An Approach in Auto Valuing for Optimal Threshold of Viola Jones

Indrabayu¹, Nurzaenab², Ingrid Nurtanio³
Universitas Hasanuddin, Makasar
¹E-mail: indrabayu@unhas.ac.id, ²nurzaenaab@ieee.org, ³Ingrid@unhas.ac.id

Abstract. The biometric features for people identification on the face include eyes, nose, and mouth. In this study, the three features are detected by using Viola Jones method. The affecting parameter on the accuracy of face feature detection by using Viola Jones is the threshold value because it is the input criterion for Mergethreshold parameter on Matlab. The appropriate threshold value will decrease the detection error so that yielding appropriate ROI. Therefore, this study employs a process of finding out the automatic threshold value by using Viola Jones feature detection. The input data of the system is in the form of frame video from 4 respondents which yield totally 938 face frames as the result of automatic detection that is extracted with matrix 130x110. The result of the testing yields accuracy as much as 79.17% for eyes feature, 94.81% for nose feature, and 99.49% for mouth feature. The whole features yield approximate accuracy as much as 91.15% while the testing result of the time needed by 4 respondents with total 20 video frames is 0.368 seconds for eyes feature detection, 1.065 seconds for nose feature, and 4.665 seconds for mouth feature.

Keyword. Frame Video; Matlab; Region of Interest; Threshold Value Automatic; Viola Jones

1. Introduction

Face detection is the main and the first step that is done in several biometric applications, such as face recognition system, security access system and video supervision [1]. The initial step in doing face recognition is preprocessing step which is detecting the face in order to separating it and its background. Several methods which are used to do face recognition include Viola Jones, Convolutional Neural Network (CNN), and Support Vector Machines (SVM) combination [2][3].

Rina Anriani Tacok did a study to classify eyes and mouth area into drowsy and awake categories. To do the classification, eyes and mouth preprocessing steps are detected by using Viola Jones method. To get the appropriate result, this study searches for optimal threshold value that is suitable with the input image. On the eyes detection step, several experiments were done with tested threshold values were 20, 40, 60, 80, and 100 on the 300 input image. The result of the threshold value yields high accuracy in mouth detection as much as 60 with approximate accuracy 70.28%. Therefore, On the mouth detection step, several experiments were done with tested threshold values were 120, 140, 160, 180, and 200 on the 300 input image. The result of the threshold value yields high accuracy in mouth detection as much as 200 with approximate accuracy 83.05% [4][5]. The following study was done by Naveen Kumar Gondhi et al. that detected face spots by combining Haar-like and Corner detection. For the process of feature detection, this study employs additional subarea by using antropometric limitation to limit the area searching for decreasing fake detection and increasing accuracy significantly. In this feature detection process, this study employs optimal threshold value as much as 200 [6].
Nowadays, the challenges for the researchers in doing face recognition include the appearance of picture variation or video in terms of scale, location, orientation, pose, expression, and lighting condition on the data of input [7][8]. Upon using Viola Jones as the feature detection method, a good image of photo and video frame have different pixel intensities that they need different threshold values as well [5][9]. Therefore, one of the problems of Viola Jones method is the accuracy in determining the threshold value as the parameter value of Mergethreshold of Matlab that it affects on the more accurate detection process due to the less detection error.

Therefore, this study is conducted to find out the optimal threshold value automatically based on the input image in the form of video. The accurate threshold value will decrease detection error on face detection of eyes, nose, and mouth.

2. Proposed Method

The system design to find out the accurate threshold value is generally shown in Figure 5 and it is used to find out the automatic threshold value of eyes, nose, and mouth.

2.1 Data Description

The data of input in this study is video from 4 respondents that have been done face detection by using Viola Jones, extracted the frame, and changed into matrix 130x110. For the result of eyes, nose, and mouth feature detections, each has been changed into matrix 40x95, 40x60, and 40x60. For each of the frame, it has 96x96 dpi. Several data of input can be seen in Figure 1.

