

HANDWRITING RECOGNITION USING THE HIDDEN MARKOV MODEL METHOD

Daniel¹, Irfan Maliki²

^{1,2}Universities Komputer Indonesia

Jalan Dipatiukur No. 112 Bandung, Jawa Barat 40132

E-mail : sembiringdaniel1@gmail.com¹, irfanmaliki007@gmail.com²

ABSTRACT

Current technology developments utilize the computer growing on almost all fronts. Human beings increasingly dependent on making every effort to do Computer storage or pengolhan information. The processing of information that is currently being developed is about handwriting recognition, the introduction of writing is currently a lot of digunakan like identikasi important documents and others. The process of initial data preprocessing system is on training and test, then performed with characteristic extraction method of DEF (Directional Elment Featured). DEF one of the characteristic extraction method is proven to deliver the best results in handwriting recognition. As an example is never applied to Chinese characters based on the contour, and then combined with a method of HMM (Hidden Markov models) digunakan as a method of classification to handwriting recognition with DEF method (use their Directional element featured), from handwriting recognition results on research training data is owned as many as 720 data consist of letters A-Z sebnayak 520 each letter represents 20 data, and the number 0-9 has 200 record with each number consists of 20 data. From the training data digunakan 20% data for testing the introduction of writing 100% existing data, after doing the testing accuracy of owned.

Kata kunci : Handwriting Recognition, Directional Element Feature (DEF),Hidden Markov Model (HMM).

1. INTRODUCTION

Current technology developments utilize the computer growing on almost all fronts. Humans are increasingly dependent on the ability of a computer to perform information processing or storage, and is developing at this time i.e. the ability of a computer that can recognize an introduction to writing. Research in the field of handwriting recognition is widely used because it is in everyday life, such as the identification of important documents, such as proof of endorsement are concerned with, in the world of banking, and the other involves handwriting in it [1]. Handwriting each person has different shapes and patterns. The pattern is a defined entity through traits

(featured). The characteristics that distinguish a pattern. Pattern recognition (pattern recognition) can be defined as the process of classification. Two things are very good influence on whether or not the introduction of handwriting patterns is a feature of extraction and classification. Based on handwriting recognition for the study has been done on process constraints recognized result is found, this is because some of the letters of the alphabet have a semblance of form with other letters [2], as well as on its application introduction writing new hand implemented in the case of research in General. Handwriting recognition will be influenced by the result accuracy when you want to generate handwriting characteristics and effective method of classification that can recognize handwriting image characteristics well into the process of handwriting recognition.

This will be used in the research method of Directional Element Feature (DEF) as a feature of the extraction method. Directional Element Feature (DEF) is one of the characteristic extraction method has been used in the title of the Java script handwriting recognition use their Directional Element Feature and Multi Class Support Vector Mechine obtained 93.6%, result [3]. Then the classification method used is the method of hidden markov model (HMM), HMM had been used in previous studies especially pattern recognition letters Alif berharakat use their Modified-DFE and HMM and obtained results 69.4 [4]. The average accuracy of the result is the best introduction to research before Implementation method of Hidden Markov models for Handwriting detection accuracy about 74.72% [5]. HMM have diimplementasikan on several characters using image online as well as offline in the process of its classification. HMM is an extension of markov chains, where statenya cannot be directly observed (hidden), but only observed variables are variables that are influenced by the deliberations of the state will be written into the thesis with the title, "**HANDWRITING RECOGNITION USING THE HIDDEN MARKOV MODEL METHOD**".

2. CONTENT OF PAPER

2.1 Theoretical Basis

2.1.1 Image

The image is a two-dimensional image is generated from a two-dimensional analog image continuous discrete images into the sampling process [5]. the imagery is distinguished into two types, namely the image of analog and digital imagery. The image is the output of a data recording system that is optical, analog or digital. Image data recording is divided into 2 IE Analog Image and Digital Image. [7].

