GENDER CLASSIFICATION FROM FACIAL IMAGES USING SUPPORT VECTOR MACHINE

[1]IKA CANDRADEWI, [2]BAMBANG NURCAHYO PRASTOWO, [3]DJAENUDDIN LATHIEF

[1,2,3]Universitas Gadjah Mada, Department of Computer Science and Electronics, Indonesia

E-mail: [1]ika.candradewi@ugm.ac.id, [2]prastowo@ugm.ac.id, [3]djaenuddin.lathief@gmail.com

ABSTRACT

The face is a biometric feature of a person. This biometric feature encourages researchers to model it. This unique features modeled and applied in many objects. One of them is used to classified the gender of someone like done in this research. There were four main step, consist of pre-processing using a histogram of equalization, face detection using LBP Haar cascade classifier to get face region, feature extraction, and classification. This extraction used to obtain an unique information named features. The method to get the features named Histogram of Gradient (HOG). After setting the parameters in extraction method we will get many features named HOG features. Meanwhile, The last step is classification using Support Vector Machine (SVM) classify gender form facial images. SVM was chosen because it's one of classification method that robust in classify two group of data. The type of the kernel that used in this research is the linear kernel. This kernel has an advantage in the robustness to classify the big features. Implementation result show that the best achieved when HOG parameters (cell size = 20x20 pixels, block size = 5x5 pixels, block stride = 5x5 pixels, bin = 5 or 7) and SVM parameters (C=1, epsilon = 1e-6 dan iteration = 10.000). The accuracy of this proposed method is 92%.

Keywords: *Gender Classification, Support Vector Machine, Histogram of Oriented Gradient, LBP_Haar Cascade Classified, Face Detection*

1. INTRODUCTION

The human face is a unique biometric feature of a person. There is plenty of information contained on the face. Such information may include skin color, face shape, and attributes attached to the face (like a mustache, beard, glasses, and hairstyle, etc.). In addition, the information on the face can also show the social interaction between a person [1].

The interesting thing about this face and then inspired many researchers to compute and modeling of information that exists on the face. One study using the information on the face pretty much studied is the classification of the human sexes. Applications of this research are widely applied in the field of security systems, biometric authentication [2], see the customer profile [3], digital signage [4], surveillance, human-computer interaction [5], and so on. Various methods and approaches to the classification of sex research have been carried out by many previous investigators, like. [6], the approach appearance-Based Face Average Difference (AAFD) is performed to process low-resolution face images in a database FERET, the image is then processed to obtain the average face value of men and women.

Then acquired gender face mask on each pixel, gender face mask is used to calculate the critical information of the face. Then the important information in the form of pixels that have done the mask is then used as input in classification process human sexes. From this process, the classification results obtained are quite effective and easily implemented in real-time systems. Bissoon and Viriri [7], using Principal Component Analysis (PCA) and Linear Discriminant Analysis (LDA). The use of LDA proven to improve classification accuracy values of human gender. The combined approach of local and global features of the face is applied to the study.

The global feature is a feature based on the appearance and local features is a feature geometry based [8]. Studies of gender classifications generally use the normal facial image (unmasking). Has not been much research on gender classification which uses the image of a human face in the main part only (masking). In fact, people often have difficulty and has a high error rate in identifying the gender of the main parts of the face image [9]. Support Vector Machine (SVM) is a classification method that is sufficiently robust in classifying the data [9]. With the advantages

possessed by the expected SVM classification sexes using a facial image, masking can also produce results accuracy as good as using non-facial image masking.