2.2 Retrieval Threshold Value Automatic Process

The followings are the steps to find out the optimal threshold value. Where the initial stage is the phase detection feature of the eyes, nose, and mouth by using Viola Jones method and following described the proposed algorithm to set the automatic threshold value on the system.

2.2.1 Feature Detection. Face detection is the main step in this study. The method employs Viola Jones method that is used to detect eyes, nose, and mouth features on the data of input. Viola Jones method was introduced by Viola and Jones with detection accuracy 93% and speed that is fifteen times faster than Rowley Baluja-Kanade detector and is approximately six hundred faster than Scheiderman-Kanade detector [10]. Viola Jones combines four main keys to detect the features which are Haar-like features, Integral Image, AdaBoost Learning, and Cascade Classifier. Haar-like Features. This algorithm uses four kinds of features as shown in Figure 2.

Figure 2 shows that Haar-like Features algorithm has two features consisting of two rectangulars (A and B), one feature with three rectangulars (C), and one feature with four rectangulars (D). To find out Haar-like features, it can be used the following equation (1).

\[ F(H) = \sum F_{WH} - \sum F_R \]  

Where \( F(H) \) is the total features value got without passing threshold (limit) as determined. \( \sum F_{WH} \) is the feature value of the light area and \( \sum F_R \) is the feature value of the dark area.

Integral Image, it is a technique to count the feature value quickly by changing the value of every pixel to be a new image representation. The example of integral image is shown in Figure 3. Point \((x,y)\) on Figure 3. can be counted by using the following equation (2).

\[ i(x,y) - \sum x' \leq x, y' \leq y \rightarrow i(x',y') \]  

Where \( i(x,y) \) is the image integral on \( x,y \) locations and \( i(x',y') \) is the pixel value on the original image.

AdaBoost Learning, it is a technique used to increase the work of classification by using simple learning which is combining many weak classifiers to form strong classifier. The weak classifier can be counted by using the following equation (3).

\[ h_j(x) = \begin{cases} 1, & \text{if } p_j f_j(x) < p_j \theta_j(x) \\ 0, & \text{otherwise} \end{cases} \]
Where $h_j(x)$ is the weak classifier, $p_j$ is the party to $j$, $\theta_j$ is the threshold to $j$, and $x$ is the dimension of sub image.

The strong classifier can be counted by using the following equation (4).

\[
h(x) = \begin{cases} 
1, & \sum_j 1 \ a_j h_j(x) \geq \frac{1}{2} \sum_j 1 \ a_j \\
0, & \text{otherwise} 
\end{cases}
\]  

(4)

To combine the complex classifier in a structure with levels, it is used Cascade classifier.

*Cascade Classifier*, deals with increasing the speed of object of detection by focusing on the potential image area. The form of Cascade Classifier flow can be seen in Figure 4. Figure 4 shows that on the first level of classification, every subimage will be classified by using one feature. The classification result yields sign T (True) if the pictures fill the certain Haar features and F (False) if it does not. The classification will approximately leave 50% subimage to be classified on the second step. The second classification result yields sign T if the picture fills the integral image process and F if it does not. As the increasing of classification level, it is needed more specific requirement so that the feature will yield much. The total of subimage onto the classification will decrease down to 2%. The last classification result yields sign T if the pictures fill Adaboost and F if it does not [10].

![Figure 1. Video Frame Image Data](image1)

![Figure 2. Haar-like Features Type [10](image2)

![Figure 3. Integral Image [10]](image3)
2.2.2 To Find Out Automatic Optimal Threshold. **Threshold value**, the value of even number determined to define the criteria needed as merged threshold value parameter on Matlab in determining detection on the face area. The group of detection result is allocated based on the threshold value to yield one box of limitation around the object of target. Increasing the threshold value can help decrease the fake detection around the object [11].