2.1.2 Citra Analog

The analog image is the image of a continuous nature, such as the images on television monitors, x-rays, photos printed on photo paper, paintings, landscapes, the results of the CT scan, images recorded on a cassette tape, and others [7]

2.1.3 Citra Digital

A digital image is a two-dimensional image that can be displayed on a computer screen as a set of discrete/digital called the pixel value/picture elements. In mathematical reviews, image is a continuous function of the intensity of the cahata on the two dimensional field [7]. An image can be defined as a function $f(x, y)$ size M rows and N columns, with x and y coordinates are spatial, and amplitude of f at the point at coordinates (x, y) were named the intensity or level of keabuan of the image at that point. If the value of x, y and value of amplitude f overall finite (finite) and the discrete value then it can be said that the image is a digital image [7]

2.1.4 Grayscale

Grayscale image handling tint black and white, which produces the effect of gray color, the color of the image declared with intensity. In this case, the intensity range from 0 up to 255. A value of 0 is black and 255 States declare white the following is conversion of colored (RGB) image into grayscale.

$$I = (0.2989 * R) + (0.5870 * G) + (0.1141 * B) \quad (2.1)$$

Description :

- R = Red value component (Red) of a pixel point
- G = green value component (Green) from a pixel point
- B = blue value components (Blue) of a pixel point

The above equation is one of the formulas that are used to convert a colored image into grayscale. [7].

2.1.5 Sauvola threshold

Sauvola threshold threshold method is capable of detecting handwriting image by calculating the threshold value of the dynamic range of the grayscale image standard deviation value [9]. Sauvola included in local threshold threshold value which is determined by the value of the neighboring pixels, threshold is a process he does aim to simplify the form image.

$$T(x, y) = m(m, y)n * (1 + k * \left(\frac{s(x, y)}{R} - 1\right)) \quad (2.2)$$

On the formula (2.2) there is a formula to calculate $m(x, y)$ which can be calculated using the formula (2.3) and the formula to calculate $s(x, y)$ can be seen in the formula (2.4)

$$m(x, y) = \frac{\sum_{i=\min}^{i=\max} \sum_{j=\min}^{j=\max} img(i, j)}{i * j} \quad (2.3)$$

$$s(x, y) = \frac{\sum_{i=\min}^{i=\max} \sum_{j=\min}^{j=\max} (img(i, j) - m(x, y))^2}{(i * j) - 1} \quad (2.4)$$

Description :

R: the maximum value of the standard deviation (128 to the image grayscale)

k: kernel with values between 0.2 – 0.5

m: the function that generates the average value from a number of image pixels.

s: a function that produces a deviation from a number of image pixels

Q: the function produces a value threshold (threshold)

x coordinate value: image width in the formula (i)

y: the coordinates of image height value in the formula (j)

After the threshold values $T(x, y)$ is didaptkan, then enter into the equation (2.5) below [9].

$$f(x, y) = \begin{cases} 0, & img(x, y) < T(x, y) \\ 255, & img(x, y) \geq T(x, y) \end{cases} \quad (2.5)$$

Description :

f: function that produces a value of 0 or 255

IMG: image grayscale values

2.1.6 Segmentation

Image segmentation is the separation of one object with another object in an image or between objects by setting contained in an image [10].

2.1.7 DEF Characteristic Extraction

DEF method on OCR which saw the difference in contour and without undergoing the process of skeletonizing [12] here are the stages that exist within the DEF extraction [12]

1. Image Scaling

The image will be scaling so that ukurannya becomes uniform $N \times N$ pixels, this process is done so that all images have the same size disegmentasi have to get a good characteristic

2. Dot Orientation

Directional Element Feature is the search value featured based on label directions of a pixel. can be seen on the image Example divider 3 zone on the binary images can be seen on the picture [3]

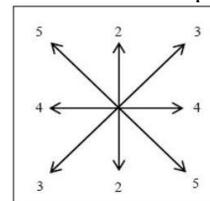


Figure 1 Direction Value

3. Vector Construction

This stage to calculate the value of each pixel neighborliness before, through a process of dot orientation, initially will be divided into the image (pixels)/ $N1 \times N1$ areas that overlap as much as $N1/2$ pixel against a neighboring area.

The next area, each area was divided again into 4 sub-areas, namely A, B, C and D, which are exclusive of each other. sub-areas with weights (w) in the sub-areas and aggregated from each element in the sub-areas with the following formula [3]:

$$x_j = w_A x_j + w_B x_j + w_C x_j + w_D x_j \\ j = 1, \dots, 4$$

Where j is the pixel neighborliness. Furthermore the DEF diperolh characteristic vector by combining the number of elements in every area into a single column vector V .

$$V = [X_1^1, X_2^1, X_3^1, X_4^1, \dots, X_1^N, X_2^N, X_3^N, X_4^N]$$

Where N is the number of characters from each area. A vector V is what nanntinya so the input to the process of learning and testing.

