



Frontal Face Detection using Haar Wavelet Coefficients and Local Histogram Correlation

Iwan Setyawan¹, Ivanna K. Timotius² & Andreas A. Febrianto

Department of Electronic & Computer Engineering,
Satya Wacana Christian University,
Jl. Diponegoro 52 – 60, Salatiga, Indonesia
Email: ¹iwan.setyawan@ieee.org, ²ivanna_timotius@yahoo.com

Abstract. Face detection is the main building block on which all automatic systems dealing with human faces is built. For example, a face recognition system must rely on face detection to process an input image and determine which areas contain human faces. These areas then become the input for the face recognition system for further processing. This paper presents a face detection system designed to detect frontal faces. The system uses Haar wavelet coefficients and local histogram correlation as differentiating features. Our proposed system is trained using 100 training images. Our experiments show that the proposed system performed well during testing, achieving a detection rate of 91.5%.

Keywords: *biometric; face detection; haar wavelet; image processing; local histogram correlation; pattern recognition.*

1 Introduction

Face detection is a process in which an input image is processed using an algorithm in order to identify which part(s) of the image contains a human face. The output of such a process is generally a cropped facial area (or coordinates thereof) that can be fed to another system for further processing. Face detection is an important component of any “smart” applications dealing with the automatic tracking or recognition of human faces. An example of such a system is the “smart focusing” feature found in more recent digital cameras in which the cameras can automatically tracks areas containing faces and keep those areas in focus. Another example is the feature in some digital cameras where the camera will take a picture as soon as the subject smiles (in other words, the cameras recognize human expression). However, a more widely known example is the face recognition system. This system is now used in various applications. It can be found in applications that enable us to automatically tag a person in a photograph. This enables us to organize our digital photos collection more easily. More importantly, it is also employed in security systems (e.g., to identify criminals). This application is gaining importance since facial recognition is regarded as the least intrusive among other biometric methods.

The relationship between the face detection process and the other systems is presented in Figure 1.

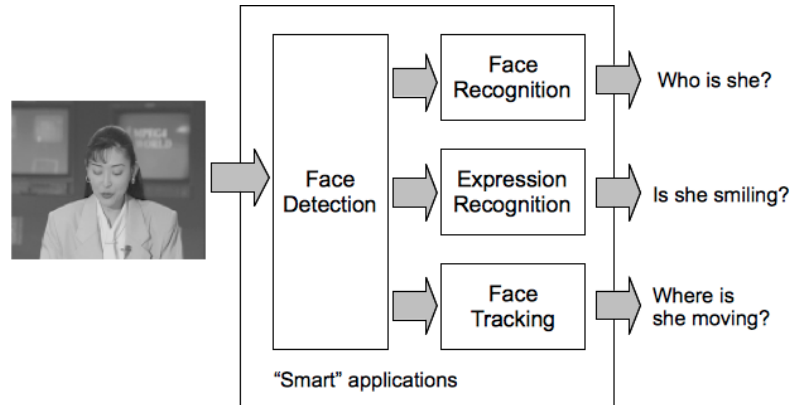


Figure 1 Relationship between face detection and other systems dealing with human faces.

It can be seen in Figure 1 that the aforementioned systems rely heavily on the face detection process. The performance of the face detection process directly influences the overall performance of the system. Ideally, a face detection system should be able to locate and identify all faces in an input image. The system should be able to do this regardless of position, scale, orientation, age and expression [1]. However, it can be said that in general current face detection systems have not been able to attain those goals. Therefore, face detection is currently still an active and challenging field of research.

We can find various approaches proposed in the literature to detect and locate the presence of human faces in an input image. An example of such approach is the use of human skin colour to find facial area candidates [1],[2]. This approach is attractive since it does not have a high computational cost. However, this approach is also known for its sensitivity to human skin colour variation caused by different lighting conditions, presence of shadows and ethnicity of the subject. For this reason, this approach is rarely employed by itself and must be combined with other approaches. Another popular approach that we can find in the literature is the use of template matching [3],[4],[5]. Since the facial areas contain specific objects that do not appear in other parts of the body (e.g., nose, lips, eyes, etc.), another possible approach is detecting facial areas by seeking the presence of such objects [2]. Finally, some authors propose the use of statistical-based methods [6],[7]. First the statistical models of both face and non-face regions are constructed. Then these models are used to identify areas of the input picture that contain faces.

