Support Vector Machine In Personality Detection System Based On Signature Pattern

Bhakti Prasetya Utama[1], Ken Kinanti Purnamasari[2]
Informatic Engineering – Universitas Komputer Indonesia
Jl. Dipatiukur 112-116 Bandung, 40312
bhaktiprasetyautama95@gmail.com[1], ken.kinanti@email.unikom.ac.id [2]

Abstract—*Signatures have a specific pattern based on features that are viewed from graphology to be able to express the personality of the author, but now the system built to detect personality based on the signature pattern automatically is not so good to use. In this research will try to test the system with different method that is support vector machine from some previous research to identify signature pattern. The feature test is based on the nine features with five features: the start of the curve, the final stroke, the shell, the middle stroke and the bottom line using the support vector machine (SVM). And four other features are extreme margins, point structures, separate signatures, and streaks of disconnected lines using the rulebased method. Training with the SVM method uses 50 training data for each feature. The result of SVM identification of 50 signature test data on each feature has the highest accuracy 89% and the whole accuracy is 60.7% but the lowest accuracy is 41%. In rulebased method, the highest accuracy of signature classification was 70% and 34% with the average accuracy of 52.8%.*

Keyword— *Support vector machine, SVM, signature pattern, graphology, rulebased.*

I. INTRODUCTION

Personality can be detected through handwriting. The detection of a person's personality by handwriting is an art or science called graphology [1]. In graphology, handwriting is analyzed with structural graphical elements in order to obtain information about the author's personality. This analysis is based on 7 (seven) categories put forward by European intellectuals in the 19th century. It is a dimension, shape, pressure, speed, direction, layout and continuity [1]. The formation of this category makes the theory and guidance in analyzing handwriting patterns. Similarly, the signature is part of the handwriting. Signatures can also reflect a person's personality based on a certain pattern.

Currently graphology has attracted a lot of interest researchers in conducting the development of computer-based systems. In its development personality detection can be recognized by utilizing machine learning method.

II. RELATED WORK

Similar studies have previously been conducted by Esmeralda, Risna and Nur Ramdlan to analyze handwriting patterns in personality detection using the Multi Layer Perceptron (MLP) method. The results of the study obtained an accuracy of 80% [2]. Similar research is also done by K.Nithya, A.Keerthana and R.Lakshmi by using fuzzy method. From this research, the accuracy is 86% [3]. However, both studies still result in less classification accuracy than studies conducted by Shitala, Vivek and Akshay using the *Support vector machine* (SVM) method. From their research yielded an accuracy of 90% [4].

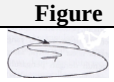
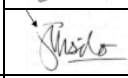
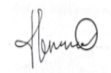
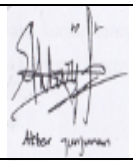
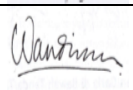
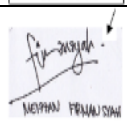
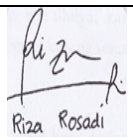
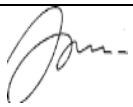
The main focus of this research is in the case of signature pattern analysis to know the personality of the author. Several previous studies on signature pattern cases, ie research conducted by Djamal and Sheldy using MLP method and research by Thoyib using Learning Vector Quantization (LVQ) method. Both studies received an accuracy of 56-78% [8] and 53% [9]. From these studies it can be seen that the classification results for signature pattern analysis is still not very good. Therefore, the testing of signature pattern analysis is expected to improve the quality of classification accuracy.

Based on the above explanation, this research will conduct testing of personality detection system based on signature pattern by using *Support vector machine* method as its classification

III. SIGNATURES IN GRAPHOLOGY

Signature analysis is based on nine features to illustrate whether there are certain personality appearances of the 15 types reviewed. For more details, it is shown in Table 1, the signature pattern along with the accompanying personalities.