2.1.7 Hidden Markov Model

Hidden Markov Model (HMM) is modeling the probability of a system by looking for parameters that are not known to ease the process of analysis of the system [15]. Hidden Markov Models can be used for applications in the field of temporal pattern recognition == temporal pattern recognition such as voice recognition, writing, gestures, bioinformatics, computer vision, sentence compression, economics, finance, and the introduction not beam .

HMM is a variation of finite state machine that has a hidden condition, a value of output Q O (observation), the possibility of A transition, the possibility of output B , an initial condition π . Current conditions do not terobservasi. But, each State produces output likely b . Typically, Q dam O incomprehensible, so HMM called triple (A, B, π) .

- 1 the set of observed state: $O = o_1, o_2, \dots, o_N$.
- 2 the set of hidden state: $Q = q_1, q_2, \dots, q_n$
- 3 the probability of transition: $A = a_{01}, a_{02}, \dots, a_{n1}$
- 4 ... a_{nm} ; a_{ij} is the probability to move from state i to state j .
- 5 emission Probability or observation likelihood: $B = b_i(O_t)$, O_t observation probability is raised by state i .
- 6 State the beginning and end: q_{end}, q_0 , unrelated to the observations.

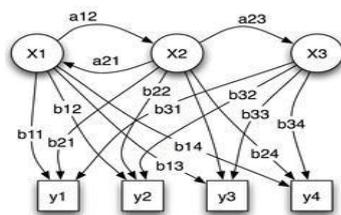


Figure 2 HMM Representation Parameter
Penjelasan Gambar 2.5:

- x = condition
- y = the possible observation
- a = the possibility of a transition state
- b = the possibility of an output example case

HMM

2.2 ANALYSIS AND IMPLEMENTATION

2.2.1 Process Analysis

The process is done in the application is divided into several parts where each process has a role in do

handwriting recognition. The process is done in Figure 3

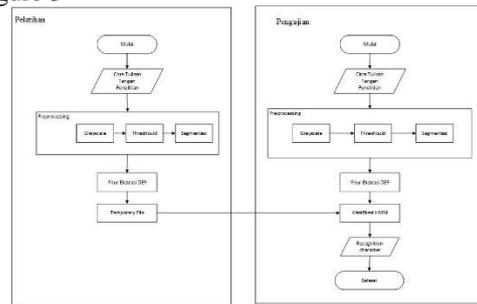


Figure 3 An Overview Of The System

The block diagram above, the initial process is done namely insert handwriting image into the application. The next step through image processing include grayscale thresholding, and feature extraction. From the stage of image processing will be obtained as a decimal value from the array every hallmark characters, decimal array is then processed back at the stage of classification method of HMM good training or testing. The result of the handwriting recognition was obtained by digital text

2.2.2 Grayscale

Change the format of the color to grayscale functions to shrink the range of colors to 0 up to 255.

As for the measures undertaken are as follows.

1. the image color is grouped by the values of red, green and blue
2. then using equation 2.1, it will be grayscale image color values obtained.
3. The value of grayscale obtained using the RGB value at each pixel.

Suppose the image at pixel (0,0) have the values Red = 255, Green = 219, Blue = 255 then based on equation 2.1 be

$$\begin{aligned} I &= (0,2989 * R) + (0,5870 * G) + (0,1141 * B) \\ &= (0,2989 * 255) + (0,5870 * 219) + (0,1141 * 255) \\ &= 76,2195 + 128,553 + 29,0955 \\ &= 233,868 \\ &= 234 \end{aligned}$$

From the above calculation, then the pixel that was worth a Red = 255, Green = 219, Blue = 255 = grayscale values updated to 234.

In Figure 5 below is the result of the process of data input on the image grayscale.



Figure 5 Grayscale Result Image

Berikut adalah tabel matrik warna *Red* (R), *Green* (G), dan *Blue* (B) dari hasil prose *grayscale*.

img(x,y)	Nilai Gray	m(x,y)	s(x,y)	T(x,y)	Nilai Baru
(0,0)	234	248	1.85	174	255
(1,0)	234	248	1.85	174	255
(2,0)	234	248	1.80	174	255
(3,0)	234	246	3.21	247	255
(4,0)	234	248	1.81	147	255
.
(143,77)	37	44	0.09	341	0
.
(360,173)	251	250	0,09	175	255

Table 1 Grayscale Matrix

X/Y	0	1	2	3	173
0	234	248	251	251		249
1	234	248	251	251		249
2	234	248	251	250		250
3	234	234	251	250		250
.....						
360	234	250	249	249		251

2.2.3 Thresholding

Threshold method used in this study i.e. sauvola method.

