result of closing operation

The next process is the edge detection of the image shown in Fig. 8 simply using the Roberts operator. The result of the edge detection process is shown in Fig. 8.

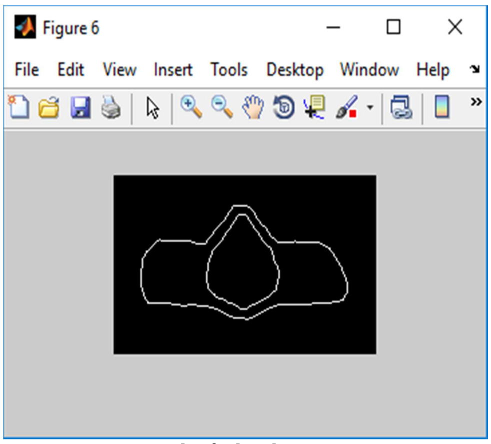
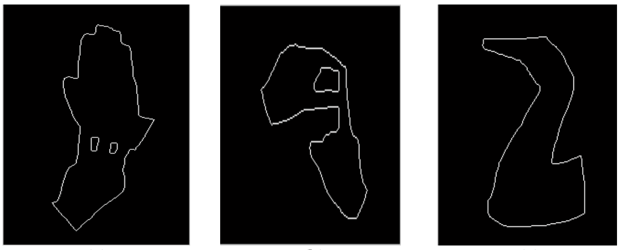


Fig. 8 Result of edge detection process

Examples for edge detection of the alphabets M, P and Z in the BISINDO alphabets are shown in Figs. 9(a), 9(b) and 9(c).



(a) (b) (c)

Fig. 9 Edge detection process for (a) Alphabet M, (b) Alphabet P and (c) Alphabet Z

The chain code of each contour following at the BISINDO alphabets M, P and Z are shown in Fig. 10 to Fig. 12 using Freeman code with direction approach for 8 neighboring pixels.

In this study, we have added a separator between one contour with another contour with number 9 in the chain code if the contours are found more than one contour and each contour has its own chain code by separated by the number 9 is red shown in Fig. 10 and Fig. 11.

On the alphabet M is shown in Fig. 10 has three contours so that the separator has as much as 2 pieces are represented by number 9 is red.

On the alphabet P is shown in Fig. 11 has two contours so that the separator has 1 pieces are represented by number 9 is red while alphabet Z is shown in Fig. 12 only has one contour just so there is no separator.

The probability of chain code of BISINDO alphabets contour for A to Z used as reference data in the database shown in Table 1 i.e., in column 0, 1, 2, 3, 4, 5, 6 and 7. They are showing the probability of each chain code 0 to 7 while column L is showing of alphabets A to Z.

The method and algorithm for feature extraction and recognition in BISINDO alphabet, i.e., using contour chain code and Euclidean distance in this study is good. The number of contours in this study visually can be seen from the results of edge detection on BISINDO alphabet image. In Table 2 is an example testing on test image with the reference image. The test images have unequal forms that are not exactly, and size may be smaller or larger than reference images.

For the first test image was the tested image with the name of bisindo_4. In this image has a smaller size than reference image i.e., 200x118 shown in Table 2 section (a) while reference image, i.e., 200x125 shown in Table 2 section (c) while in Table 2 section (b) and (d) are images result of edge detection from test image segmentation process in Table 2 section (a) and segmentation process of reference image in Table 2 section (b). The images have two closed contours that produce similarity of probability with reference image is 96.5273 and BISINDO alphabet is known as alphabet A.

For the second test image was tested image with the name of bisindo_5. In this image has a greater size than reference image i.e., 200x145 shown in Table 2 section (e) while reference image has size i.e., 200x131 shown in Table 2 section (g), while in Table 2 section (f) and (h) are the image result of edge detection from test image segmentation process in Table 2 section (e) and segmentation process of reference image in Table 2 section (g). The images have three closed contours that produce similarity of probability with reference image is 94.7661 and BISINDO alphabet is known as alphabet B.

```
07000070000770076700700007667666666
66676707070076766676666666666667666
66676666666666666676676676667666666
66667666666666666667666666666666770
70070000700007056565656565656565656
655656555656555556653444456566556565
66666666566656666766676666666666666
66656665655555555455554555545555555
55555655565555455555555565555553332333
23323323333332333233233323230100011
21221221222122222122212221221222212
21212121011011112222122221222222222
22222122222222221222222222223323233
23232322322223232232232232232322223
22222232222222221220101007007000001
222222222222222222222221222221222222
22222212222222121212101007000001212
22222121900000076666656665666666544
443222222222222222221900000766666665
65654444322221222222
```