Table III Features of Signature[2]

Fitur	Figure	Characteristic	Personality
Shell		Closed Curve	Excessive fear, introvert, no care around, do not like to hang out and work together.
Early curves		Curve backwards	Comfortable to the past.
		The arch is sharp	Able to formulate thoughts sharply.
		Gentle arch	Be careful, friendly, diplomatic.
Final Graffiti		Ascending	Open, forward-looking, forward, confident.
		Downhill	Lack of passion, realistic thinking, lack of confidence, easy to despair.
Scribble Medium		The existence of graffiti	Possessive
Underline		The existence of Underline	Having unique ideas and ways of thinking, requiring decision support, and having reliability in the lead.
Extreme margin		Tend to the right	Sloppy, lacking attention
		Tend to the left	Fear of failure, fear of others, lack of confidence, pessimism
		Tends to be on top	Respect to yourself, reflects a happy person.
		Tend down	Depression, shy, feel alien
Dot structure		There is a dot	Establishment is stable, has a sense of suspicion, always keep the distance is not easy to believe.
Separate Signature		There is a separate signature	Have had less enjoyable experiences in the past.
Separate line		There is a separate line	Limit his desire, not dare to take risks, often discouraged and hesitate to make decisions.

IV. THEORETICAL BASIS

A. Digital Image

Digital imagery is a two-dimensional image that can be displayed on a computer screen as a set / discrete digital value called pixels. In a mathematical review, the image is a continuous function of the intensity of light in a two-dimensional plane [14]. A photo is an example of two-dimensional images that can be easily processed. Each photo in the form of a digital image can be processed using a particular software. In this research, image processing is done such as grayscale, threshold, segmentation, resize and feature extraction.

B. Grayscale

Grayscale is used to simplify the advanced process by simplifying the values in the image matrix to the color range of puttih and gray. Here is the process of changing the image to grayscale yand can be seen in Figure 1.1 below .

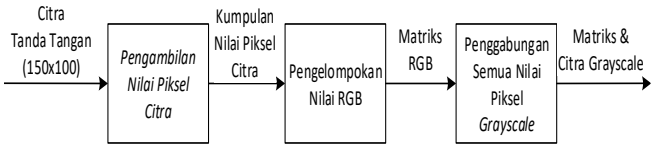


Figure IV Block diagram Process *Grayscale*

Grayscale is a process that converts the RGB matrix value (Red, Green, Blue) to the image to the gray value using equation 1 below.

$$f(x,y) = wR.Red + wG.Green + wB.Blue \quad (1)$$

where, x = row and y = columns in the image matrix. The values of wR, wG, and wB are respectively weightings for the Red, Green and Blue color elements corresponding to the weighing definition for the conversion of color images to the gray scale where wR = 0.299, wG = 0.587 and wB = 0.114 by NTSC (National Television System Committee)[15].

The workings of this process are:

1) Image colors are grouped according to Red, Green and Blue values.
2) Then using equation 1 above, it will get the color value grayscale image.
3) The resulting grayscale value uses the RGB value on each pixel.Here is an example of a grayscale image



Figure 1.1 Grayscale Results

C. *Canny Edge Detection*

Edge is a change in the value of the intensity of the gray degree quickly or suddenly (large) within a short distance. The purpose of detecting the edge itself is to group the objects in the image, and also used to analyze the image further. There are many algorithms used to detect edges, one of which is Canny edge detection (Canny Edge Detection). Canny was developed by John F. Canny in 1986 and used a multi-stage algorithm to detect various edges in images. Step Algorithm Canny that runs in this research that is:

1. Smoothing: Blur the image to remove noise
2. Finding gradient: The edge should be marked on the image that has a large gradient. In this study using gradian operator sobel
3. Non-maximum-suppresion: Only local maximum values should be marked as edges.
4. Double thresholding: The potential edge as edge is determined by thresholding. The threshold value is divided into two, the upper and lower threshold values. In this study the upper threshold value is 0.075 and the lower threshold value is 0.175.