In this study, parameters are used as follows.

The number of neighboring pixels = 21. 21 suitable value because if the neighboring values is too large, then the time needed be longer, or if too small then the results didaptkan not the maximum.

1. Constant R = 128
2. K = 0.3
3. The value of m (x, y) didaptkan of the equation 2.3
4. The value of s (x, y) obtained from equations standard deviation of 2.4

Table 2 The Example of Matrix Image

X/Y	0	1	2	3	4	5	6	7	8	9	10
0	234	248	251	251	249	250	250	249	248	249	249
1	234	248	251	251	249	250	250	249	248	249	249
2	234	248	251	250	248	250	250	249	248	249	249
3	234	234	251	250	248	250	250	249	248	249	249
4	234	247	251	250	248	250	250	249	248	249	249
5	234	248	251	250	248	250	250	249	248	249	249
6	234	248	251	251	250	250	250	249	248	249	249
7	234	248	251	251	249	250	250	249	248	249	249
8	234	250	250	250	250	250	250	250	250	250	250
9	252	250	249	250	250	250	249	249	249	249	249
10	234	249	249	250	250	249	249	249	249	249	249

The first step in looking for the average value of m (0,0) with the equation 2.3, so the result as follows.

$$m(x,y) = \frac{\sum_{i=\min}^{i=\max} \sum_{j=\min}^{j=\max} img(i,j)}{i * j}$$

$$K(0,0) = \frac{234+248+\dots+249+249}{11*11} = 248$$

Then to find the value of the standard deviation s equations are (0,0) 2.4 so the result as follows.

$$s(x,y) = \sqrt{\frac{\sum_{i=\min}^{i=\max} \sum_{j=\min}^{j=\max} (img(i,j) - m(x,y))^2}{(i * j) - 1}}$$

$$s(0,0) = \sqrt{\frac{(234-248)^2+(251-248)^2+\dots+(249-248)^2+(249-248)^2}{121-1}} 1.86$$

the next step enter the value m (0,0) and s (0,0) 2.2 into the equation to get the value of the threshold T is (0,0). So the result into

$$T(x,y) = m(x,y) * (1 + k * (\frac{s(x,y)}{R} - 1))$$

$$T(0,0) = m(0,0) * (1 + 0,3 * (\frac{s(0,0)}{121} - 1))$$

$$= 248 * (1 + 0,3 * (\frac{1.86}{121} - 1))$$

$$= 174$$

The calculation of threshold values obtained above 107. The immediate next steps are included in the equation 2.5 so it will be a new pixel value is obtained. From the example above then the pixel value of the 153 initially will be changed to 255 since 153 greater than 107. The following is a table of results of calculations using sauvola threshold Here is a picture of the 4 images that already processed using sauvola threshold

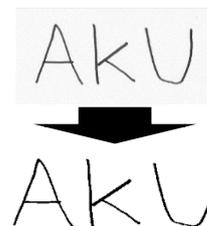


Figure 4 Sauvola Threshold Result

After threshold, then it is a process of binerisasi or change the value of the threshold value matrix that has previously been obtained into the binary values, namely 0 and 1, then the value 0 was changed to 1 and the value of 255 is changed to 0. Here is a flowchart of the process of image threshold binerisasi:

The following is a table of 4 which is the result of a process of calculation of binary threshold value matrix becomes:

Table 4 Biner Matrix

X/Y	0	1	2	3	...	173
0	0	0	0	0	...	0
1	0	0	0	0	...	0
2	0	0	0	0	...	0
3	0	0	0	0	...	0
...
360	0	0	0	0	...	0

2.2.4 Image Segmentation

In this process, the used input i.e. black and white imagery results sauvola threshold. Further cuts will be made to get the image of handwritten letters.

Cutting is performed for each row (horizontal) on the input image first, and then perform the cutting of each column (vertical) on each line in the cutting results image.



Figure 5 Segmentation Image

Berikut adalah gambar 6 hasil segmentasi citra tulisan tangan.



Figure 6 Cutting rows and columns on the image A,K,U

2.2.5 Characteristic Extraction

In this process, the input being used is the result of a character image segmentation has been done before. Next up will be performed the extraction of ciri [12]

1. Image Scaling

In this process the image that has been disegmentasi will be resized into a 15 x 15 pixels.

2. Teat Orientation

In this process will be determined neighborliness relations between pixels. The process of determining neighborliness follow the rules such as in Figure 2.2, and the value of the direction as shown in table 3.7 2.3 which is a matrix of black and white values from the image of A ketetanggan to be determined between the pixels. After that is done on the image, then the image will be divided again into 4 sub-areas, namely A, B, C and D would then be counted the number of pixels at each ketetanggan elmen sub-areas.

