IMPLEMENTATION OF SVM AND GABOR FILTER METHOD ON MOBILE APPLICATION IMAGE RECOGNITION FOR MAKING TEAK WOOD FURNITURE (A Case Study of PD Amoy Jaya Furniture)

Fachmi Miftahillah[1], Irfan Maliki[2]

[1,2] Teknik Informatika – Universitas Komputer Indonesia
Jl. Dipatiukur 112 – 114 Bandung
E-mail : fachmimiftahillah11@gmail.com[1], irfanmaliki007@gmail.com[2]

ABSTRACT

Wood is one of the raw construction materials widely used for a variety of purposes, such as building materials, bridges, craft items and as well as a material in furniture making process. PD Amoy Jaya Furniture is an industry which is use wood as construction material in furniture making process. The kind of wood is used is teak. Teak wood is generally used because it has different characteristics. The lacks of knowledges of teak wood from consumers are also increasingly difficult to choose whether the raw material is appropriately used for furniture making process.

Gabor filter is an object recognition technique to display a special texture characteristic of the image that has been convoluted by kernel. *Support Vector Machine* (SVM) is a supervised classification technique because when the training process is required specific learning targets. In an image processing, the texture can be used for classification on the image which is to identify and recognize an image with a certain texture based on a set of classes or texture classifications that has given.

Based on the result of the implementation of the mobile application teak wood recognition that is using *support vector machine* and *gabor filter*, the accuracy of recognition result is 92%. And the results of a questionnaire survey showed the percentage of 90%. Which mean that PD Amoy Jaya furniture benefits by the mobile application in recognition the teak wood of furniture making process.

Keywords: Wood, *Gabor filter*, *Support Vector Machine*, Classification**,** Mobile

1. PRELIMINARY

Wood is one of the raw construction materials widely used for a variety of purposes, such as building materials, bridges, craft items and as well as a material in furniture making process. PD Amoy Jaya Furniture is an industry which is use wood as construction material in furniture making process. The kind of wood is used is teak. With the age level of a wood, will produce different furniture quality. Many consumers of the company which makes furniture from teak because it has different characteristics, among others : the power level of the wood, durability of the wood, the beauty of wood, and easiness of construction [1]. Based on the interview result to Mr. Hasanudin Jafar as the owner of the company, lack of knowledge of teak wood from consumers are also increasingly difficult to choose whether the raw material is appropriately based on the age. Therefore, image recognition of teak wood application very needed for consumers in PD Amoy Jaya Furniture, so that it can help in the manufacture of furniture.

Image is the image on two-dimensional field, then an image a spatial dimension or field which contains the color information that is not dependent on time [2]. Texture is one of the basic elements of the image. These elements form the basis of characteristics or traits that may be in the image and form a pattern with certain intervals and direction repeatedly that meet most or all areas of the image. In image processing, texture can be used for classification of the image namely to identify and recognizing an image with a certain texture based on a set of classes or texture classification which are given. Each area on image the texture have unique characteristics. To get the texture characteristics, a useful feature extraction required to take different characteristics of each region in the image texture.

By using mobile technology as the basis for image recognition, recognition process will be easier and more efficient [3], thus accelerate in generating information of the image to be recognized, These days, mobile technology can be used by everyone, especially the consumer and the owner of the company PD Amoy Jaya, thus assist in providing information to consumers in choosing teak wood for making furniture according teak wood age.

Based on the research [4] by using Gabor filter produces an accuracy of 79.31%. And the results of research [5], the classification using SVM method produces an accuracy of 90.67%. From the research and background above, the authors of this research will implementation Support Vector Machine and Gabor Filter on mobile application for teak wood image recognition.

1.1 Purpose and Objectives

The purpose of this research is to create a mobile-based applications by implements the methods Gabor Filter and Support Vector Machine (SVM) in the selection of teak wood and age.

The purpose of this research is to assist PD Meubel Amoy Jaya consumers in choosing teak wood and age for making furniture.