The following is the system design of how to find out automatic optimal threshold value as shown in Figure 5:
Figure 5. shows the steps in finding out the automatic optimal threshold value. $T_0$ value is the initializing the initial value of threshold value and is the minimum value of mergethreshold parameter on Matlab. $T_0$ value is used to detect the feature, the detection result will be shown in the form of box around the face (as in Figure 6). The detection result box of face shows the formed ROI based on the threshold value inputted. The more ROI formed, the less accurate the threshold value inputted as mergethreshold parameter. Feature detection is said to be successful if the ROI that is formed is only one. Therefore, there is a condition matching process on the ROI to see whether the number of ROI is one or more. If the condition of the number of ROI is more than one, then the system will add the threshold value automatically and it becomes the new input for the mergethreshold parameter on Matlab. Yet, if the condition of the number of ROI is one, then the system will do automatic cutting on the detected feature. In this system, it is given a condition in which if the number of ROI is zero or condition in which the feature of the input image can no longer be detected though using the minimum threshold value which is 0, the system will clarify it as the broken or ignored feature. In the following will be explained the steps of finding out the automatic threshold value as explained in Algorithm 1.

**Step 1:** If the data of input is in the form of video frame with matrix dimension 130x110.

**Step 2:** Initializing the $T_0=0$ value (the value of initial threshold equals 0).

**Step 3:** Doing feature detection process by using Viola Jones method and input value $T_0=0$. The features detected can be in the form of eyes, nose, and mouth features.

**Step 4:** Taking the number of ROI formed from feature detection with value $T_0=0$.

**Step 5:** If $\sum$ROI $\geq$ 1 (the number of ROI is more than one), then the system will renew the $T_0$ to be $T_t=T_0+1$ (adding threshold value automatically) and be processed again on step 3 and stop until $\sum$ROI $=1$ and then doing feature cutting of detection result.

**Step 6:** If it is found a condition in which $\sum$ROI $=0$ or the ROI is empty, then the system will ignore the feature and continue feature detection to the next image.

The step explained in Algorithm 1 focuses on three conditions that will be running on the input image. First condition, if the feature detection result on the image has ROI number more than one, then the feature detection process needs additional automatic threshold value up to the number of ROI equals one. Second condition, if the feature detection result on the image has number of ROI equals one, then the feature detection process has been optimal because the feature detection result ideally happens if the number of ROI equals one. Third condition, if the number of ROI is lesser than one or zero, then the feature detection process needs threshold value decreasing. However, the system initializes the initial threshold value which is the initial limit of threshold value. Therefore, the feature with number of ROI equals zero, it will be considered as not good and the system will continue the process to detect the next feature.

2.3 System Performance

System performance is needed to count the accuracy of the work of the system in detecting eyes, nose, and mouth features with automatic threshold value. The performance can be counted by using the following equation (5).

$$P_s = \left(\frac{P_T}{D_T}\right) \times 100$$

(5)

Where $P_T$ is the feature detection result as detected by the system. $D_T$ is the number of video frame. The performance of the testing result will be explained in two performances which are detected feature and ignored feature as determined by the system.

3. Result and Discussion

The initial step is inputting the image. The images used is divided into video frame with matrix 130x110 as shown in Table 1. The example of input image can be seen again in Figure 1.
Table 1. Table Test Data

| No. | Respondents | Number of Face Detection Result on Video (Frame) | Size of Matrix (Pixel) | Resolution of Video Frame (dpi) |
|-----|-------------|-------------------------------------------------|------------------------|-------------------------------|
| 1   | R1          | 300                                             | 130x110                | 96x96                         |
| 2   | R2          | 155                                             | 130x110                | 96x96                         |
| 3   | R3          | 43                                              | 130x110                | 96x96                         |
| 4   | R4          | 440                                             | 130x110                | 96x96                         |
|     | Total       | 938                                             | 130x110                | 96x96                         |

The next step is doing feature detection process of face by using Viola Jones method, and the feature detection result of face can be seen in Figure 6.

Figure 6 (a). shows the threshold value that equals 0 that is inputted to detect eyes, nose, and mouth are not suitable with input image, so it yields the number of ROI more than one and the system cannot take the eyes, nose, and mouth features appropriately. While Figure 6 (b). shows that every input image has different threshold values based on the feature that will be detected. Yet, finding out the threshold value manually based on the input image will need several trials until the threshold value yields one ROI successfully or detects the feature appropriately. Therefore, system to find out automatic threshold value will be needed because it decreases the time needed to find out the threshold value manually.