It can not be denied that all images taken from the camera or scanner will contain some noise. In preventing the wrong noise is detected as the edge, the noise should be reduced. Therefore in the first step the image must be smoothed by using the filter gaussian shown in the equation below.

$$g(x,y) = \frac{1}{159}\begin{bmatrix} 2 & 4 & 5 & 4 & 2 \\ 4 & 9 & 12 & 9 & 4 \\ 5 & 12 & 15 & 12 & 5 \\ 4 & 9 & 12 & 9 & 4 \\ 2 & 4 & 5 & 4 & 2 \end{bmatrix}$$

Information :

$g(x,y)$ = Convolution kernel matrix

The gradient of each smoothed image pixel is determined by applying the Sobel operator. The next step is to estimate the gradient in the x and y directions. This is shown in the Equation below.

$$G_x = \begin{bmatrix} -1 & 0 & 1 \\ -2 & 0 & 2 \\ -1 & 0 & 1 \end{bmatrix} dan\ G_y = \begin{bmatrix} 1 & 2 & 1 \\ 0 & 0 & 0 \\ -1 & -2 & -1 \end{bmatrix}$$

The magnitude of the gradient (also known as edge strength) can be determined as the Euclidean distance measured by applying the pythagoras law as shown in (2)

$$G[f(x,y)] = \sqrt{G_x^2 + G_y^2} \quad (2)$$

Where, Gx and Gy are gradients in each of the x and y directions.

Then to determine the actual edge, the next step is to determine the edge direction using equation 4 below.

$$\theta = arctan\left(\frac{|G_y|}{|G_x|}\right) \quad (3)$$

Information :

$\theta = theta\,(sudut\ arah\ tepi)$

Then if a negative value is found in the matrix after performing the calculation through equation 3 then the following equation 4 is used to obtain a positive value across all matrix values.

$$M(x,y) = p\ ;\ if\ \ (M(x,y) < 0) \quad (4)$$
$$untuk\ p = 360 + M(x,y)$$

Information :

$M(x,y)$ = Edge Direction Matrix

After all the values of the matrix are positive the next step is to filter the edges into values (0, 45, 90 and 135) in degrees. Here's the picture 3 used to determine the edge range.

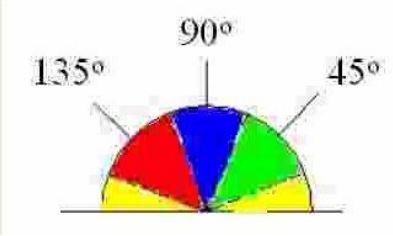


Figure 3 Direction of Canny Edges

In determining the span of the edge direction, the matrix values obtained from equation 5. From that equation the falling edge values in the yellow range (0-22.5 and 157.5-180) will be normalized to 0 degrees. If the edge direction that falls within the green range (22.5-67.5) it will be normalized to 45 degrees. Then the edge direction falling in the blue range (67.5-112.5) will be normalized to 90 degrees and the edge direction falling in the red range (112.5-157.5) will be normalized to 135 degrees.

In this step it aims to convert the edges that are still blurred on the magnitude image to produce sharp edges by maintaining the local maximum value in the gradient image and erasing everything else. The last step in the canny algorithm is to determine the double thresholding. The condition of thresholding is that the image has determined the non-maximum value of suppresion and defines the upper and lower threshold values. The value of the upper and lower threshold has been defined in the previous explanation that is worth 0.075 for the top and 0.175 for the bottom. The purpose of double thresholding is to change the pixel value of the matrix between 0 and 255. Here to determine the double thersholding is written with equation 5 as below.

$$Nmax(x,y) = \begin{cases} a & if\ Nmax(x,y) < ta \\ b & if\ Nmax(x,y) > tb \end{cases}$$

dimana,

$a=0$
$b=255$
$ta = 0.075 \times \max(Nmax(x,y)$
$tb = 0.175 \times \max(Nmax(x,y)$ (5)

More clearly the value of ta and tb is also called hysteresis. Hysteresis is the unbound edge linking to bring up other strongly detected edges. If the pixel gradient value is between the upper threshold and bottom threshold values, then the pixel is a weak edge.

D. Object Segmentation

In computer vision, segmentation is the process of partitioning digital images into segments. The purpose of segmentation is to simplify or change the presentation of

images to something more meaningful and easier to analyze [14]. The results of this segmentation process will be used for further high-level processes that can be performed on an image. The following is the process of object segmentation in this study which can be seen in Figure 4

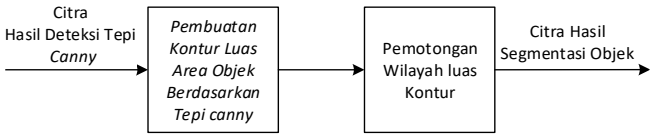


Figure 4 Block diagram Process Object Segmentation

Cutting the object is done with the contours available in the OpenCV library. The contour takes the edge detection value generated by the canny operator to determine the area around the object. After the area is obtained will be cutting grayscale image as in the following picture 3.11.