2. RESEARCH CONTENT

2.1 Image

Image is the image on two-dimensional field, then an image a spatial dimension or field which contains the color information that is not dependent on time. Digital image is an image consisting of pixels, where each pixel representing color (gray level for an image black and white) at a point in the image [2]. In general, digital image processing represent 2-dimensional image processing using a computer. Digital image is an array that contains real value or complex which is represented by a specific bit stream. An image can be defined as a function f (x, y) sized M rows and N columns, by x and y are the spatial coordinates, and the amplitude of f at the point of coordinates (x, y) is called intensity or gray level of the image at the coordinates. If the value of x, y and overall amplitude value f finite and has a discrete value, it can be said that the image is a digital image.

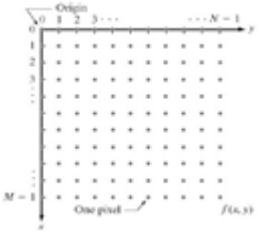


Figure 1 Digital Image Coordinates

2.2 Image Resolution

The image resolution is the level of image detail. Higher resolution images will enhance image detail. Measurement of image resolution can be either physical size (number of lines per picture height). Resolution images can be measured in many ways:

1. Pixel Resolution
   Pixel resolution is a calculation the number of pixels in a digital image. An image with high N and M pixel width means it has a resolution of M x N. Pixel resolution will give two integer numbers sequentially will represent the number of pixel width and height on the image.
2. Spatial Resolution
   The spatial resolution indicates how close each line on images. The distance depends on the system that creates the image. The spatial resolution produce the number of pixels per unit length. The spatial resolution generates the number of pixels per unit length.
3. Spectral Resolution
   A digital image intensity differentiate into multiple spectrum. Multispectral imagery will provide a spectrum or wavelengths are better used for color displays.
4. Temporal Resolution
   Temporal resolution related with the video. A video is a collection of static frames that form the image displayed sequentially and quickly. Temporal resolution gives the number of frames that can be displayed per unit of frames per second (fps).

2.3 Image Grayscale

Grayscale is the color pixels that are within the range of shades of black and white. Intensity grayscale image data is usually stored as 8 bit or 256 color intensity gray from 0 (black) to 255 (white). Grayscale image is a digital image that has only one channel on each pixel value, in other words the value of RED = GREEN = BLUE. This value is used to indicate the level of intensity. The color is black, gray, and white. Gray level is gray color with various levels from black to white color. This color model contains 8 bits of color information. Thus, the colors that can be displayed in grayscale color mode with total 256 gray level. Scale numerical calculations can approach brightness value scale of 0 to 255 as in figure 2 below:

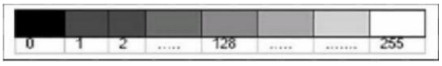


Figure 2 Composition Gray Color

To convert color images into grayscale, can use the formula 2.1:

$$G = 0.3.R + 0.59.G + 0.11.B \quad (1)$$

Where :
G = Greyscale Image
R = Red
G = Green
B = Blue
R = red value * 15 / 255
G = green value * 15 / 255
B = blue value * 15 / 255
Examples from results grayscaling process shown in Figure 3 below:



Figure 3 Examples of Grayscale Image

2.4 Gabor Filter Extraction Method

Gabor filter method is an object recognition method that aims to bring out the special characteristics of the image that has been dikonvolusi to the kernel. As the filter is used Gabor 2D filter kernel obtained by modulating the 2D sine

wave at a frequency and a particular orientation with Gaussian envelope. Basic equation function 2D Gabor filter shown in equation (2), where σ is the standard deviation of the Gaussian envelope dimensions x and y. Lamda (λ) and tetha (θ) is the length and orientation of the 2-D sine wave. The spread of the Gaussian envelope is defined in the form of a sine wave λ. Rotation of the x - y by angle θ generate Gabor filter at orientation θ [20].

$$g(x, y; \lambda, \theta, \varphi, \sigma, \gamma) = \exp\left(-\frac{x'^2+\gamma^2 y'^2}{2\sigma^2}\right) \cos\left(2\pi \frac{x'}{\lambda} + \psi\right) \quad (2)$$

Where :

$x' = x \cos\theta + y \sin\theta$

$y' = -x \sin\theta + y \cos\theta$

σ = standard deviation of *Gaussian envelope*

λ = wavelength *sinus* on 2 dimension

θ = the degree of orientation in the 2-dimensional sine wave

γ = aspect ratio spatial

ψ = *phase offset*

Phi (ψ) has a value of between -180 degrees, -90, 0, 90, 180. The value of gamma (γ) of 0.5. Sigma (σ) does not have a value which can be set, but it can be replaced with a bandwidth value (b), namely 1. Wavelength (λ) and the degree of orientation (θ) differ applied to a certain point (x, y), then got a lot of response filter to the point, for example: use four wavelengths and eight orientation, it will generated 32 response filter for each image point will be convoluted with the filter.