The system proposed in this paper uses two differentiating features to identify areas that contain human faces. The first feature is the 1-dimensional Haar wavelet coefficients. To build this differentiating feature, we construct reference feature vectors by performing 1-dimensional Haar wavelet transformation on the training samples. These reference vectors are then compared to the feature vectors extracted from an input image. The second differentiating feature is the local histogram correlation. We use a reference facial image to generate the reference facial area histogram. This reference histogram is then compared to the local histogram of the input image. The use of local histogram correlation is to reduce the possibility of a false positive response from the detector.

The rest of this paper is organized as follows. In Section 2 we shall discuss in greater details the differentiating features we use in the proposed system. Section 3 will discuss the proposed face detection algorithm. Section 4 will discuss the experiment setup that we use and present the results of the experiment. Finally, in Section 5 we shall present a discussion of the results and some pointers to our future work.

2 Differentiating Features

Our proposed method uses Haar wavelet transform and local histogram similarity to generate differentiating features in order to identify areas that contain human faces. In this section, we will discuss these differentiating features in more details.

2.1 Haar Wavelet Transform

Haar wavelet transform is the oldest wavelet transform. This transform is also the wavelet transform with the simplest basis function [8]. The Haar wavelet basis function consists simply of two step functions. This basis function can be scaled so that it spans either a large portion of the image (i.e., more spatial extent or lower frequency resolution) or so that it covers only a small portion of the image (i.e., less spatial extent or higher frequency resolution).

At its highest frequency resolution (i.e., its lowest spatial extent), the basis function covers two adjacent pixels of the image being processed. The wavelet representation of the image in this case can thus be calculated by simply using the following equations [7]:

$$I_h(i, j) = I(i + 1, j) - I(i, j) \quad (1)$$

$$I_v(i, j) = I(i, j + 1) - I(i, j) \quad (2)$$

In Eqs. (1) and (2), I_h and I_v refers to the horizontal and vertical 1-dimensional Haar wavelet representations of an input image I , respectively. The indices i and j refers to the spatial position of the image pixels.

Similar to other wavelet transforms, Haar wavelet transform is widely used in image compression methods [8],[9]. However, Haar transformation can also be utilized in pattern recognition applications. In particular, Haar-like transform has been widely used in face detection and recognition applications. This is due to the fact that Haar wavelet coefficients can capture the features of the input image. The authors in [1] describe the use of Haar-like features in face detection problem. The author in [7] also uses Haar wavelet coefficients as one differentiating feature to detect faces.

Despite their simplicity, Eqs. (1) and (2) can be used to capture the horizontal and vertical features of human facial regions. Differentiating features that present in the facial region, such as eyes, nose and mouth can be captured using these equations because they introduce rapid changes in luminance values. In other words, these differentiating features form the high-frequency components of the facial regions. An example of such features are shown in Figure 2.



Figure 2 Horizontal and vertical facial features.

In this figure, the leftmost image represents the original grayscale image of a facial region. The middle and rightmost images represent the horizontal and vertical Haar wavelet representations, respectively. It can be seen from this example that details important for identifying a region as a facial region, such as eyes and nose, can be nicely captured.

2.2 Local Histogram of an Image

The histogram of an image shows the distribution of the pixel values in the image. Therefore, the histogram can convey some information about the image [8]. One such information is the quality of the image. For example, by examining the histogram we can determine whether the contrast of the image is good or poor. We can also use the histogram as a basis for enhancing the quality of an image. A widely used method to enhance the contrast of an image, the

histogram equalization, directly manipulates the image histogram to enhance image quality.

Image histogram can also be used in pattern recognition application. Since the appearance of an object within an image is unique compared to the other areas of the image, it is very likely that the local histogram (i.e., the histogram of the area of the aforementioned object) will also be unique. This information can be used in finding the location of an object within an image. This idea can also be extended as follows. If objects belonging to a certain class are similar to each other, then their histograms would also be similar. Therefore, given that we have a representative histogram for this class of objects, we can use it to locate similar objects in an input image by looking at the local histogram similarity of the areas in the image. An example of this approach in the literature is presented in [10]. Here the authors used histogram similarity to find objects in video frames. Another example is presented in [11]. In this work, the authors use histogram similarity to locate object within a scene captured using a camera. This information is then used to guide a robotic arm.