2.   RESEARCH METHOD

In this study, the face image is derived from the downloads through various internet providers in the face dataset. Face image used in this study consisted of 11 source database that is Cohn Kanade, Face, FaceDB, Essex, Iranian, must-a, Nottingham, Pain, Stirling, Utrecht, and Yale faces. Face images are derived from the database have a file type that is different as * .jpg, * .pgm and * .tiff. But before the facial images used as input will be aligning to * .jpg file type. The image dataset still contains hair and shoulder, we do face detection to get the accurate region of the face only. After that the image used in training and testing stage. Generally, the proposed block diagram is shown in Figure 1.
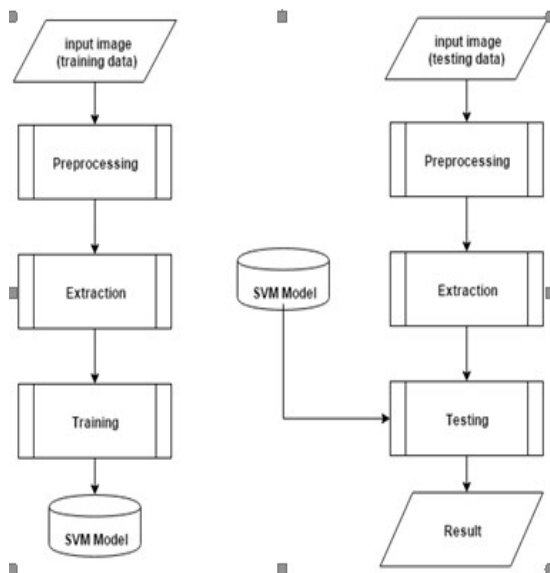

Figure 1: Block Diagram Gender Classification

2.1. Gender Classification

We used an LBP Cascade Classifier to detect face region, for feature extraction method using HOG and SVM as the method of classification. There is four step to perform the process of Gender Classification :

1) Image acquisition process and pre-processing of face detection

2) Feature extraction using Histogram of Oriented Gradient

3) Training data to get SVM model

4) Testing data as evaluation performance of systems based on training result

The number of images used in this study was 600 pieces with the single pose (frontal). The face image is then divided into training data image and the image of the test data. Comparison between the training data and testing is 1: 2. So there are 400 images were used as training data and 200 images as the test data.

2.1.1.   Image Acquisition and Preprocessing

At this stage, LBP cascade classifier is used to localize face region. After that, we crop the image at the region of the face as face detection result. We convert RGB face cropping image into grayscale and to normalize intensity we use histogram equalization. We resize equalize histogram image to 160 x 160. We perform image masking and unmasking image to be evaluated in feature extraction and classification stage

2.1.2.   Feature Extraction

In this paper, HOG feature is used for extraction of facial features for gender classification. HOG features were introduced developed for human detection [10]. This method widely used on computer vision and image processing for object detection. HOG method has produced output descriptor feature. This method calculates occurrences of gradient orientation in localized regions of an image. The main idea behind the HOG descriptor is that display local objects and forms an image can be described using a gradient intensity distribution (distribution of intensity gradients) or toward the edges (edge directions). Calculations are done by considering the direction of the edge of some parameters such as a cell, block, etc. overlap. The cell is a square area in an image consisting of one or more pixels. Block is an area composed of multiple cells. Overlap is a shift in the image block. There are several steps done by HOG in extracting features of an image, among others:

1.   Gradient Computation

The first step is calculating the value of the gradient of an image. The method used is the center point discrete derivative mask 1D well on the field horizontally and vertically. This method requires the filter on grayscale images using filter [-1,0,1] in both vertical dan horizontal direction. So if I am an

image, we obtained derivative s and y using convolution: $I_X = I * D_X$ dan $I_Y = I * D_Y$. The gradient is represented by the magnitude G(x,y) dan gradient orientation θ(x,y) [13].

2. Creating cells

Dividing the image into the small connected region, called cells, for each cell compiling a histogram of gradient directions for the pixels within the cell. The whole image is divided into cells of a square grid whose edges are formed by cellSize pixels.