Fig. 10 Chain code of BISINDO for alphabet M

```
07070000000070000000000000000000070707
00770700707770706707006766666666666666
6766666666676666667666667666667667676
70707766766666666666676666666766676676
67776775656665656556556556555444444444
43434334323343334333333333233232323232
32323323333333323222321221000000000
00012111000110000000111222222222222222
23444444444444544444444444444544444444
4444544545454454544544544444544454444
44453323233232322323223222323110111
11121121112112112112110011011010000
01900000000000707000000766666666666666
54444443444444445444444444433222221121
111122
```

Fig. 11 Chain code of BISINDO for alphabet P

For the third test image was tested image with the name of bisindo_17. In this image has a greater size than reference image i.e., 200x132 shown in Table 2 section (i) while reference image has size i.e., 200x119 shown in Table 2 section (k), while in Table 2 section (j) and (h) are image result of edge detection from test image segmentation process in Table 2 section (i) and segmentation process of reference image in Table 2 section (k). The images have one closed contour that produces similarity of probability with reference image is 95.7168 and BISINDO alphabet is known as alphabet E.

TABLE I

THE PROBABILITY EACH CODE CHAIN

L	**0**	**1**	**2**	**3**	**4**	**5**	**6**	**7**
A	0.218	0.094	0.085	0.047	0.331	0.047	0.066	0.113
	0.039	0.141	0.160	0.096	0.154	0.096	0.160	0.154
B	0.183	0.102	0.083	0.104	0.238	0.058	0.116	0.116
	0.165	0.175	0.041	0.01	0.381	0.021	0.134	0.072
	0.340	0.021	0.085	0.064	0.298	0.064	0.106	0.021
C	0.204	0.130	0.067	0.056	0.285	0.081	0.092	0.083
D	0.160	0.120	0.067	0.135	0.197	0.035	0.197	0.089
	0.147	0.118	0.112	0.100	0.188	0.041	0.241	0.053
E	0.351	0.043	0.076	0.042	0.325	0.072	0.049	0.042
F	0.255	0.068	0.083	0.092	0.257	0.067	0.090	0.088
G	0.161	0.084	0.177	0.064	0.187	0.070	0.185	0.072
H	0.119	0.107	0.171	0.083	0.154	0.072	0.211	0.082
I	0.171	0.050	0.267	0.043	0.105	0.128	0.185	0.050
J	0.134	0.107	0.212	0.058	0.11	0.141	0.175	0.063
K	0.072	0.084	0.186	0.124	0.139	0.079	0.134	0,182
L	0.175	0.059	0.229	0.050	0.149	0.104	0.172	0.064
M	0.092	0.07	0.300	0.077	0.014	0.131	0.256	0.060
	0.136	0.023	0.341	0.023	0.091	0.068	0.295	0.023
	0.152	0.03	0.273	0.030	0.121	0.091	0.273	0.030
N	0.068	0.109	0.271	0.062	0.047	0.113	0.285	0.045
	0.158	0.000	0.351	0.018	0.070	0.158	0.193	0.053
O	0.140	0.169	0.107	0.055	0.199	0.132	0.121	0.077
	0.167	0.091	0.121	0.106	0.182	0.091	0.152	0.091
P	0.173	0.090	0.118	0.118	0.173	0.059	0.180	0.088
	0.222	0.097	0.083	0.042	0.306	0.028	0.181	0.042
Q	0.170	0.115	0.136	0.085	0.157	0.134	0.111	0.091
	0.200	0.062	0.169	0.062	0.200	0.108	0.123	0.077
R	0.116	0.099	0.207	0.108	0.051	0.147	0.187	0.085
S	0.207	0.098	0.085	0.068	0.288	0.066	0.073	0.114
T	0.227	0.099	0.136	0.038	0.225	0.101	0.140	0.034
U	0.107	0.076	0.184	0.141	0.088	0.034	0.297	0.073
V	0.049	0.086	0.274	0.066	0.097	0.051	0.302	0.074
W	0.114	0.097	0.198	0.117	0.061	0.137	0.175	0.102
X	0.199	0.078	0.135	0.112	0.148	0.099	0.150	0.080
Y	0.096	0.139	0.195	0.047	0.139	0.137	0.162	0.085
Z	0.107	0.131	0.169	0.047	0.197	0.056	0.234	0.059