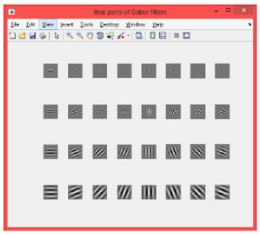


Figure 1 Filter Gabor

2.5 *Support Vector Machines* (SVM) Method

Support Vector Machines (SVM) is a classification of types of integrated methods (supervised) because when the training process, required a certain learning targets. SVM is also a working algorithm using nonlinear mappings to change the original training data to a higher dimension. In this case a new dimension, will seek hyperplane to separate linear and nonlinear mapping the right to a dimension that is high enough, the data from the two classes can always be separated by the hyperplane. SVM find this hyperplane using support vector and margin [6].

SVM first appeared in 1992 by Vladimir Vapnik and colleagues Bernhard Boser and Isabelle Guyon. The basis for the SVM has been around since the 1960s (including early work by Vapnik and Alexei Chervonenkis on statistical learning theory).

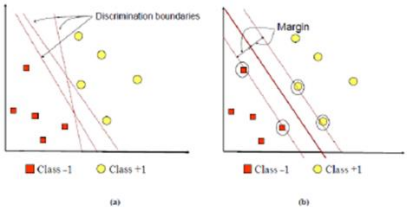


Figure 5 SVM trying to find the best hyperplane which separates the two class -1 and +1

Hyperplane best separator between the two classes can be found by measuring the hyperplane margin and look for maximum points. Margin is the distance between the hyperplane to the nearest pattern of each class. The closest pattern is called a support vector. Solid line in Figure 5-b shows the best hyperplane, which are located right in the middle of the second class, while the red dots and yellow which are in black circle is a support vector. Attempt to find the location of this hyperplane is the core of the learning process on SVM [6].

SVM concepts can be explained simply as an attempt to find the best hyperplane which serves as a separator are two classes in the input space. Figure 5-a shows some patterns that are members of two classes: +1 and -1. Pattern belonging to the class -1 symbolized by the color red (box), while the pattern in the class +1, symbolized by a yellow color (circle). Problem classification can be translated to the business of finding a line (hyperplane) that separates between the two groups. In linear Support Vector Machine separator is a linear function. Training data is expressed by (x_i, y_i) and x_i = { x_1, x_2, ... x_{iq}} is an attribute (feature) class training data set for all i . For yi € {-1,1} stated class label. Defining the equation of a hyperplane dividing written by:

$$w * x_i + b = 0 \quad (3)$$

Data xi are divided into two classes, which belongs to the class -1 (negative sample) is defined as a vector that satisfies the inequality (4) below :

$$w * x_i + b < 0 \text{ untuk } y_i = -1 \quad (4)$$

While belonging to a class +1 (positive samples) satisfies the inequality (5) below :

$$w * x_i + b >= 0 \text{ untuk } y_i = +1 \quad (5)$$

Where :

xi = data input

yi = label given

w = the value from normal field

b = position relative field at coordinates

The parameters w and b are parameters that will be sought. When the data label yi = -1, then the barrier become the equation (6) the following:

$$w * x_i + b \leq -1 \quad (6)$$

When the data label yi = +1, then the barrier becomes the equation (7) below:

$$w * x_i + b \geq +1 \quad (7)$$

The margin can be searched by maximizing the distance between the two classes and field barrier closest point, i.e. 2/| w |. It is formulated as a

problem of quadratic programming (QP) is looking for the minimum point of equation (8) with regard equation (9) below:

$$min\ \tau(w) = 1/2\|w\|^2 \quad (8)$$

$$yi(w * xi + b) - 1 \geq 0,\ (i=1,..,n) \quad (9)$$

This problem can be solved by various computational techniques. More easily solved by changing the equation (8) into the Lagrangian function in equation (10), and simplifying to the equation (11) below:

$$L(w,b,a) = 1/2\|w\|^2 - \sum_{i=1}^{n} a_i(y_i((w^T x_i + b) - 1)) \quad (10)$$

$$L(w,b,a) = 1/2\|w\|^2 - \sum_{i=1}^{n} a_i y_i((w^T x_i + b) - 1)) + \sum_{i=1}^{n} a_i \quad (11)$$