Area 1

Sub-Area A = (4.4) = 3

Sub-Area B (2.5) + (3.5) + (4.5) + (5.4) = 2 + 2 + 2 + 2 = 8

Sub-Area C (1.5) + (1.6) + (6.3) + (6.4) = 3 + 2 + 3 + 2 = 10

Sub-Area D (0.6) + (0.7) + (1.7) + (7.3) = 4 + 2 + 5 + 2 = 13

Area 2

Sub-Area A = 0

Sub-Area B = (4.2) + (5.2) = 5 + 2 = 7

Sub-Area C = (2.1) + (3.1) + (6.2) + (6.3) = 5 + 2 + 2 + 5 = 14

Sub-Area D = (0.0) + (1.0) + (7.3) = 2 + 5 + 5 = 12

Area 3

Sub-Area A = 0

Sub-Area B = (2.2) + (2.3) + (2.4) + (2.5) + (3.2) = 3 + 2 + 5 + 4 + 2 = 16

Sub-Area C = (1.3) + (2.6) + (4.1) + (5.1) = 2 + 4 + 3 + 2 = 11

Sub-Area D = (0.3) + (2.7) + (6.0) + (7.0) = 2 + 4 + 3 + 2 = 11

Area 4

Sub-area A = (3.3) + (3.4) + (4.4) = 5 + 5 + 5 = 15

Sub-Area B = (2.2) + (2.3) + (2.4) + (3.5) + (4.5) + (5.5) = 3 + 2 + 5 + 5 + 5 + 5 = 25

Sub-Area C = (1.3) + (2.1) + (5.6) + (6.6) = 2 + 4 + 5 + 5 = 16

Sub-Area D = (0.3) + (2.0) + (6.7) + (7.7) = 5 + 4 + 5 + 5 = 19

Once retrieved amount of pixels in each sub area, will be multiplied by the respective amount of each element of neighborliness sub-areas with weights (w) in the sub-areas and aggregated from each element in the sub-areas with 2.6 equation, as follows.

$$x_j = w_A x_j + w_B x_j + w_C x_j + w_D x_j$$

$$j = 1, \dots, 4$$

J= Pixel Tetangga

W= Sub Area

X1 = (3.1) + (8.4) + (10.4) + (13.4) = 3 + 32 + 40 + 52

X2 = (0) + (7.2) + (14.4) + (12.3) = 0 + 14 + 56 + 36

X3 = (0) + (16.5) + (11.4) + (11.4) = 0 + 80 + 44 + 44

X4 = (15.3) + (25.6) + (16.4) + (19.4) = 45 + 150 + 64 + 76

A vector can be formed next DEF characteristics are obtained by combining the number of elements in each area into a single column vector V by equations 2.7:

$$V = [X_1^1, X_2^1, X_3^1, X_4^1, \dots, X_1^N, X_2^N, X_3^N, X_4^N]$$

Vektor Karakter A:

v = [3, 32, 40, 52, 0, 14, 40, 36, 0, 80, 21, 21, 45, 150, 64, 76]

Vektor Karakter A:

V[3, 32, 40, 52, 0, 14, 56, 36, 0, 80, 44, 44, 45, 150, 64, 76]

Vektor karakter K

V[0, 0, 80, 308, 4, 24, 14, 10, 0, 100, 410, 280, 5, 68, 68, 56]

Vektor karakter U

V[0, 0, 36, 90, 0, 0, 0, 98, 20, 44, 20, 100, 0, 10, 60, 184]

After that, the vector will be normalized with min-max.

Table 5 Biner Vb Matrix

i	1	2	3	4	5	6	7	8	9	1	1	1	1	1	1	1
A	1	1	0	0	0	1	1	0	0	1	0	0	1	1	0	0
K	0	0	1	1	1	1	0	0	0	1	1	1	0	0	1	0
U	0	0	0	0	0	0	0	1	1	0	0	0	0	0	0	1

Biner Vector A:

V[1,1,0,0,0,1,1,0,0,1,0,0,1,1,0,0]

Biner Vector K:

V[0,0,1,1,1,1,0,0,0,1,1,1,0,0,1,0]

Biner Vector U:

V[0,0,0,0,0,0,0,1,1,0,0,0,0,0,0,1]