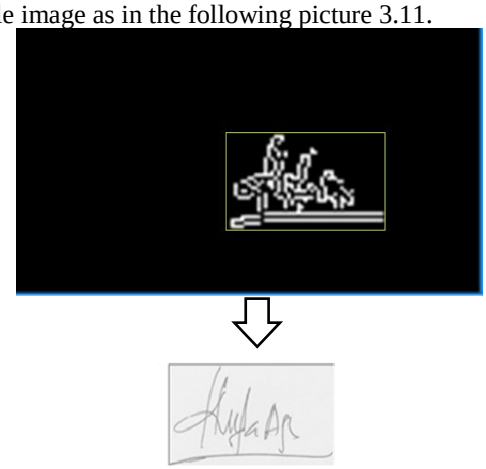


Figure 5 Result of Object Segmentation

The Cutting Image results represent an object in the image. Cutting that obtains the object can provide the size of the matrix. which is different when done using another input image.

E. *Resize*

Resize aims to equalize the image size of Object Segmentation results to be processed by the next process is the feature extraction of Vertical & Horizontal Segment. In this study used image resize size 69 x 36 pixels because the size does not reduce the image quality too far from the original image size, for the purposes of analysis in this study used image size of 36 x 24 pixels. Here is a picture 3.12 resize.



Figure 6 Result of Resize

F. Binary Image

At this stage the matrix of the resized process that has been done Object Segmentation simplified the value of the matrix between 0 and 1 or in other words to binaryize the image. This is done by simple thresholding method and after getting the result of matrix threshold then divide all pixels in coordinates (x, y) with 1/255. Here is the block diagram 7 of Image Binerization process.

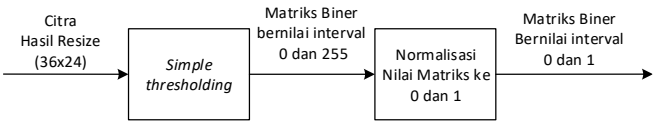


Figure 7 Block diagram Process Binary Image

The threshold used in this study uses Simple Thresholding from the OpenCV library whose purpose is to distinguish objects and backgrounds from the signature image so that information characteristics are easier to recognize. Here is the matrix of thresholding process results that can be seen in table 8 below.

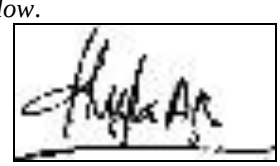


Figure 8 Result of Threshold

G. Segmentation Vertical & Horizontal

At this stage the image of the binary process will be continued gradually vertical and horizontal segmentation. Vertical segmentation is processed by dividing the two regions of the signature image region vertically to distinguish the region of the initial features of the curve and the final stroke. The division of this region is done by cropping the size of the threshold matrix [36x24] into two parts each with a matrix [12x24]. As with vertical segmentation, for horizontal segmentation can be done by dividing the two parts of the image region of the signature horizontally to distinguish the feature areas of the graffiti in the middle and the bottom line. The two parts are each divided by the matrix size [36x8]. Here is the block diagram 9 vertical and horizontal segmentation process.

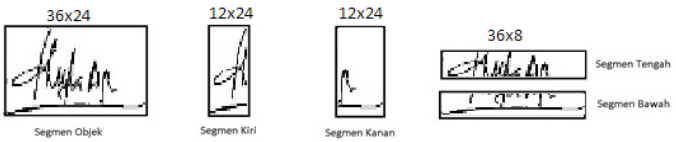


Figure 9 Result of Segmentation

H. SUPPORT VECTOR MACHINE

Support vector machine is one of machine learning method to convert input data into vector data. This method was developed by Boser, Guyon, Vapnik in 1992. SVM is a technique for finding hyperplanes that can separate two sets of data from two different classes. Hyperplane is the boundary

line of data separation between classes, while the margin is the distance between the hyperplane with the closest data in each class. The closest data to hyperplane in each class is called *support vector* [16].