For the fourth test image was tested image with the name of bisindo_106. In this image has a smaller than reference image i.e., 200x250 shown in Table 2 section (m) while reference image has size i.e., 200x310 shown in Table 2 section (o), while in Table 2 section (n) and (p) are image result of edge detection from test image segmentation process in Table 2 section (m) and segmentation process of reference image in Table 2 section (p).The images have one closed contour that produces similarity of probability with reference image is 94.2409 and BISINDO alphabet is known as alphabet Z.

The test result of the test image conducted as many as 52 images with reference image contained in the database shown in Table 3.

The measurement result of chain code probability on the test image and probability of chain code on the reference image using Euclidean distance measurement shown in Table 3, i.e., column of accuracy rate.

000000007677777707706777777677676766
767767676776767666666666666665666566
566565665665656565656566566566565666
566656665656566656665666666566667000
000010000100010101001001010066766666
666766666667666667666666666666666666
666666666666666545454545445445454445
444454444454444454445444444444444444
454444444444444444444444444434443434
334343343342323222222222222211121221
221221222212222212222121221222211212
121212121121212121121212112112112111121
221212121211212112112112112112121222
223343323323323422333223333433234444
343434444344343444434343444344434433
232122221000000000000000000000000000
00000000000000000010000000000000

Fig. 12 Chain code of BISINDO for alphabet Z

TABLE II
SAMPLE TESTING BETWEEN TEST IMAGE AND REFERENCE IMAGE

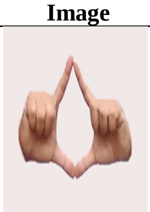
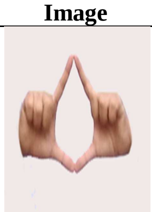
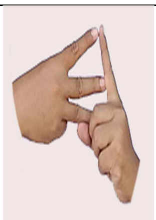
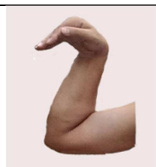
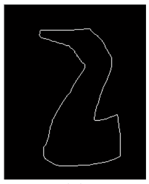
	Test Image	Reference Image	Alphabet	Accuracy Rate
1	 200x118 (a)  (b)	 200x125 (c)  (d)	A	96.5273
2	 200x143 (e)  (f)	 200x131 (g)  (h)	B	94.7661
3	 200x132 (i)  (j)	 200x119 (k)  (l)	E	95.7168
4	 200x250 (m)  (n)	 200x310 (o)  (p)	Z	94.2409

TABLE III
TESTING BETWEEN TEST IMAGE AND REFERENCE IMAGE

	Test Image	Test Image Size (Pixels)	Similarity of Reference Image	Reference Image Size (Pixels)	Accuracy Rate (%)
1	bisindo_2	200x126	A	200x125	96.6015
2	bisindo_4	200x118	A	200x125	96.5273
3	bisindo_5	200x143	B	200x131	94.7661
4	bisindo_7	196x140	B	200x131	94.1134
5	bisindo_9	196x169	C	200x147	94.9759
6	bisindo_11	196x166	C	200x147	93.3258
7	bisindo_13	200x144	D	200x161	96.2251
8	bisindo_15	200x157	D	200x161	96.4254
9	bisindo_17	200x132	E	200x119	95.7168
10	bisindo_19	200x104	E	200x119	95.7289
11	bisindo_21	200x195	F	200x206	97.1749
12	bisindo_23	200x208	F	200x206	94.2865
13	bisindo_25	200x170	G	200x190	97.5839
14	bisindo_28	200x188	G	200x190	94.8113
15	bisindo_29	200x172	H	200x203	96.6107
16	bisindo_33	196x194	H	200x203	93.1558
17	bisindo_35	200x259	I	200x221	93.3759
18	bisindo_37	200x231	I	200x221	94.3290
19	bisindo_38	286x325	J	200x169	93.7622
20	bisindo_41	200x184	J	200x169	92.9435
21	bisindo_42	200x216	K	200x206	97.6771
22	bisindo_45	196x201	K	200x206	97.7150
23	bisindo_46	146x180	L	196x188	95,2339