Here are the steps to determine the Viterbi algorithm class letter A V[1,1,0,0,0,1,1,0,0,0,0,0,1,1,0]

2.2.7 Inisialisasi

This step uses the first value of a vector (z1), i.e. 1.

$$\delta_1(i) = \pi_i * B_{i,z_1}, \quad i = 1, \dots, N$$

$$\psi_1(i) = 0$$

State A:

$$\begin{aligned}\delta_1(A) &= \pi_A * B_{A,1} \\ \delta_1(A) &= 0,33 * 0,4375 \\ \delta_1(A) &= 0,144375 \\ \psi_1(A) &= 0\end{aligned}$$

State K:

$$\begin{aligned}\delta_1(K) &= \pi_K * B_{K,1} \\ \delta_1(K) &= 0,33 * 0,5 \\ \delta_1(K) &= 0,165 \\ \psi_1(K) &= 0\end{aligned}$$

State U:

$$\begin{aligned}\delta_1(U) &= \pi_U * B_{U,1} \\ \delta_1(U) &= 0,33 * 0,1875 \\ \delta_1(U) &= 0,061875 \\ \psi_1(U) &= 0\end{aligned}$$

2.2.8 Rekursive

This stage is carried out for the remainder of the vector values i.e. the second value to the value of the sixteenth (z_2, \dots, z_{16}).

$$\begin{aligned}\delta_n(j) &= \max_{1 < i \leq 3} (\delta_{n-1}(i) * A_{ij}) * B_{j,z_n} \\ \psi_n(j) &= \operatorname{argmax}_{1 < i \leq 3} (\delta_{n-1}(i) * A_{ij})\end{aligned}$$

Examples for the second value (z_2) is 1, then the calculations are as follows.

State A:

$$\begin{aligned}\delta_2(A) &= \max(\delta_1(A) * A_{AA}, \delta_1(K) * A_{AK}, \delta_1(U) * \\ &A_{AU}) * B_{A,1} \\ \delta_2(A) &= \max(0,144375 * 1, 0,165 * 0, 0,061875 * \\ &0) * 0,4375 \\ \delta_2(A) &= 0,144375 * 0,4375 \\ \delta_2(A) &= 0,063164 \\ \psi_2(A) &= \operatorname{argmax}(\delta_1(A) * A_{AA}, \delta_1(K) * \\ &A_{AK}, \delta_1(U) * A_{AU}) \\ \psi_2(A) &= \operatorname{argmax}(0,144375 * 1, 0,165 * \\ &0, 0,061875 * 0) \\ \psi_2(A) &= A\end{aligned}$$

State K:

$$\begin{aligned}\delta_2(K) &= \max(\delta_1(A) * A_{KA}, \delta_1(K) * A_{KK}, \delta_1(U) * \\ &A_{KU}) * B_{K,1} \\ \delta_2(K) &= \max(0,144375 * 0, 0,165 * 1, 0,061875 * \\ &0) * 0,5 \\ \delta_2(K) &= 0,165 * 0,5 \\ \delta_2(K) &= 0,0825 \\ \psi_2(K) &= \operatorname{argmax}(\delta_1(A) * A_{KA}, \delta_1(K) * \\ &A_{KK}, \delta_1(U) * A_{KU}) \\ \psi_2(K) &= \operatorname{argmax}(0,144375 * 0, 0,165 * \\ &1, 0,061875 * 0) \\ \psi_2(K) &= K\end{aligned}$$

State U:

$$\begin{aligned}\delta_2(U) &= \max(\delta_1(A) * A_{UA}, \delta_1(K) * A_{UK}, \delta_1(U) * \\ &A_{UU}) * B_{U,1} \\ \delta_2(U) &= \max(0,144375 * 0, 0,165 * 0, 0,061875 * \\ &1) * 0,1875 \\ \delta_2(U) &= 0,061875 * 0,1875\end{aligned}$$

$$\delta_2(U) = 0,011602$$

$$\begin{aligned}\psi_2(U) &= \operatorname{argmax}(\delta_1(A) * A_{UA}, \delta_1(K) * \\ &A_{UK}, \delta_1(U) * A_{UU}) \\ \psi_2(U) &= \operatorname{argmax}(0,144375 * 0, 0,165 * \\ &0, 0,061875 * 1) \\ \psi_2(U) &= U\end{aligned}$$

Following are the results of the calculation of the recursive stages for the other z_i values.