In this paper, we use local histogram correlation to determine whether the area contains a human face. Our main motivation for the use of histogram correlation is to reduce the probability of false-positive response from the detector. In our previous work [12], we observe that many false-positive response occurs in areas whose frequency content (i.e., Haar wavelet coefficients) are similar to a face although the histogram is completely different.

3 The Proposed Face Detection System

In this section, we shall discuss the proposed face detection system in more detail. We will first give an overview to the related work found in the literature and show the novelty of the proposed system. Next we will discuss the construction of the differentiating feature vectors. Then we will proceed to describe the details of the proposed face detection algorithm.

3.1 Related Works

As described in the Introduction, there are various proposed face detection systems found in the literature. In this section, we shall limit our discussion to those system that are closely related to the proposed system. The authors in [1] and [7] proposed systems that use Haar wavelet coefficients as differentiating features. However, these proposed systems do not take into account the local histogram similarity between the input image and the training images. The authors in [10] and [11], on the other hand, use local histogram similarity to detect objects in a scene. However, neither of the proposed systems are

designed to detect faces. Furthermore, these systems do not use the Haar wavelet coefficients as a differentiating feature.

Based on the previous discussion, we can conclude that to the best of our knowledge there has not been any proposed face detection system that combines the Haar wavelet coefficients and local histogram correlation as differentiating features. Therefore, the novelty and main contribution of this work are twofold. First, we propose a novel face detection system using both the Haar wavelet coefficients and local histogram correlation as differentiating features. Second, we show that the use of local histogram correlation as a differentiating feature can significantly reduce the false alarm rate compared to a system that use only Haar wavelet coefficients.

3.2 Construction of the Differentiating Feature Vectors

The first differentiating feature vector is constructed from Haar wavelet coefficients. To construct this differentiating feature, we need to create reference Haar wavelet coefficients. This reference is constructed from training samples of facial regions. The steps to construct the reference Haar wavelet coefficients are as follow. First, we compute the average face sample from the available training samples. These training samples have to be of the same pose. In this paper we use frontal face samples to construct the average face sample. Then we use Eqs. (1) and (2) to compute the horizontal and vertical Haar wavelet representations, respectively, of the average face sample. We shall call these \mathbf{R}_h and \mathbf{R}_v for the horizontal and vertical reference representations respectively. Next, we reshape \mathbf{R}_h and \mathbf{R}_v into vectors by concatenating its rows. Then we normalize the vectors by using the following equation.

$$\mathbf{R}' = \frac{\mathbf{R} - \mu}{\sigma} \quad (3)$$

In Eq. (3), \mathbf{R}' is the normalized version of \mathbf{R} . The terms μ and σ represent the mean and standard deviation of \mathbf{R} , respectively. We shall call the normalized horizontal and vertical reference vectors \mathbf{R}_h' and \mathbf{R}_v' , respectively. These two vectors, each containing normalized Haar wavelet representations, are then stored as the reference coefficients.

The second differentiating feature vector used in the proposed system is the reference histogram. This reference histogram is constructed from the same average face sample that we use to construct the previous reference vectors. The reference histogram vector is called \mathbf{R}_p .

3.3 Face Detection Algorithm

In the following, we shall discuss the proposed face detection algorithm in more details. The input of the algorithm is an image that may contain facial regions. This input image is processed using a moving window, S_{mn} , of size $M \times N$ pixels. In this paper, we chose $M = N = 48$. In each iteration, we compute the horizontal and vertical 1-dimensional Haar wavelet representations of the window. Let S_h and S_v denote the matrices that contain the horizontal and vertical Haar wavelet representations of the window S_{mn} , respectively. We reshape these matrices into vectors (using the same procedure described in Section 3.1). These vectors are then normalized using Eq. (3). The output of this step is the normalized Haar representation vectors \mathbf{S}_h' and \mathbf{S}_v' . We also compute the local image histogram (i.e., the histogram of the window S_{mn}). This histogram is stored as the vector \mathbf{S}_p . Therefore, we now have three vectors that contains the differentiating features of the currently processed window namely \mathbf{S}_h' , \mathbf{S}_v' and \mathbf{S}_p .