Where ai lagrange multiplier whose value is zero or positive ($ai \geq 0$). The optimal value from equation (11) can be calculated by minimizing L against w, b and a. Can be seen in equation (12) to (14) below:

$$\frac{\partial L}{\partial w} = w - \sum_{i=1}^{n} a_i y_i x_i = 0 \quad (12)$$

$$\frac{\partial L}{\partial b} = \sum_{i=1}^{n} a_i y_i = 0 \quad (13)$$

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

Then the Lagrange problem for classification can be declared at equation (15) below:

$$Min\ L(w,b,a) = 1/2\|w\|^2 - \sum_{i=1}^{n} a_i y_i (w^T x_i + b) - \sum_{i=1}^{n} a_i \quad (15)$$

Having regard to the equation (16) and (17) the following:

$$w - \sum_{i=1}^{n} a_i y_i x_i = 0 \quad (16)$$

$$\sum_{i=1}^{n} a_i y_i = 0 \quad (17)$$

Model equation (15) above is the primal model of Lagrange. Meanwhile by maximizing L to ai, becomes the equation (18) below:

$$Max \sum_{i=1}^{n} a_i a_j y_i y_j^T x_i x_j^T \quad (18)$$

Having regard to the equation (19) below:

$$\sum_{i=1}^{n} a_i y_i = 0, a_i \geq 0\ (i, j = 1, \dots, n) \quad (19)$$

To find the value xi and yi, can be done when the value from the weighting of feature extraction and class initialization has been obtained. Results from the extraction weighting is converted into data format svm, while the class data becomes the data label svm. To get the value ai, the first step is to transform each abstract vector value (*support vector*) = $\begin{pmatrix} x \\ y \end{pmatrix}$. Then the value from each weight vector extraction is inserted into the equation (20) phi kernel trick, below:

$$\phi\begin{bmatrix} x \\ y \end{bmatrix} = \begin{cases} \sqrt{x_n^2 + y_n^2} > 2\ maka \begin{bmatrix} \sqrt{x_n^2 + y_n^2} - x + |x - y| \\ \sqrt{x_n^2 + y_n^2} - y + |x - y| \end{bmatrix} \\ \sqrt{x_n^2 + y_n^3} \leq 2\ maka \begin{bmatrix} x \\ y \end{bmatrix} \end{cases} \quad (20)$$

The value of x is obtained from the equation (21) linear kernel for x below:

$$\sum_{i=1,j=1}^{n} x_i x_j^T, (i, j = 1, \dots, n) \quad (21)$$

The value of y is obtained from equation (22) linear kernel for y the following:

$$\sum_{i=1,j=1}^{1} y_i y_j^T, (i, j = 1, \dots, n) \quad (22)$$

To get optimum distance perpendicular to consider the positive vector, the final result from the substitution of the value of x and y to the equation (20) rated bias = 1. Then, find the parameter ai to find the value of each abstract function using equation (23), then look for the value ai at linear equations using equation (24) by observing i, j = 1, .., n below:

$$\sum_{i=1,j=1}^{n} a_i S_i^T S_j \quad (23)$$

$$\sum_{i=1,j=1}^{n} a_i S_i^T S_j = y_i \quad (24)$$

Once the parameters ai obtained, then enter into the equation (25) below:

$$W = \sum_{i=1,j=1}^{n} a_i S_i \quad (25)$$

Results obtained using equation (25), then use equation (26) to get the value w and b.

$$y = wx + b \quad (26)$$

then obtained value w and b or value hyperplane to classify both classes.

2.6 Mobile

2.6.1 Mobile Android

Android is an operating system for Linux-based mobile devices that includes an operating system, middleware. Android provides an open platform for developers to create their own applications that can be used by a variety of device driver [3].

Based on the opinion by Nazruddin exposure and Firdan can be concluded, Android is a mobile operating system based on Linux that is open source, so it gives developers to develop as expected. Thus, the very fast development of this operating system.

2.6.2 Android Architecture

Broadly speaking, the Android architecture has five layers, namely, *Application and Widgets*, *Application Frameworks*, *Libraries*, *Android Run Time*, and Linux *Kernel* (Nazruddin, 2012: 6).