3. Orientation Binning

The gradient in Cell is histogrammed. Let hi where i=1,2,...........n Bins denote histogram bins and (hr, hq) denote the smallest possible interval where angle θ (x, y) fits. Every pixel with an angle and magnitude votes into histogram bins hr, hq by a linear combination as shown in equation 1 and 2.

$$H(r) = H(r) + \frac{\theta(x,y) - h_q}{h_q - h_r} G(x,y) \qquad (1)$$

$$H(q) = H(q) + \frac{h_q - \theta(x,y)}{h_q - h_r} G(x,y) \qquad (2)$$

$$\text{Where, } h_r \leq \theta(x,y) < h_q$$

4. Block Descriptor

Images are often experienced as a result of the influence of luminance and contrast, it is necessary to normalize the gradient. Gradient normalization is performed on a larger group of the cell block. HOG descriptor is a component of the vector of the normalized histogram of the entire cell block. Implementation of this descriptor can be found by dividing the image into small parts, called a cell, and each cell will draw up a HOG or orientation direction of the edge on the pixels within the cell.

This set of histograms then represents the descriptor. Block is generally experiencing a shift that indicates that each cell produces more than one descriptor. There is two block geometry of the R-HOG HOG and C-HOG. Block R-HOG square shape is shown with 3 pieces of parameters such as a number of cells per block, the number of pixels of each cell, and the number of channels per cell histogram.

5. Block Histogram Normalisation

Normalization block is used to normalize all the histograms in the block. How that is done the same as normalizing the cell. Several methods of normalization in the normalization block indicated by equation (3), (4) and (5).

$$L2 - norm: f = \frac{v}{\sqrt{\|v\|_2^2 + e^2}} \qquad (3)$$

$$L1 - norm: f = \frac{v}{\|v\|_1 + e} \qquad (4)$$

$$L1 - sqrt: f = \sqrt{\frac{v}{\|v\|_1 + e}} \qquad (5)$$

Where v is the vector that has not been normalized histogram containing all within a block, $\|v_k\|$ is k-norm for k = 1.2 and e is a constant whose value is small so it does not affect the results. All block histograms are stacked together into one feature vector. The final step in gender classification using Histogram of oriented gradient is to feed the feature vector into Support vector machine [13].

2.1.3. Training and Testing using Support Vector Machine

Support vector machine is a pattern recognition algorithm which belongs to a group directed learning (supervised-learning), this algorithm originally created for binary classification. Directed learning is a machine learning algorithms that learn through the use of data labeled. A number of labeled data need to be trained to use this algorithm in which each dataset need to be included in a particular class. SVM make one or more hyperplane/separators plane for separating each data class. The classic example for example, in a collection of the 2D point that divides the two classes, the SVM will be looking for the most optimal dividing line between the two classes.

Figure 2 (a) shows an alternative to the use of SVM hyperplane. Based on the above picture can be seen a best alternative interface that separates all the dataset class. Let {x1, x2, x3, ... xn} is data and yi ε {+ 1, -1} is the class label of the data xi. Based on this equation, the search for the best hyperplane can be done by finding a hyperplane that has the largest margin and generalization general. The margin is the distance between the field while limiting generalization common is the ability classifier to classify the test data correctly.

Figure 2 (b) shows the most optimal hyperplane on two classes of data [12]. Before looking for the best hyperplane previous step is to define the field delimiters of two classes. Field delimiter can be compiled from the existence of support vector. Support vector is the data that is in the field delimiter and the position closest to the data of other classes.
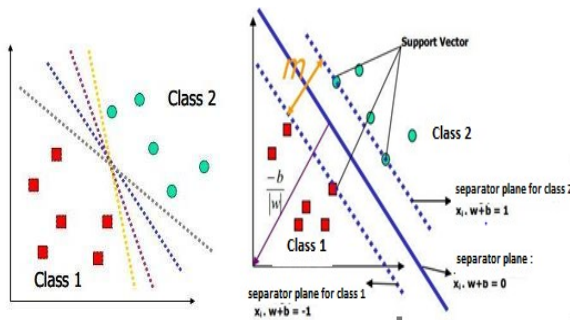
Figure 2. (a) Alternative search best hyperplane in the field of 2D using linear kernel SVM algorithm (b) Best Hyperplane