2.2.9 Termination

Search the value of max and argmax of last termination phase δ .

$$\begin{aligned}p^*(X|\Theta) &= \max_{1 < i \leq 3} (\delta_{16}(i)) \\ p^*(X|\Theta) &= \max(\delta_{16}(A), \delta_{16}(K), \delta_{16}(U)) \\ p^*(X|\Theta) &= \\ &\max(0,00000734, 0,00000504, 0,00000180)\end{aligned}$$

$$\begin{aligned}p^*(X|\Theta) &= 0,00000734 \\ q_{16}^* &= \operatorname{argmax}_{1 < i \leq 3} (\delta_{16}(i)) \\ q_{16}^* &= \operatorname{argmax}(\delta_{16}(A), \delta_{16}(K), \delta_{16}(U)) \\ q_{16}^* &= \\ &\operatorname{argmax}(0,00000734, 0,00000504, 0,00000180)\end{aligned}$$

$$q_{16}^* = A$$

Then the state last observation is A.

2.2.10 Track Back

At this stage the best sequence will be made from the state based on the results of observation. Sequence can best be seen from the values of ψ . Sequence created from end to beginning of the ordered.

$$q_n^* = \psi_{n+1}(q_{n+1}^*), n = 16 - 1, 16 - 2, \dots, 1$$

Example for the value of the 15th (z_{15}), the calculations are as follows.

$$\begin{aligned}q_{15}^* &= \psi_{16}(q_{16}^*) \\ q_{15}^* &= \psi_{16}(A) \\ q_{15}^* &= A\end{aligned}$$

Do to all of z_i , resulting in the following sequence.

$Q^* = A, A$
Take the last sequence value as a prediction letter class. Then the results of the prediction of the class letters on the image are tested is A.

2.2.11 Use case Diagram

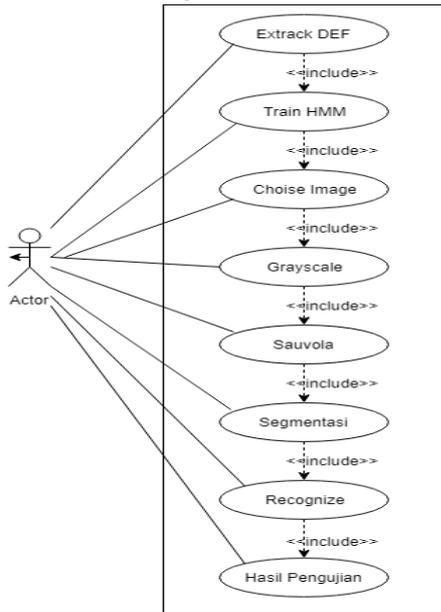


Figure 7 Use Case Diagram

2.2.12 Hardware Implementation

Here is an implementation of the interface of the system introduction to writing

Komponen	Spesifikasi
Processor	AMD A9-9420 (2CPUs), ~3,0GHz
Memory	4GB DDR 4

2.2.13 Software Implementation

The software used in building applications such as on a table

Komponen	Spesifikasi
Sistem Operasi	Windows 10 Home 64-bit
Java	NetBeans IDE 8,2

2.2.14 Interface Implementation

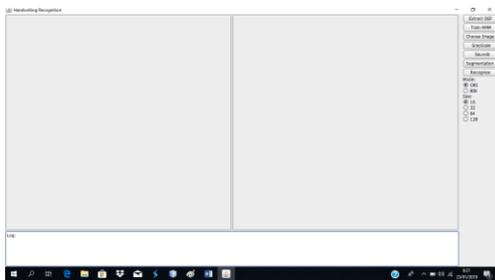


Figure 8 Interface Of a Program

2.2.15 Implementasi pengenalan pada program

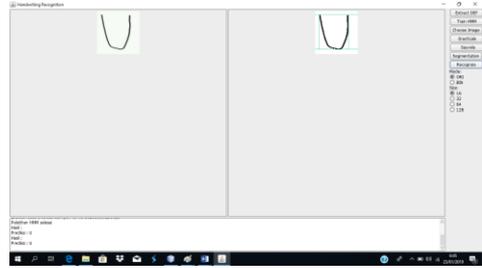


Figure 9 Interface of Testing

2.2.16 Accuracy Testing

Testing the accuracy of data is done by first Testing training as many as 820, 820 data consist of letters A-Z as much as 520 each letter has 20 data samples. And the number 0-9 has 300 sample data that each has 30 sample data has been previously trained and image that has not been trained. The following is the formula for calculating the accuracy.

$$\text{Akurasi} = \frac{TP}{(TP + FN_1 + \dots + FN_6)} \times 100\%$$

Description :

1. True Positive (TP) is the positive data is detected, and
2. False Negative (FN) is the opposite of the True Positive, so that the positive data, but the data is detected as negative.
3. Each column in a matrix represent the predicted class, while each row represents the actual class.