1. *Application and Widgets*
   Application and Widgets is the layer where we relate to the any application, where usually we download the app and then install and run the application.
2. *Application Frameworks*
   Application Frameworks is the layer where application developer do the development / creation of applications that will run on the Android operating system, because in this layer, the application can be designed and manufactured.
3. *Libraries*
   Libraries is the layer where features of android, application developers typically access the libraries to run applications.
4. *Android Run Time*
   Layer that makes the android application can be run, where the implementation process are using linux OS.
5. Linux *Kernel*
   Linux *kernel* is the layer where the core from the android operating system is located. Contain system files that manage the system processing, memory, resource, drivers, and other android operating system.

2.6.3 Android Advantages

The advantages of android as follows:

1. User Friendly
   This feature is very attached to Microsoft's Windows operating system, It was as we very easily operate a computer just by learning a few days or even a few hours, and this also applies to Android that runs on the Smartphone.
2. Appearance
   In terms of appearance, Android is no less good than iOS Apple, because from the beginning, android almost using the same technology as IOS, only in a cheap version.
3. Application
   For an application, user will be presented with millions of apps, from a free, to paid applications, which can be downloaded on Google Play.

2.6.4 Android Disadvantages

Disadvantages of android is as follows:

1. System Update
   To update the system is not easy, had to wait from each vendor to release the latest version of the update.
2. Battery Problem
   This often occurs when using the data packets and using the widgets with applications that run excessively, to fix this, users should reduce the time of application activity on smartphones.
3. Lag
   This deficiency relates to the specifications from each devices, but there were times when android was not friendly with some applications, due to insufficient of RAM or Processor.

2.7 Image Pre-Processing Analysis

2.7.2 Image Resize

Resize is the phase where the size of the input image scaled down from its original size, this phase aims to get the same input from all of the input images. Because the input images that will be tested has a large size, which is 1200 x 1600 pixels, resize will change the image size to 200 x 200 pixels.

2.7.3 *Grayscaling*

Stages Stage grayscaling image is to simplify the model image. The three layers of color images, the R, G, and B is converted to a grayscale layer. To change the color image, which has a value matrix, each R, G, and B, into a grayscale image, by dividing the period of the third layer, with equation (1).

2.8 Gabor Filter Method Analysis

Basic equation function 2D Gabor filter shown in equation (2), where σ = 1 is the standard deviation of Gaussian Envelope, on dimensions x and y. λ = 2 and $\theta = 90^0$ is the length and orientation of the sine wave 2D, π = 3.14, $\gamma = 0{,}5$, and $\psi = 0$ because it has a value degree of -180,-90,0,90,180.

After performing the calculation process, the result values in the image feature extraction of teak with 4 x 4 pixels as shown below.

(x,y)	2	3	4	5
45	-0.77	-0.77	-0.48	-0.32
46	-0.41	-0.29	-0,2	-0.38
47	-0.41	-0.5	-0.52	-0.57
48	-0.32	-0.27	-0.48	-0.29

After getting the results of the image feature extraction, then performed a search of the weight values of the image characteristics by finding the mean and standard deviation values.

a. *Mean*

$$\mu = \frac{\sum x_i}{n} \quad (27)$$

Where :
μ = mean
n = amount of data
$\sum x_i$ = the numbers of data values
= -0.77 + (-0.77) + (-0.48) + (-0.32) + (-0.41) + (-0.29) + (-0.2) + (-0.38) + (-0.41) + (-0.5) + (-0.52) + (-0.57) + (-0.32) + (-0.27) + (-0.48) + (-0.29) / 16 = -0,43625

b. Standard deviation

$$s = \sqrt{\frac{\sum_{i=1}^{n}(x_i-\mu)^2}{n-1}} \quad (28)$$

Where :
s = standard deviation
x_i = data value
n = amount of data

As known :
μ = -0,43625
Questioned *s ?*
Calculation :