Based on the example of Figure. 2 (b), there is obtained a boundary field of the first and second classes with the form of equation (6) and equation (7).

$$x_i.w + b \geq +1 \; for \; y_i = +1 \quad (6)$$
$$x_i.w + b \leq -1 \; for \; y_i = -1 \quad (7)$$

With the description, w is the normal field and b is the position of the field relative to the coordinate center. From equation (6) and equation (7), it can be determined that the margin value can be searched by equation (8).

$$\frac{1-b-(-1-b)}{w} = \frac{2}{|w|} \; \frac{1-b-(-1-b)}{w} = \frac{2}{|w|}$$

Equations (6) and (7) are scaling constraints that can be satisfied by recalling b and w. In addition, since maximizing is equal to minimize $|w|^2$ and if both boundary plane is represented in the form of an unequaled equation, as shown in equation (9) then the best hyperplane search with the largest margin value can be formulated into a constraint optimization problem as in (10).

$$y_i(x_i.w + b) - 1 \geq 0 \quad (9)$$
$$min \frac{1}{2} |w|^2$$
$$s.t \; y_i(x_i.w + b) - 1 \geq 0 \quad (10)$$

A solution of equation (10) will be easier to do if converted to the lagrangian formula by using Lagrange multiplier. Thus the constraint optimization problem can be converted to equation (11). With the addition of a constraint, (value of the LaGrange coefficient) by minimizing Lp to w and b, then from equation (12) is obtained the equation (13).

$$\min_{w,b} L_p(w, b, \alpha) = \frac{1}{2}|w|^2 - \sum_{i=1}^{n} \alpha_i \, y_i (x_i.w + b) + \sum_{i=1}^{n} \alpha_i \quad (11)$$

$$\frac{\partial}{\partial w} L_p (w, b, \alpha) = 0 \quad (12)$$

$$\sum_{i=1}^{n} \alpha_i \, y_i = 0 \quad (13)$$

The vector w is often of great value, but the value α_i is finite. For that reason, the primitive lagrangian problem L_p is solved into a lagrangian dual problem L_D. Equation (14) is a dual problem lagrangian equation. Substituting to the equation L_p we obtained L_D by a different constraint.

$$L_D(\alpha) = \sum_{i=1}^{n} \alpha_i - \frac{1}{2}|w|^2 - \sum_{i=1,j=1}^{n} \alpha_i \alpha_j \, y_i \, y_j x_i x_j \quad (14)$$

$$s.t \; \sum_{i=1}^{n} \alpha_i \, y_i = 0, \alpha_i \geq 0$$

Thus a value α_i can be obtained that can be used to find w and values α_i for each training data. The training data which have value $\alpha_i > 0$ is support vector, while the rest have value $\alpha_i = 0$. Thus the decision function is only influenced by the support vector.

The hyperplane search formula is a quadratic programming problem so that the global maximum value of α_i(8)always can be found. After the quadratic programming problem solution is found (value α_i), then the class of the test data x can be determined based on the value of the decision function according to equation (15).

$$f(x_d) = \sum_{i=1}^{n} \alpha_i \, y_i \, x_i.x_d + b \quad (15)$$

Where x_i is support vector, n is the number of support vector and x_d is the data which will be classified.

At training stage, face image will through three important stages namely preprocessing, extraction and the training itself. Image preprocessing result of this has the same standard. This image is its then be extracted. The extraction process is done by using HOG. Features HOG vectors resulting from the extraction then being trained using SVM. From the training, results will be obtained SVM classification models of gender. In the testing phase, the test images will be processed as in the training stage. The difference is at the stage of testing the features of the test images will be compared with SVM models. After that, the outcome will be decided, whether the test images classified into male or female. The proposed stage of training and testing are shown in Figure 1.

3. RESULT AND ANALYSIS

In this research, testing on a wide variety of test data. This variation includes variations of image masking, unmasking, and parameter in SVM and parameters used in HOG.