Table 6 Confusion Matrix Testing Result

Sebanarnya	Prediksi										
	0	1	2	3	4	5	6	7	8	9	
0	1										50 %
1		1	1								50 %
2			1								50 %
3				1	1						50 %
4					1						50 %
5						1					50 %
6							1			1	0 %
7					1			1			0 %
8		1							1		50 %
9				1						1	50 %
	Rata – Rata Akurasi										40 %

Accuracy of the results in table 6 is an example of the calculation of the confusion matrix of experiments testing numbers 13.

3. CONCLUDING

3.1 Conclusion

Based on the results of testing accuracy use their Confusion matrix against 72 test data that has been completed, the conclusion the results obtained against the accuracy of handwriting recognition is as follows:

1. Training Data from the method used at the time HMM training greatly affect accuracy.
2. Neatness in the training data are made very influential to achieve good accuracy results
3. after the test with 72 test data then the accuracy results that have the highest percentage of the obtained results the accuracy of 56.94%.
4. the test results in table 6 is an example of the confusion matrix calculation

3.2 Suggestions

The advice of researchers after completing this final task for further research related to handwriting recognition using Hidden Markov models. As follows:

1. the Required image segmentation method of handwritten text that is attached to each other or stick.
2. Neatness in the training data are made very influential to achieve good accuracy results
3. It takes a feature extraction method is better for images such as Principal Components Analysis (PCA), the Wavelet Transform.

REFERENCES

- [1] R. Tiofan, "Pengenalan Tulisan Tangan Menggunakan Metode Curvelet Transform Dan Jaringan Saraf Tiruan Backpropagation Untuk Kasus Penilaian Esai," in *Unikom*, Bandung, 2017.
- [2] I. A. Ersyaputra, "Analisis Dan Implementasi Pengenalan Citra Tulisan Tangan Alfabet Menggunakan Metode Modified Direction Feature dan Klasifikasi Jaringan Saraf Tiruan-Learning Vector Quantization," *Skripsi Fakultas Teknik Informatika, Universitas Telkom*, 2013.
- [3] X. D. H. L. Xianliang Wang, "Writer Identification Using Directional Element Features and Linear Transform," *Department of Electronic Engineering*, 2003.
- [4] A. A. Nugraha, "Pengenalan Pola Huruf Hijaiyah Berharakat Menggunakan Modified-DFE dan HMM.," *Institut Teknologi Telkom*, 2013.
- [5] E. F. Yuwitaning, D.. Bambang Hidayat and S. M. Nur Andini, "Implementasi Metode hidden Markov Model Untuk Deteksi Tulisan Tangan," *e-Proceeding of Engineering*, vol. I, p. 396, Desember 2014.
- [6] Sugiyono, *Metode Penelitian Kombinasi (Mixed Methods)*, Bandung: Alfabeta, 2013.
- [7] K. D. A and S. A, *Teori dan Aplikasi Pengolahan Citra*, Yogyakarta: ANDI, 2013.
- [9] K. Khurshid, I. siddiqi, C.Feure dan N. Vincent, "Comparasion of Niblack Inspired Binarization Methods for Ancient Document," *Document Recognition and Retrival XVI*, 2009.
- [10] P. A. Cahyan, I. M. Muhammad Aswin and S. M. Ali Mustofa, "Segementasi citra dengan menggunakan algoritma watershed dan lowpass filter sebagai prosedur awal," *Jurusan Teknik Elektro, Universitas Brawijaya*, november 2013.
- [11] S. Suyanto, *Artificial Intelligence (Searching, Reasoning, Planning dan Learning)*, Bandung: Informatika, 2014.
- [12] Aulia Husni Nurul A I , Mahmud Dwi Sulistiyo and Retno Novi Dayawati, "Pengenalan aksara jawa tulisan tangan menggunakan directional element feature dan multi class support vector machine," *Konferensi Nasional Teknologi Informasi dan Aplikasinya*, 13 September 2014.
- [13] J. a. K. M. Han, *Data Mining Concept and Techniques*, McGraw Hill, Inc, 2006.
- [15] P. M. E. Budi, "Teori Dasar Hidden Markov Model," *Makalah I2092 Probabilitas dan statistik*, 2010/2011.