The data label is a member of two data classes, namely +1 and -1. Data belonging to class -1. In linear *Support vector machine* separator is a linear function. Training data expressed by (x_i, y_i) and x_i = { x_1, x_2, …,x_i} is an attribute (feature) set for i-th class practice data. For y_i€{-1,1} declared the class label. Defining the equation of a separator hyperplane is written by:

$$wx_i + b = 0 \quad (6)$$

Data x_i divided into two classes, which belong to class -1 (negative samples) are defined as vectors that satisfy the following inequalities 7

$$wx_i + b < 0 \; untuk \; y_i = 1 \quad (7)$$

While those belonging to the +1 class (positive samples) meet the following 8 inequalities.

$$wx_i + b > 0 \; untuk \; y_i = +1 \quad (8)$$

Dimana:

x_i = data input (vektor)

y_i = label yang diberikan (skalar)

w = nilai dari bidang normal (vektor)

b = posisi bidang relatif terhadap pusat koordinat (skalar)

The parameters w and b are parameters to be searched for. When labeling data $y_i = -1$, then the boundary becomes the following equation 9.

$$wx_i + b \leq -1 \quad (9)$$

When labeling data y_i = +1, then the boundary becomes the following equation 10.

$$wx_i + b \geq +1 \quad (10)$$

The greatest margin can be searched by maximizing the distance between the boundary fields of both classes and their nearest point: $\frac{2}{|w|}$. This is formulated as a quadratic programming problem (QP) problem that is to find the minimum point of equation (11) by considering the following equation (12).

$$\min \tau(w) = \frac{1}{2}||w||^2 \quad (11)$$

$$y_i(wx_i + b) - 1 \geq 0, (i = 1, \dots, n) \quad (12)$$

This problem can be solved by sharing computational techniques. It is more easily solved by converting the equations into the Lagrangian function. Value a_i is a lagange multiplier that is zero or positive ($a_i \geq 0$). The optimal value of equation 16 can be calculated by minimizing L against w, b, and. Can be seen in the following 14 to 16 equations.

$$\frac{\partial L}{\partial w} = w - \sum_{i=1}^{n} a_i \, y_i \, x_i = 0 \quad (14)$$

$$\frac{\partial L}{\partial b} = \sum_{i=1}^{n} a_i \, y_i = 0 \quad (15)$$

$$\frac{\partial L}{\partial a} = w - \sum_{i=1}^{n} a_i \, y_i (w^T x_i + b) + \sum_{i=1}^{n} a_i = (16)$$

Then the Lagrange problem for classification can be expressed in the following equation 17 by maximizing L against a_i, the equation becomes the following equation.

$$Max \sum_{i=1}^{n} a_i - \frac{1}{2} \sum_{i=1,j=1}^{n} a_i y_i {y_j}^T x_i {x_j}^T \quad (17)$$

Consider the following equation 24.

$$\sum_{i=1}^{n} a_i y_i = 0, a_i > 0 \; (i, j = 1, \dots, n) \quad (18)$$

Looking for value x_i and y_i can be done when already got the result of feature extraction process and class initialization. The result is converted into svm data format (*x*), while the class data becomes svm data label (*y*).

Nilai a_i obtained by the way is to change each input data into vector value (support vector) =$\begin{pmatrix} x_i \\ y_i \end{pmatrix}$. Then the vector value of each input data is entered into the 19 kernel trick phi equation (φ) below.

$$\varphi \begin{bmatrix} x_i \\ y_i \end{bmatrix} = \begin{cases} \sqrt{x_n^2 + y_n^2} > 2, maka \; \begin{bmatrix} \sqrt{x_n^2 + y_n^2} - x + |x - y| \\ \sqrt{x_n^2 + y_n^2} - y + |x - y| \end{bmatrix} \\ \sqrt{x_n^2 + y_n^2} < 2, maka \; \begin{bmatrix} x_i \\ x_i \end{bmatrix} \end{cases}$$

The value of x is obtained from the linear kernel equation for x below..

$$\sum_{i=1,j=1}^{n} x_i^T x_j \; (i, j = 1, \dots, n) \quad (20)$$

The y value is obtained from the linear kernel equation for y follows.

$$\sum_{i=1,j=1}^{n} y_i^T y_j \; (i, j = 1, \dots, n) \quad (21)$$

The optimal perpendicular distance can be considered by considering the positive vector, then the result of the calculation of the substitution of the value of x and the value of y to equation 20 given the bias value = 1. Then find the parameters, by first looking for the value of each abstract function using equation 20, then look for the value on the linear equation using equation 21 with respect to $i, j = 1, \dots, n$ below

$$\sum_{i=1,j=1}^{n} a_i {T_i}^T T_j \quad (22)$$

$$\sum_{i=1,j=1}^{n} a_i {T_i}^T T_j = y_i \quad (23)$$

Where, T is the value of the support vector.