3.1. Preprocessing Result

Face detection process is performed in this study was performed using two features. The first and second features haar and LBP cascade features. From the testing of the image in the database 11, the success rate is performed using higher than the LBP feature uses Haar features. Therefore, the LBP feature is used to detect faces in the preprocessing stage. After that, we do cropping step for face region which detected. The result is shown in Figure 3.
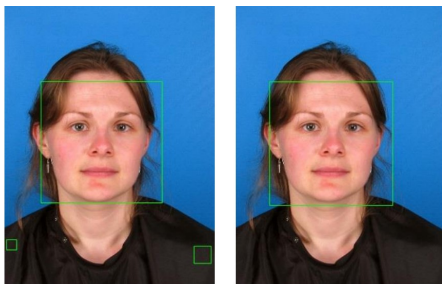

Figure 3. Haar vs LBP Cascade Face Detection

After cropping, we mask the region of the face with a mask image. The process of masking is the process to get the main part of the face. Masking method used is the masking ellipse. At first, there are two images that result from the facial image cropping and image-shaped ellipse. The image of this ellipse can set width and height as well as coordinates. Both images are then combined. As a result, there will be a face image within the ellipse.
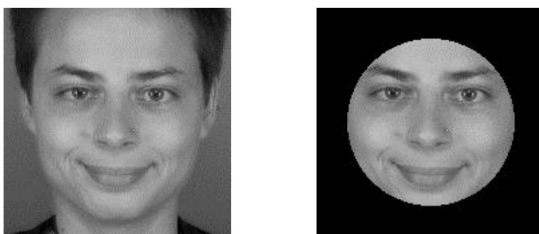

Figure 4. Cropping Image and Masking Image

The area inside the ellipse will not be colored and the area outside the ellipse image will then be filled with black color. Thus, the image of a face that looks only a facial image main parts. The result of masking step shown in figure 4.

3.2. SVM Classification Result

3.2.1. Tests using image masking and non-masking

Testing unmasking image processing is done by unmasking images into the system. The results of this test can be used as a reference standard of image masking effect. This is because of the unmasking image, obtained from the feature extraction process more than masking the image so that the image-making features in more detail. We used training image dataset for the testing process and a set testing image dataset which never trained before. Figure 5 (a) shown tests with themselves (training dataset) with non-masking face image and Figure 5 (b) with masking face image. The result shown, we obtained accuracy more than 97 % for this scenario test stage.
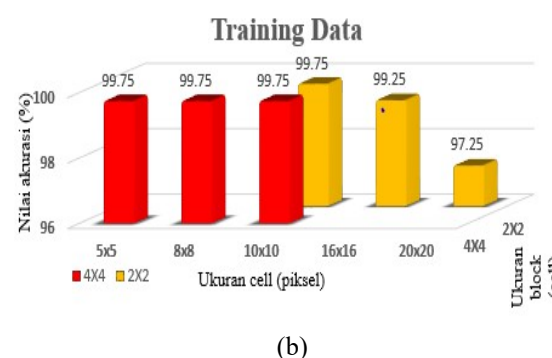

(a)

(b)

Figure 5. Tests using training dataset for (a) non-masking face image (b) masking face image

We also test for testing data set, the result as shown in Figure 6 (a) of the 6 test variations using unmasking image accuracy values obtained ranged

between 88% -91.5%.