Table 1 Standard Deviation Calculation

x_i	$x_i - \mu$	$(x_i - \mu)^2$
-0.77	-0.33375	0.111389063
-0.41	0.02625	0.000689063
-0.41	0.02625	0.000689063
-0.32	0.11625	0.013514063
-0.77	-0.33375	0.111389063
-0.29	0.14625	0.021389063
-0.5	-0.06375	0.004064063
-0.27	0.16625	0.027639063
-0.48	-0.04375	0.001914063
-0.2	0.23625	0.055814063
-0.52	-0.08375	0.007014063
-0.48	-0.04375	0.001914063
-0.32	0.11625	0.013514063
-0.38	0.05625	0.003164063

-0.57	-0.13375	0.017889063
-0.29	0.14625	0.021389063
$\sum_{i=1}^{n}(x_i-\mu)^2$		0.413375

$$s = \sqrt{\frac{\sum_{i=1}^{n}(x_i-\mu)^2}{n-1}}$$

$$s = \sqrt{\frac{0.413375}{16-1}}$$

$$s = \sqrt{\frac{0.413375}{15}} = 0{,}166007028$$

Thus, the value of the weight vector of the image of teak wood with a size of 4 x 4 pixels, is (-0.43625, 0.166007028).

2.9 Teak Wood Age Analysis

In this research, the age of teak manually observed by looking at the results from the extraction of Gabor filters that can be distinguished based on the results of the mean weight value. Where the mean value is the result from the sum of the value extraction Gabor filter, so that the value can represent the characteristic values of an image. This research, the age of teak wood used is aged <6 years and aged > 6 years. Teak with aged < 6 years old, has a mean value < 0.4 while teak wood with aged > 6 years have mean values> 0.4.

2.10 SVM Training Phase

In the training phase aims to determine the value of the vector a, w and b values to get the best hyperplane. After getting the values characteristic of the entire teak image pixel, on this training process the data will be converted into the form of vector data svm. In this research for data representation format input SVM classification uses the format [*kelas urutan_bobot$_1$:bobot$_1$ urutan_bobot$_n$:bobot$_n$*] [30]. The first data are given initialization label +1 or -1, which is expressed as two initial label given. Table 2 below is an example of changing the characteristic value of the image into a vector.

Table 2 Example of Changing the Characteristic Value of the Image Into Vector

Image Name / Class	Train1.jpg / Kayu Jati
Weight Value of Image	[0.4794115724185327, 0.31935630333301923]
Vector	[1 1:0.4794115724185327, 2:0.31935630333301923]

From the example of Table 2 where the number 1 on the first character states that data entered into the positive label. Then the next number 1 is an index of the n vector, the next number is 0.4794115724185327 weight value. This process will be repeated until all the weight values on training images converted into the vector. The flowchart of the SVM training can be seen in Figure 6 below.

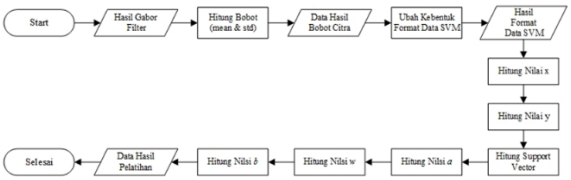


Figure 6 Flowchart SVM Training

In this research, using some of the functions that exist in LibSVM. The first process in this LibSVM is giving mzodel on training data in the form of the model label +1 and -1.

The accuracy of the model that will be produced from the SVM training process is very dependent on the kernel functions and parameters used. Therefore, its performance can be optimized by searching (estimating) the best parameters.

After the training data is calculated, using the parameters that has been determined, then stored in svm_model class function, the which will be used in the classification that generate predictions.

2.11 SVM Testing Phase

After getting the value w and hyperplane on training data, the data can then be determined that will be tested include the +1 or -1 to class by performing the same calculation that is from the preprocessing stage to get weight value of the image then, look for the value of vector in an image that will be tested. Figure 7 below is a flowchart for generate of testing on svm classification.