24	bisindo_48	404x410	L	196x188	95.8722
25	bisindo_52	196x293	M	200x301	91.5407
26	bisindo_54	296x448	M	200x301	94.8717
27	bisindo_55	196x267	N	200x354	92.2653
28	bisindo_56	196x266	N	200x354	92.3860
29	bisindo_59	196x172	O	196x170	93.8996
30	bisindo_60	196x172	O	196x170	92.0228
31	bisindo_65	246x296	P	196x171	94.3788
32	bisindo_66	196x170	P	196x171	93.6569
33	bisindo_67	200x157	Q	200x221	91.1089
34	bisindo_68	200x143	Q	200x221	90.4708
35	bisindo_71	392x488	R	200x284	95.9184
36	bisindo_74	196x230	R	200x284	96.7334
37	bisindo_75	200x155	S	200x149	97.4280
38	bisindo_77	200x186	S	200x149	96.3901
39	bisindo_80	200x139	T	200x147	96.6287
40	bisindo_82	200x179	T	200x147	95.8423
41	bisindo_83	200x177	U	200x205	97.7958
42	bisindo_84	200x266	U	200x205	94.7464
43	bisindo_87	196x268	V	196x217	95.0422
44	bisindo_90	200x255	V	196x217	95.4262
45	bisindo_91	438x304	W	200x141	96.6675
46	bisindo_94	200x147	W	200x141	95.6035
47	bisindo_96	200x121	X	200x139	92.3201
48	bisindo_97	200x164	X	200x139	97.5145
49	bisindo_99	200x268	Y	200x265	95.0551
50	bisindo_102	200x219	Y	200x265	91.1140
51	bisindo_105	200x236	Z	200x310	98.0474
52	bisindo_106	200x250	Z	200x310	94.2409

IV. CONCLUSIONS

Method for feature extraction and recognition that we used in this study to perform matching of similarity feature using Euclidean distance with probability of chain code as reference feature is very effective where method we used can read test image size have smaller or larger than reference image size with a shape that does not exactly match the image in the database. The test results for the test image as much as 52 images with the image in the database produce average accuracy rate above 94%.

NOMENCLATURE

ΔE City block distance or Manhattan distance
$\oplus$ Morphological dilation
$\ominus$ Morphological erosion
$\cdot$ Morphological Closing
B Element structure of matrix B
v_1, v_2 Vector
N Vector length
j euclidean distance

Greek letters
Δ delta
Σ sigma

Subscripts
Th Threshold
R_1,R_2 Red_1 and Red_2
G_1,G_2 $Green_1$ and $Green_2$
B_1,B_2 $Blue_1$ and $Blue_2$