Table 1: The variation of block and cell size

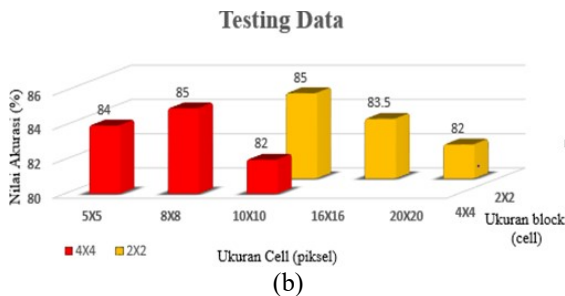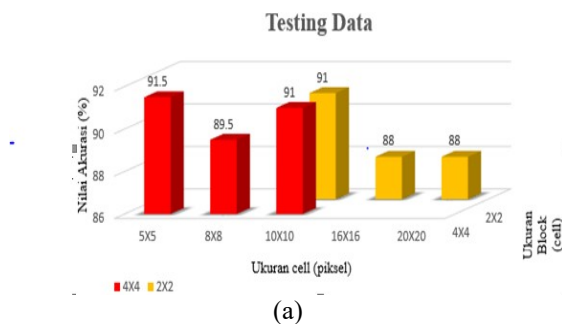| No	Cell Size (pixel)	Block Size	
		(pixel)	(*cell*)
1.	10x10	20x20	2x2
2.	5x5	20x20	4x4
3.	16x16	32x32	2x2
4.	8x8	32x32	4x4
5.	20x20	40x40	2x2
6.	10x10	40x40	4x4

(a)

(b)

Figure 6. Tests using testing dataset for (a) non-masking face image (b) masking face image

This is different from what obtained in the image masking as found in Figure 6 (b). Value of accuracy in all test variations using masking image is lower than the value of accuracy on unmasking image. Therefore, further tests will only be carried out by using the image of the nation masking.

3.2.2. Tests using varies of cells size and blocks size on feature extraction step

Tests on block and cell sizes used in the extraction process cannot be separated from one another. This is because the size of the block and the cell affect each other in the process of taking the feature. The cell is a collection of pixels that unite

in a certain size. Meanwhile, the block is a collection of cells that merge into a single unit. These two parameters have an effect on the number of features generated from the extraction process. Variation of block size used in this research there is 2 kinds that are 2x2 and 4x4 (each in the unit cell). While the cell size used in this research that is 5x5, 8x8, 10x10, 16x16, 20x20 (each in pixels).

Based on Figure 6 (a) it can be analyzed that an accuracy value of 91.5% is obtained in the variation of cell size and a block of 10x10 and 20x20 pixels and bin number 9. From the variation of cell size, this block and bin obtained features of 121.104. If sorted by the number of features generated the following sequence obtained 1,3,5,2,4,6 (see Table 1). But in testing it turns out, there is an anomaly relationship between the variation of cell size and block to the value of accuracy. In case number 3 although the resulting feature is more than the number. But, the accuracy value obtained at number 3 is smaller than the number 5. Accuracy value obtained on number 3 by 89.5% and number 5 by 91%.

3.2.3. Tests Using Varies Of Bin Size

Testing the variation of bin count in masking image will be done by using a cell size of 5x5 (pixel) and block stride size 4x4 (cell). The size of the cell and block is chosen because the accuracy value of this variation yields the highest result so it is best stated.

Figure 7. Values of Accuracy using variation of bin count

The accuracy value obtained from the test using a variation of bin count can be seen in Figure 7. The greatest accuracy value obtained in bin 5 and 7 is 92%. While the smallest accuracy value is obtained in bin 8. From the result of testing the influence of HOG parameter (cell size, block, and bin) got parameter value which can yield highest accuracy level. These values will be used to determine the effect of other parameters in SVM.

3.2.4. Testing variation of iteration on linear kernel SVM

Iteration Shows The Number Of Loops That Occur In The Training Process. This Loop Is Used To Get The Best Solution From Constrained Quadratic Optimization Problem In SVM Calculation. The Variation Of The Iterative Values Used In This Study Ranges From 1-100.000. Table 2 Shows The Accuracy Value In Testing Variation Of Iteration Value.

According to Table 2, it can be observed that an iteration value of 1 obtained an accuracy of 48%. This accuracy value is increased until the iteration variation of 1000. An iteration value above 1000 will result in a stagnant accuracy of 80%.