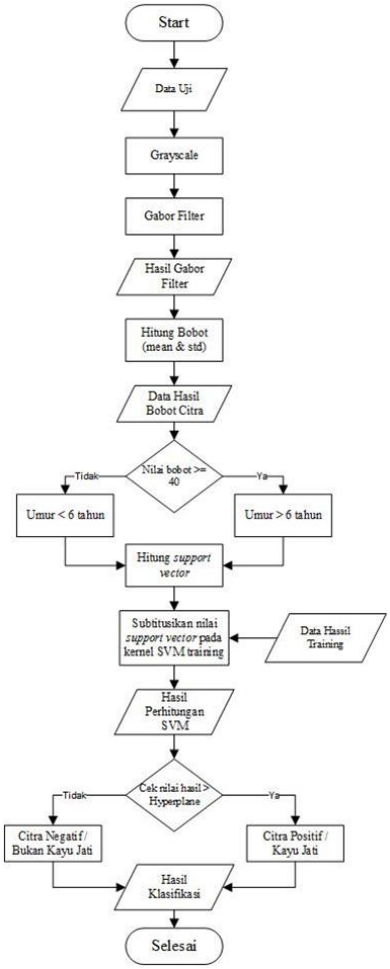


Figure 7 Flowchart SVM Classification

In LibSVM, after saving the results of calculation parameter SVM on svm_model, The next is to calculate the prediction of svm library method, namely svm_predict_probability using the method parameters svm_model that have been processed previously. After the prediction calculation process is completed, it will get the results of image classification which included in class 1 or -1.

2.12 Process Flow Description

General description on the flow of image recognition process is started from the stage of image acquisition to the end namely classifies the image. In figure 8 below is the stage of the process to be performed.

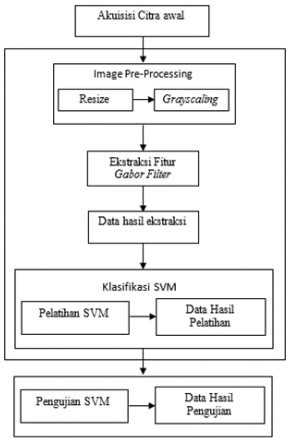


Figure 8 Image Recognition Process Flow of Teak Wood

2.13 Implementation

2.13.2 Interface Implementation

Interface implementation describes and illustrates each display, which is built on this application.

2.13.2.1 Interface Implementation of Main Menu

The main menu is the first screen when the application is run. In this interface there are some icons, including the camera icon for taking pictures of test image through the camera, and a menu slides that contain multiple menus for teak wood recognition process.

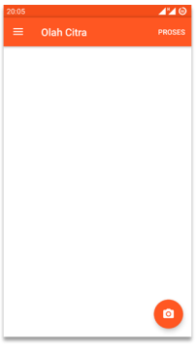


Figure 9 Main Menu Interface

2.13.2.2 Interface Implementation of Slide Menu

Slide menu is an interface which contains functions for processing the train and test images.

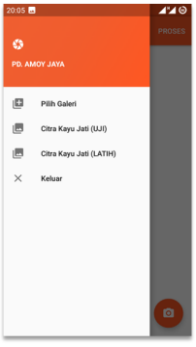


Figure 10 Slide Menu Interface

2.13.2.3 Interface Implementation of Training Menu

Training Menu interface contains the images of teak wood which is used as the training image. In this menu, there is function for adding the training images to the application, which can be done through the media gallery or camera.



Figure 11 Interface Menu of Image Training

2.13.2.4 Interface Implementation of Testing Menu

Testing Menu is an interface to add test image into the application through camera or gallery, which will be tested.

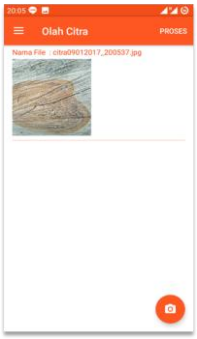


Figure 12 Interface Menu of Image Testing

2.13.2.5 Implementation Interface of Classification Menu

This interface is a menu that provides information on the recognition process of test images with image training. In this menu there are information about the age of the recognized teak, wood type, and usability of the teak wood.



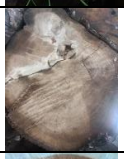
Figure 13 Interface Menu of Classification

2.14 Testing

2.14.2 Testing Recognition Image of Teak Wood

To demonstrate the accuracy of teak, the test is applied to the test image corresponding (teak) with the image training, and the test image that does not accordance with the image training (not teak), test images inputted via the camera or gallery by observing the following tables:

Table 5 Testing Results Figure Test Teak Wood

No	Image Testing	Image Training	Image Description	Result
1			Teak wood Age > 6 Years	Succeed
2			Teak wood Age > 6 Years	Succeed
3			Teak wood Age < 6 Years	Succeed
4			Teak wood Age < 6 Years	Succeed
5		Nothing	Apart from teak wood	Failed
6		Nothing	Apart from teak wood	Failed
7			Teak wood Age < 6 Years	Succeed
8			Teak wood Age < 6 Years	Succeed
9			Teak wood Age > 6 Years	Succeed