After the parameters a_i obtained, then enter into the following equation 24.

$$\tilde{W} = \sum_{i=1}^{n} a_i T_i \quad (24)$$

The results obtained using equation 24, then used equation 31 to get the value of w and b are as follows.

$$y = wx + b \quad (25)$$

Thus, the values of w and b or hyperplane values are obtained to classify the two classes. The following kernel functions can convert non-linear data sets into linear in the new space.

a) Linier

$$K(x_i x_j) = x^T x \quad (26)$$

b) Polynomial

$$K(x_i x_j) = (x^T x_i + 1)^p \quad (27)$$

c) Gaussian

$$K(x_i x_j) = \exp - \|x_1 - x_2\|^2 / 2\sigma \quad (28)$$

In this research, the kernel used is linear kernel and classification process of *Support vector machine* using LIBSVM Scikit-learn python.

I. *RULEBASED*

The rule based method is one of the methods in the decision support system in which it has a knowledge base. The mechanism of rule-based is done by testing the rules one by one in a certain order. When each rule-tested rule will evaluate whether the condition is right or wrong. If the condition is correct then then proceed to the next rule tested. Conversely the conditions are wrong, the rules are not stored and the next rule is tested. This process will repeat until the entire rule base is tested under various conditions.

The input data to be processed *Support vector machine* is a scanned image of the stored signature in the form .jpg file. In this analysis the amount of data taken from the correspondent amounts to 100 different signature data consisting of each of 50 data signatures for each feature.

J. GENERAL SYSTEM OVERVIEW

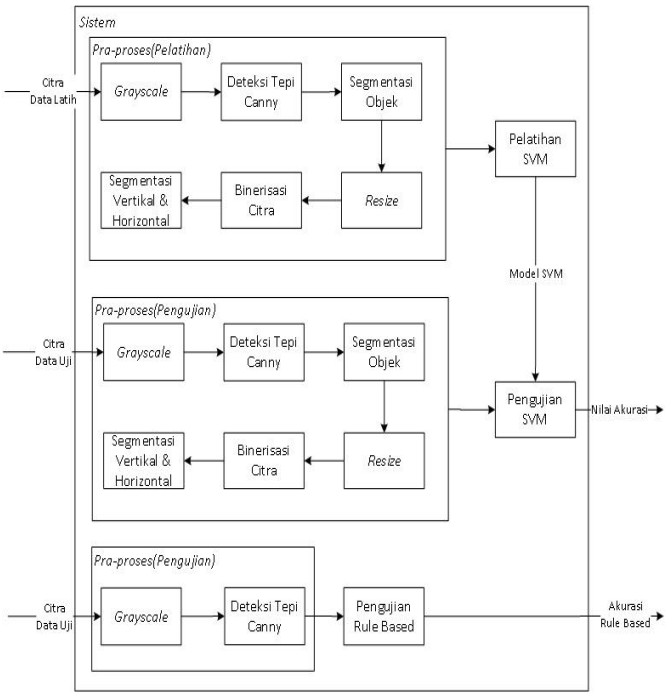


Figure 11 the general architecture of the system

In the block diagram above, the initial process is done that is entering the signature image into the system. The next step through the image processing stage includes grayscale, canny edge detection, object segmentation, resizing, image binerization and feature extraction with vertical & horizontal segmentation. The result of image processing will get the matrix value of each input image, then processed back in classification phase by SVM method both training and test. Similarly with rule-based but not through the training phase. The result of the testing phase is the accuracy of the pattern recognition classification on the signature.