Table 2. Testing variations of iterative values

No	Iteration	Non-Masking Image			Acc (%)
		True	False	Number of images	
1.	1	96	104	200	48
2.	10	131	69	200	65,5
3.	100	149	51	200	74,5
4.	1000	160	40	200	80
5.	10000	160	40	200	80
6.	100000	160	40	200	80

3.2.5. Testing epsilon variation on linear kernel SVM

In this study used variations of epsilon of 1e-6 to 1e4. From all this variation obtained an accuracy value of 80%. This shows that the variation of epsilon value on linear kernel has no influence.

Table 3. Testing variation of epsilon

No	Iteration	Non-Masking Image			Acc (%)
		True	False	Number of image	
1.	1e-6	160	40	200	80
2.	1e-5	160	40	200	80
3.	1e-4	160	40	200	80
4.	1	160	40	200	80
5.	10000	160	40	200	80
6.	100000	160	40	200	80

Table 3 shows the accuracy value resulting from variations epsilon using variations the value of C in linear kernel SVM is done by varying the C value from 0.1 to 100,000. From all variations of this C, a value obtained an accuracy value of 80%. In this study, changes in the value of C has no effect on increase or decrease in accuracy value.

3.2.6. Testing using RBF kernel

Figure 8 shows the accuracy values obtained from testing using the RBF kernel. In testing using this kernel, the value of gamma is 1 (default), C = 1 (default), iteration = 10,000 and epsilon is 1e-6.
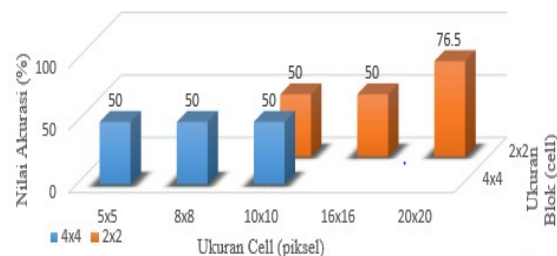

Figure 8. Non-masking image testing using the RBF kernel

Based on Figure 8, the highest accuracy value obtained in the test using the RBF kernel is 76.5%. In some cases, the RBF kernel is a robust kernel in classifying an object. In this study, however, the kernel is worse than a linear kernel. The comparison between Figure 6 (a) and Figure 8, shows that the accuracy value obtained in the test using Linear kernel is better than the RBF kernel. For example, in 5x5 pixels cell size and 4x4 cell block size, Linear kernel obtained 91.5% accuracy value while in RBF kernel obtained 50% accuracy value.

This significant difference in value accuracy occurs because the RBF kernel is not appropriate for classifying objects that have a very large number of features. In the variation of HOG parameters as above will be obtained a feature of 121.104. this feature is a feature in 1 image. If there are 400 images in the training then there are 48,441,600 features overall.

3.2.7. Testing uses variation of gamma values in RBF

One parameter that can be varied in the RBF kernel is gamma. Gamma is a parameter that serves

to transform the training set into the feature space. In this test, variations in gamma values from 1e-6 to 10,000 are performed. From the total variation of this value obtained the same accuracy value of 50% (except on the gamma value of 1e-2 ie 71.5%). Figure 9 shows the accuracy values obtained from testing using variations in gamma values.
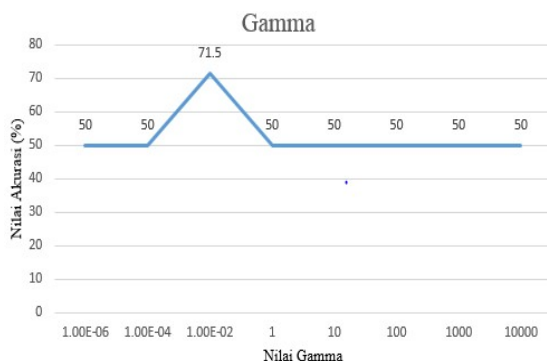

Figure 9 Test uses the variation of gamma values in the RBF kernel SVM

4. CONCLUSION

Based on the research that has been done can be obtained some conclusions that have been successfully implemented SVM method for the classification of human gender on PC Acer Aspire-One Intel Pentium Dual-Core 2.16 GHz OS Linux 14.04 LTS. The use of the main face image (masking image) will decrease the accuracy of the human sex classification system.