REFERENCES

[1] N. Sugianto and F. Samopa, "Analisa manfaat dan penerimaan terhadap Implementasi Bahasa Isyarat Indonesia pada latar belakang komplek menggunakan Kinect dan Jaringan Syaraf Tiruan (Studi Kasus SLB Karya Mulia 1)", JUISI, Vol. 01, No. 01, Februari 2015.
[2] I. Kautsar , R.I. Borman and A. Sulistyawati, "Aplikasi Pembelajaran Bahasa Isyarat Bagi Penyandang Tuna Rungu Berbasis Android Dengan Metode Bisindo", Seminar Nasional Teknologi Informasi dan Multimedia 2015.
[3] B. Seddik, S. Gazzah and N. E. B. Amara," Modalities Combination for Italian Sign Language Extraction and Recognition", Springer International Publishing Switzerland 2015, pp. 710–721, 2015.
[4] M. Zadghorban and Manoochehr Nahvi, "An algorithm on sign words extraction and recognition of continuous Persian sign language based on motion and shape features of hands", Springer-Verlag London 2016.
[5] P.V.V. Kishore , M.V.D. Prasad, D.Anil Kumar and A.S.C.S.Sastry," Optical Flow Hand Tracking and Active Contour Hand Shape Features for Continuous Sign Language Recognition with Artificial Neural Networks", IEEE 6th International Conference on Advanced Computing, 2016.
[6] Nagendraswamy .H.S, Chethana Kumara .B.M and Lekha Chinmayi .R,*"Indian Sign Language Recognition: An Approach Based on Fuzzy-Symbolic Data", 2016 Intl. Conference on Advances in Computing,* IEEE, Communications and Informatics (ICACCI), Sept. 21-24, 2016.
[7] Aini Najwa Azmi and Dewi Nasien, "Feature Vector of Binary Image using Freeman Chain Code (FCC) Representationbased on Structural Classifier", Int. J. Advance Soft Comput. Appl., Vol. 6, No. 2, pp. 1-19, 2014
[8] Mohan Mahanty, D. Kiranmayi, P. Chandana,"Sketch Retrieval using Contour Detection by Chain Codes",International Journal of Advanced Research in Computer and Communication Engineering, Vol. 4, Issue 10, pp.210-213, 2015
[9] Zamen Fadhel Jabr, "Hand Palm Recognition using Combination of Freeman Chain Code and Texture Features with Fuzzy Logic Classifier",International Journal of Computer Science and Mobile Computing, IJCSMC, Vol. 4, Issue. 3, pg.585 – 598, 2015
[10] Mohamad M.A., Haron H., Hasan H, "Metaheuristic Optimization on Conventional Freeman Chain Code Extraction Algorithm for Handwritten Character Recognition", Asian Conference on Intelligent Information and Database Systems, ACIIDS 2017: Intelligent Information and Database Systems, pp 518-527, 2017
[11] Meenakshi Sharma and Anjali Batra, "Analysis of Distance Measures in Content Based Image Retrieval", Global Journal of Computer Science and Technology: G Interdisciplinary, Volume 14, Issue 2 Version 1.0 , 2014.
[12] Jasmine Irani, Nitin Pise, Madhura Patak, "Clustering Techniques and the Similarity Measures used in Clustering: A Survey", International Journal of Computer Applications (0975 – 8887) Volume 134 – No.7, 2016.
[13] Jasmine Irani, Nitin Pise, Madhura Phatak, "Clustering Techniques and the Similarity Measures used in Clustering: A Survey",International Journal of Computer Applications (0975 – 8887), Volume 134 – No.7, pp. 9-14, 2016
[14] Madenda, S., "Pengolahan Citra & Video Digital", Jakarta : Erlangga, 2015.
[15] R. Harrabi and E. B. Braiek, "Color Image Segmentation by Multilevel Thresholding using a Two Stage Optimization Approach and Fusion", International Journal of Engineering and Innovative Technology (IJEIT) Volume 3, Issue 11, May 2014.
[16] H.K. Singh, S. K. Tomar and P. K. Maurya," Thresholding Techniques applied for Segmentation of RGB and multispectral images", Proceedings published by International Journal of Computer Applications, MPGI National Multi Conference (MPGINMC-2012), 2012.
[17] W. Bi, Y. Zhang, W. Huang and G. Gao, "Salient Contour Matching for Object Detection", 2016 8th International Conference on Intelligent Human-Machine Systems and Cybernetics, IEEE, 2016
[18] S.K. Verma, G. Kaur and A. Kumar," Entropy Based ROI Extraction and Modified Contour Model for Image Segmentation", 2016 6th International Conference on Advanced Computing (IACC), IEEE, 2016.
[19] P.V.V.Kishore, M.V.D. Prasad, D.Anil Kumar, A.S.C.S.Sastry," Optical Flow Hand Tracking and Active Contour Hand Shape Features for Continuous Sign Language Recognition with Artificial

Neural Networks", IEEE 6th International Conference on Advanced Computing, 2016.
[20] R S Vaddi, L N P Boggavarapu, H D Vankayalapati and K. R. Anne," Contour Detection Using Freeman Chain Code And Approximation Methods For The Real Time Object Detection", Asian Journal Of Computer Science And Information Technology1:1, pp.15 – 17, 2011.
[21] W.Shahab, H. Al-Otum, and F. Al-Ghoul," A Modified 2D Chain Code Algorithm for Object Segmentation and Contour Tracing",The International Arab Journal of Information Technology, Vol. 6, No. 3, July 2009.