Figure 7. shows the description of the process to find out the optimal threshold value automatically to detect the eyes, nose, and mouth features on the system. The initializing of the threshold value is 0 and it can be seen on Figure 7 (a1), (b1), (c1) that the number of the limitation box or ROI that is detected is much affecting on the disability to detect the feature appropriately or yielding one ROI. Therefore, the system will automatically add the threshold value to be 1 and the new threshold value that has been renewed to be 1 becomes the new input of the Mergethreshold on Matlab, the result is shown on Figure 7 (a2), (b2), (c2) that the number of the limitation box or ROI decreases lower than before. The condition will stop if the number of the limitation box or ROI value is one as shown on Figure 7 (a3), (b110, and (c60) and then the system will automatically do the detected feature cutting.

**Figure 6** (a). The result of eyes, nose, and mouth detection with T=0 (b). the result of eyes detection with T=1, nose detection with T=10, and mouth detection with T=60.
Figure 7 (a1) and (a2) is the result of eyes feature detection process by using the proposed system, (b1) and (b2), … (b11) is the result of nose feature detection process by using the proposed system and (c1), (c2), ... (c60) is the result of the mouth feature detection process by using the proposed system.

The result of the system testing on the video frame is explained on Table 2, 3, and 4 by using equation (5) to count accuracy. The testing is needed to see the work of the system to find out the automatic threshold value in detecting the eyes, nose, and mouth features appropriately.

**Table 2. The Result of System Testing on The Eyes Feature**

| Respondents | Total Frame | Eyes Detected | Accuracy (%) | Ignored System | Accuracy (%) |
|-------------|-------------|---------------|--------------|----------------|--------------|
| 1           | 300         | 226           | 76.10        | 74             | 23.90        |
| 2           | 155         | 153           | 98.52        | 2              | 1.48         |
| 3           | 43          | 23            | 53.49        | 20             | 46.51        |
| 4           | 440         | 389           | 88.57        | 51             | 11.43        |

The range of feature accuracy as detected by the system 79.17 %

the range of feature accuracy as ignored by the system 20.83 %
Table 3. The Result of System Testing on The Nose Feature

| Respondents | Total Frame | Nose Detected | Accuracy (%) | Ignored System | Accuracy (%) |
|-------------|-------------|---------------|--------------|----------------|--------------|
| 1           | 300         | 286           | 95.93        | 14             | 4.07         |
| 2           | 155         | 154           | 99.26        | 1              | 0.74         |
| 3           | 43          | 43            | 100.00       | 0              | 0.00         |
| 4           | 440         | 370           | 84.05        | 70             | 15.95        |

The range of feature accuracy as detected by the system 94.81 %
the range of feature accuracy as ignored by the system 5.19 %

Table 4. The Result of System Testing on The Mouth Feature

| Respondents | Total Frame | Mouth Detected | Accuracy (%) | Ignored System | Accuracy (%) |
|-------------|-------------|---------------|--------------|----------------|--------------|
| 1           | 300         | 298           | 99.29        | 2              | 0.71         |
| 2           | 155         | 155           | 100.00       | 0              | 0.00         |
| 3           | 43          | 43            | 100.00       | 0              | 0.00         |
| 4           | 440         | 434           | 98.66        | 6              | 1.34         |

The range of feature accuracy as detected by the system 99.49 %
the range of feature accuracy as ignored by the system 0.51 %

The result of the system testing to count the time needed by the system to find out the automatic detection of threshold value on video frame is explained on table V. In the testing, the number of the tested frame is as many as 5 frames with different face poses of every respondents in which every frame yields 3 features which are eyes, nose, and mouth. Therefore, there are 60 testing results for 4 respondents.