In the linear kernel SVM, the higher the iteration value the more accurate the value increases. The maximum accuracy value is obtained at 1000 iterations. Meanwhile, the C and epsilon parameters have no effect on increasing the accuracy of the system. Tests using linear kernels yield better accuracy values than using the RBF kernel. The best test is obtained in non-masking image by using linear kernel which has variation of parameter value C = 1, epsilon = 1e-6, iteration = 10.000 and variation of HOG parameter cell size = 20x20 pixels, block = 5x5 pixels, block stride = 5x5 pixels and bin = 5/7 with an accuracy of 92%. In addition, it can be concluded that the classification of objects with a very large number of features is more appropriate to use linear kernel than the RBF kernel.

In this study the process of image capture the main face or masking image is done using static masking, so often the image of the face masking

results are not symmetrical (there is an empty part). Therefore, further research should be used ellipse dynamic masking (can follow the face position). In addition, it is also expected that the extraction process is done based on parts of the face such as nose, mouth, eyes etc.

REFERENCES :

[1] Lin, G.S. dan Zhao, Y.J., 2011, A Feature-Based Gender Recognition Method Based on Color Information, *First International Conference on Robot, Vision and Signal Processing*, pp.40–43.

[2] Khan, S.A., Nazir, M., Akram, S., Riaz, N., 2011, Gender Classification Using Image Processing Techniques: A survey, *IEEE 14th International Multitopic Conference (INMIC)*, pp.25-30.

[3] Özbudak, Ö.; K¿rc¿, M., Çak¿r, Y. Güne¿, E. O., 2010, Effects o the Facial and Racial Features of Gender Classification, *15th IEEE Mediterranean Electrotechnical Conference Melecon*, pp 26-29.

[4] Ravnik, R., dan Solina F., 2012, Interactive and Audience Adaptive Digital Signage using Real-Time Computer Vision, International Journal of Advanced Robotic System, Vol. 10.

[5] Timotius I.K., dan Setyawan I., 2014, Using Edge Orientation Histograms in race-based Gender Classification, *International Conference on Information Technology Systems and Innovation (ICITSI)*, pp.93-98.

[6] Guo, J., Lin, C. dan Nguyen, H., 2010, Face Gender Recognition Using Improved Appearance-Based Average Face Difference and Support Vector Machine, *International Conference on System Science and Engineering*, pp.637–640.

[7] Bissoon, T., dan Viriri, S., 2013, Gender Classification using Face Recognition. *International Conference on Adaptive Science and Technology (ICAST), 2013*, pp.1–4.

[8] Mirza, A.M., Husain, M., Almuzani, H., Muhammad, G., Aboalsamh, H., Bebis, G., 2013, Gender Recognition Using Fusion of Local and Global Facial Features, *9th International Symposium, International Symposium on Visual Computing*, pp.493–502.

[9] Moghaddam, B., and Yang, M., 2000, Gender Classification with Support Vector Machines. *Proceedings, Fourth IEEE International Conference on Automatic Face and Gesture Recognition, 2000.*, 2, pp.306 – 311.

[10] Dalal, N., dan Triggs B., Histogram of Oriented Gradients for Human Detection, *https://lear.inrialpes.fr*

[11] Raluca, B., 2013, Histogram of Gradient, *http://users.utcluj.ro/~raluca/prs/.*

[12] Sembiring, K., 2007, Tutorial SVM Bahasa Indonesia, Teknik Informatika ITB, Bandung, *http://sutikno.blog.undip.ac.id/files/2011/11/tutorial-SVM-bahasa-indonesia-oleh-krisantus.pdf* .