No	Image Testing	Image Training	Image Description	Result
10			Teak wood Age < 6 Years	Succeed
11			Teak wood Age < 6 Years	Succeed
12			Teak wood Age < 6 Years	Succeed
13			Teak wood Age < 6 Years	Succeed
14			Teak wood Age < 6 Years	Succeed
15			Teak wood Age < 6 Years	Succeed

Based on testing of 15 test images are inputted, the level of recognition accuracy obtained at 86.66%, with the recognizable image of 13 units as teak.

Table 6 Testing Results Figure Test Not Teak Wood

No	Image Testing	Image Training	Image Description	Result
1		Nothing	Apart from teak wood	Succeed
2		Nothing	Apart from teak wood	Succeed
3		Nothing	Apart from teak wood	Succeed
4		Nothing	Apart from teak wood	Succeed
5		Nothing	Apart from teak wood	Succeed
6		Nothing	Apart from teak wood	Succeed
7		Nothing	Apart from teak wood	Succeed
8		Nothing	Apart from teak wood	Succeed
9		Nothing	Apart from teak wood	Succeed
10		Nothing	Apart from teak wood	Succeed

Based on tests performed on the test image in addition to teak, showed that the test image is not a teak wood, which means that the application can recognize 100% that the input test image does not fit with the existing training images.

2.14.3 The Conclusion of Testing

Based on the results of testing that has been done, for the functionality of the application gives the expected output. However, for the background images contained test must first be cropping with a minimum of 200 x 200 pixels before the input. From the research, the percentage of success can be calculated using the general formula = (P / N) * 100%, where P is the number of images on a single image category intended, while the n is the total number of input images. Thus, the overall results can be obtained from the test image recognition of teak wood, of: (23/25) * 100% = 92%.

3 CLOSING

3.1 Conclusion

Based on the research that has been done, it can be concluded for results the implementation of the method Support Vector Machines (SVM) and extraction Gabor Filter on application for teak wood image recognition, the accuracy of recognition result is 92%. With good accuracy and application-based mobile, thus, can help consumers PD Amoy Jaya Furniture in choosing teak wood according to age to make furniture.

3.2 Advice

Based on the results of the implementation and testing support vector machines and extraction methods Gabor filter on application teak wood image recognition for making furniture, There are some opinions that are expected to be developed in subsequent research, namely:

1. For the method of support vector machines must be a lot of training data, because this method is a method of classification when recognition process is required learning targets. Thus, the more data training is entered will further increase the level of accuracy.
2. Needs to be redeveloped in image processing, especially using mobile in order that processing training data can be stored in the database server or media faster processing, so that the recognition process will be faster rather than the training data processing on the mobile storage.
3. Developing a method other than the feature extraction Gabor filter, in order to identify the image apart from the texture. As from the point of image capture, color, shape, etc.
4. Expected in the next research SVM and Gabor filter method can be used to recognize besides the image of wood.

BIBLIOGRAPHY

[1] Soerianegara, I. dan RHMJ. Lemmens (eds.). 2002. Sumber Daya Nabati Asia Tenggara 5(1): Pohon penghasil kayu perdagangan yang utama. PROSEA – Balai Pustaka. Jakarta. ISBN 979-666-308-2. Hal. 7.

[2] D. Putra, Pengolahan Citra Digital, Yogyakarta: Andi, 2010.

[3] H. Nazruddin Safaat, Android Pemrograman Aplikasi Mobile Smartphone dan Tablet PC Berbasis Android. Edisi Revisi. Bandung : Informatika. 2012.

[4] D. E. Kurniawan, "Rancang Bangun Sistem Pengenalan Wajah Menggunakan Filter Gabor," Universitas Dipenogoro, Semarang, 2012.

[5] S. N. Endah dan M. R. Widyanto, "Klasifikasi Pose Skeleton Pada Manusia dengan Support Vector Machine.

[6] Amalia, A. N. (2016). *Implementasi Support Vector Machine (SVM) Pada Klasifikasi Laporan Skripsi (Studi Kasus Teknik Informatika).* Bandung: Perpustakaan UNIKOM.