Table 5. The Result of System Testing Time on Feature Detection

| No. | Respondents | Number of Frame | Eyes (Seconds) | Nose (Seconds) | Mouth (Seconds) |
|-----|-------------|-----------------|---------------|---------------|-----------------|
| 1   | 1           | 5               | 0.585         | 0.644         | 5.539           |
| 2   | 0.301       | 0.648           | 6.591         |
| 3   | 0.361       | 0.621           | 6.119         |
| 4   | 0.326       | 0.839           | 8.066         |
| 5   | 0.414       | 0.749           | 3.037         |
| 6   | 0.711       | 2.407           | 5.297         |
| 7   | 0.419       | 2.065           | 2.967         |
| 8   | 0.335       | 1.773           | 6.689         |
| 9   | 0.299       | 1.252           | 12.753        |
| 10  | 0.497       | 3.373           | 8.797         |
| 11  | 0.034       | 0.634           | 1.033         |
| 12  | 0.350       | 0.816           | 5.792         |
| 13  | 0.312       | 1.231           | 3.381         |
| 14  | 0.320       | 0.608           | 8.323         |
| 15  | 0.318       | 0.599           | 6.343         |
| 16  | 0.731       | 0.615           | 0.551         |
4. Conclusions

The objective of the study is to apply face detection and face feature by finding out automatic optimal threshold value that is suitable with the input image. The result of the feature detection for 4 respondents with total 938 video frames on eyes feature yields approximate accuracy 79.17%, nose feature yields approximate accuracy 94.81%, and mouth feature yields approximate accuracy 99.49%. The range of accuracy for the whole eyes, nose, and mouth features is 91.15% while the result of the time testing needed by the system for 4 respondents with total 20 video frames for each features is eyes feature needs approximate detection time 0.368 seconds, nose feature needs approximate detection time 1.065 seconds, and mouth feature needs approximate detection time 4.665 seconds. The result shows that the system can be used for the steps of eyes, nose, and mouth feature detections on the face that is going to be the input parameter to do extraction, classification, recognition processes on the future face recognition research.

References

[1] G.-C. Luh, “Face Detection Using Combination of Skin Color Pixel,” Int. Conf. Mach. Learn. Cybernatics, pp. 13–16, 2014.
[2] M. Nehru and P. S, “Illumination invariant face detection using Viola Jones Algorithm.” pp. 4–7, 2017.
[3] M. D. Putro, T. B. Adji, and B. Winduratna, “Adult image classifiers based on face detection using Viola-Jones method,” Proceeding 2015 1st Int. Conf. Wirel. Telemat. ICWT 2015, pp. 0–5, 2016.
[4] R. A. TACOK, “RECOGNITION OF HUMAN FACE FEATURES FOR DROWSY DRIVER DETECTION,” Hasanuddin University, 2017.
[5] Indrabayu, R. A. Tacok, and I. S. Areni, “Modification on brightness enhancement for simple thresholding in eyelid area measurement,” ACM Int. Conf. Proceeding Ser., vol. Part F1309, pp. 101–104, 2017.
[6] N. K. Gondhi, N. Kour, S. Effendi, and K. Kaushik, “An efficient algorithm for facial landmark detection using haar-like features coupled with corner detection following anthropometric constraints,” 2017 2nd Int. Conf. Telecommun. Networks, pp. 1–6, 2017.
[7] A. Alqudah and O. Debeir, “Face Detection using Viola and Jones Method and Neural Networks,” no. 1, pp. 40–43, 2015.
[8] J. Yoon, J. Shin, and D. Kim, “Multi-pose face detection for silver robots,” 2013 10th Int. Conf. Ubiquitous Robot. Ambient Intell. URAI 2013, pp. 755–759, 2013.
[9] A. Fernández, R. García, R. Usamentiaga, and R. Casado, “Glasses detection on real images based on robust alignment,” Mach. Vis. Appl., vol. 26, no. 4, pp. 519–531, 2015.
[10] P. Viola and M. Jones, “Rapid object detection using a boosted cascade of simple features,” Proc. 2001 IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognition. CVPR 2001, vol. 1, p. 1-511-I-518, 2001.
[11] “Detect objects using the Viola-Jones algorithm - MATLAB.”