K. *SIGNATURE*

The input data to be processed *Support vector machine* is a scanned image of the signature that is stored in the form .jpg file. In this analysis the amount of data taken from correspondents amounts to 100 different signature data consisting of each of 50 signature data for each feature. The following data are examples of data that have been collected from correspondents like figure 12 below.

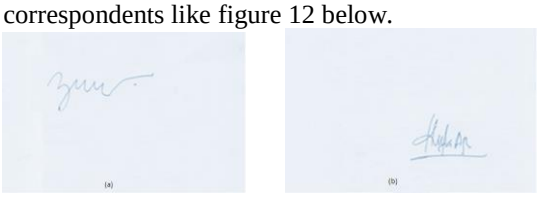


FIGURE 12 IMAGE INPUT

Figure 12 above has a size of 1000x800 pixels. Next, for example, we use the image (b) for the system input image shown in Figure 3.3 below.

L. SIGNATURE IMAGE PROCESSING

In the image processing done some process so that the input image can be used in classification stage using *Support vector machine* (SVM) method. Here is a translation of the image processing process done in accordance with the system architecture created.

1. Grayscale

   Grayscale image process. Grayscale is an early stage done in the image processing stage. The result of this process will produce an image with a color range from 0 to 255 which is a gray scale.

2. Canny edge detection

   The end result of this process is a black background image that has an edge value detected as an object in the image.

3. Segmentation

   In the process of segmentation, the image must have been through the edge detection process. This is done so that the system is able to cut off the signature objects of different sizes.

4. Binary Image

   The next step in image processing is thresholding. This process is done on the grayscale image. Threshold method used in this research is simple threshold. The end result of this process is a black and white image.

5. Resize

   The image of the segmentation, of course, the size is still varied. Therefore it needs to be uniformed into a

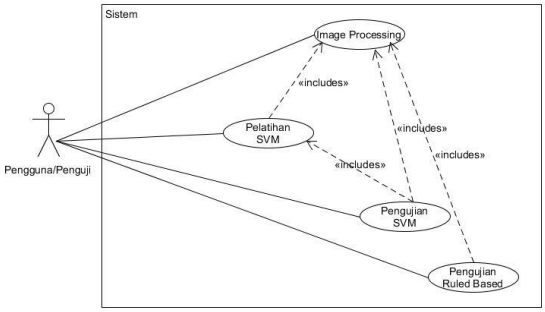
certain size so that the system can consistently read the image. In this study the image is resized to a predetermined size in each feature.

6. Feature Extraction (vertical & horizontal segmentation)

The result of this process is a series of binary numbers in each feature area that will be used as input data in classification stage using *Support vector machine* (SVM) method.

M. *USECASE DIAGRAM*

Use case diagram is a model that shows the interaction between the system with the user (actor). Here is a use case diagram for built applications.



V. TESTING ACCURACY

In the accuracy test will be calculated the value of accuracy or suitability of the application results of recognition of signature patterns in detecting personality. In this study testing the parameters of training used the same and the resulting accuracy of svm testing and ruled ie as follows.

Type of Pattern Signature	Number of Data Testing	Result of Classification		Accuracy SVM
		True	False	
Shell	48	20	28	61%
Kurva Awal	49	27	22	55%
Final Grafiti	48	24	24	50%
Scribble Medium	48	43	5	89,5%
Underline	49	33	16	67,3%
Average Acuraccy				60,7%

Type of Pattern Signature	Number of Data Testing	Result of Classification		Accuracy *Rulebased*
		True	False	
Margin Ekstrim (*x*)	50	35	15	70%
Margin Ekstrim (*y*)	50	30	20	60%
Structure dot	50	22	28	44%
Separate lines	50	17	33	34%
Separate Signature	50	28	22	56%
Average Acuraccy				52,8%

VI. RESULT

Based on the results of the research with 50 data train and 50 test data in the form of the signature image yields the highest SVM accuracy of 89% and the lowest by 41% with an average accuracy of 60.7%. Tests using the *rulebased* method with 50 test data yielded the highest accuracy of 70% and the lowest 34% with an average accuracy of 52.8%. Here is an explanation of the cause of the accuracy of the resulting system not so good accuracy.

1. The training data taken by the pattern is uniformly selected while the data being tested is taken too varied or varied makes the accuracy value low.
2. The feature extraction used to extract the information is still not good in representing the information as a pattern of the signature so as to degrade the quality of the classification. So it can be concluded that the SVM method is actually suitable for use in personality detection systems based on signature patterns, but by improving the process of extraction of good features and the increasingly diverse pattern of practice data to better calibrate and add signature features that do not yet exist on the system.

VII. SUGGESTION

Based on the test results that have been done, then the expected advice is as follows.

1. Add or replace the feature extraction process such as PCA (principal compenent analysit) method to reduce matrix and keep the image pattern information well taken.
2. Using different kernels on the SVM method as a comparison material such as the RBF kernel, polynomials etc.
3. Added features that do not exist in this study to complete the personality detection system based on the signature pattern.
4. Add more patterned pattern practice data in order to detect different types of signature patterns.

REFERENCES

[1] Abhicandra, *Panduan Lengkap Grafologi*. Yogyakarta: Araska, 2017.

[2] E. C. Djamal, R. Darmawati, and S. N. Ramdlan, "Application image processing to predict personality based on structure of handwriting and signature," *Proceeding - 2013 Int. Conf. Comput. Control. Informatics Its Appl. "Recent Challenges Comput. Control Informatics", IC3INA 2013*, pp. 163–168, 2013.

[3] K. N. Lakshmi, A. Keerthana, and P. R. Lakshmi, "Handwriting Analysis Based Human Personality Prediction," vol. 8, no. 5, pp. 105–110, 2017.

[4] S. Prasad, ivek K. Singh, and A. Sapre "Handwriting Analysis based on Segmentation Method for Prediction of Human Personality using *Support vector machine*," *Int. J. Comput. Appl.*, vol. 8, no. 12, pp. 25–29, 2010.

[5] R. Kumar, P. Singhal, and E. C. E. Vlsi, "Review on Offline Signature Verification by SVM," *Int. Res. J. Eng. Technol.*, vol. 4, no. 6, pp. 1771–1773, 2017.

[6] V. Iranmanesh *et al.*, "Online handwritten signature verification using neural network classifier based on principal component analysis.,"

ScientificWorldJournal., vol. 2014, p. 381469, 2014.
[7] J. Hu, Z. Guo, Z. Fan, and Y. Chen, *Offline signature verification using local features and decision trees*, vol. 31, no. 3. 2017.
[8] E. C. Djamal, S. N. Ramdlan, J. Informatika, T. Jenderal, and S. Cimahi, "Pengenalan Pattern Signature Menggunakan MLP Dalam Identifikasi Kepribadian," pp. 2–4, 2013.
[9] Thoyib, "Pengenalan Pattern Signature Menggunakan Learning Vector Quantization Dalam Identifikasi Kepribadian," SKRIPSI UNIKOM, 2015.
[10] Sugiyono, *Metode Penelitian Kuantitatif, Kualitatif Dan R&D*. 2012.
[11] P. Roger, *Sofware Engineering*, 5th ed. 2001.
[12] D. S. Prasetyono, *Buku Lengkap Grafologi*, 1st ed. Yogyakarta: DIVA Press, 2016.
[13] R. M. Ramdani, *Mengenali Diri & Potensi Kecerdasan dengan Grafologi*, 1st ed. Bandung: CV NUANSA AULIA, 2016.
[14] D. Putra, *Pengolahan Image Digital*. Yogyakarta: Andi, 2010.
[15] S. R. Sternberg, "Grayscale Morphology.," *Comput. vision, Graph. image Process.*, vol. 35, no. 3, pp. 333–355, 1986.
[16] K.Sembiring, "Penerapan Teknik *Support vector machine* untuk Pendeteksian Intrusi pada Jaringan Lembar Pengesahan Program Studi Teknik Informatika Penerapan Teknik *Support vector machine* untuk Pendeteksian Intrusi pada Jaringan," 2007.
[17] Budi Raharjo, *Mudah Belajar Python untuk Aplikasi Desktop dan Web*. Bandung: INFORMATIKA BANDUNG, 2015.
[18] "Library OpenCV." [Online]. Available: https://opencv.org/about.html. [Accessed: 23-Oct-2017].
[19] B. K. Hamilton and R. Miles, *Learning UML 2.0*. O'Reilly.