We then proceed to determine whether the currently processed window contains a facial area. We do this by comparing the three feature vectors, \mathbf{S}_h' , \mathbf{S}_v' and \mathbf{S}_p , to the previously constructed reference vectors. We compare the feature vectors by computing the normalized cross-correlation values between \mathbf{R}_h' and \mathbf{S}_h' (and respectively between \mathbf{R}_v' and \mathbf{S}_v' and \mathbf{R}_p and \mathbf{S}_p) as follows:

$$C_h = \frac{\sum_x \mathbf{R}_h'(x) \mathbf{S}_h'(x)}{\sqrt{\sum_x \mathbf{R}_h'^2(x) \sum_x \mathbf{S}_h'^2(x)}} \quad (4)$$

$$C_v = \frac{\sum_x \mathbf{R}_v'(x) \mathbf{S}_v'(x)}{\sqrt{\sum_x \mathbf{R}_v'^2(x) \sum_x \mathbf{S}_v'^2(x)}} \quad (5)$$

$$C_p = \frac{\sum_x \mathbf{R}_p(x) \mathbf{S}_p(x)}{\sqrt{\sum_x \mathbf{R}_p^2(x) \sum_x \mathbf{S}_p^2(x)}} \quad (6)$$

In Eqs. (4), (5) & (6), C_h , C_v and C_p are the cross-correlation values of the horizontal and vertical Haar wavelet representation vectors and local histogram, respectively. The variable x is the index of the elements of the vectors. Our algorithm declare that S_{mn} contains a facial region if all three correlation values are larger than a certain threshold. The thresholds for C_h and C_v are related, while the threshold for C_p is determined separately. Otherwise, the window S_{mn} is declared to be containing no facial region. Let T_h , T_v and T_p denote the

threshold values for C_h , C_v and C_p , respectively. These threshold values are determined empirically. The choice of values of these thresholds is a compromise between the probability of missed detections and false alarms. A missed detection occurs if the system declares that the region being processed is not a facial region while in fact it is. The flip side of a missed detection is a false alarm, in which the system declares that the region being processed is a facial region while in fact it is not. A high threshold value will reduce the probability of false alarm, but increase the probability of missed detections. On the other hand, a low threshold value will reduce the probability of missed detections but increase the probability of false alarm.

In our previous work [12], we use $T_h = T_v$. However, further experiments show that in facial regions the value of C_h is generally larger than C_v , in some cases by a factor of three. Therefore, in this paper we use $T_h = \alpha T_v$. The value of α is also determined empirically. Our experiments seem to indicate that choosing a value of $\alpha = 1.4$ gives satisfactory result for our test images. In this paper, we chose the threshold values T_h , T_v and T_p such that our system is biased towards false alarms. The rationale behind this choice is the assumption that in real-world applications, a false alarm is much more preferable than a missed detection.

After performing the comparison and deciding whether an area contains a face, we then move S_{mn} to the next position and the comparison process is repeated. The window is moved with a vertical and/or horizontal displacement δ , where δ is a positive integer and $\delta \geq 1$. The choice of δ is a compromise between processing time and probability of missed detection. A smaller δ reduces probability of missed detection but increases processing time, and vice versa. In the experiments presented in this paper, we chose $\delta = 3$. While it is possible to assign different values for horizontal and vertical displacements, in this paper we use the same value for both displacements. This process is repeated until all areas of the input image is processed.

It should be noted that a facial region in the input image can appear in various sizes. A close up picture of a person will contain a relatively large area of facial region while a wide-angle shot may contain a relatively small facial region. Therefore, we have to perform the detection process at various spatial scales to deal with this situation. To do this we scale the input image by a scaling factor. This scaling factor, ζ , is also determined empirically in our experiments. Furthermore, in some cases the face might be slightly rotated. This might due to the subject's pose or the angle with which the photograph was taken. In order to deal with this, we rotate the input image by a certain number of degrees. This rotation angle, θ is determined empirically in our experiments.

A summary of the face detection algorithm presented in this section is presented as the flowchart shown in Figure 3.

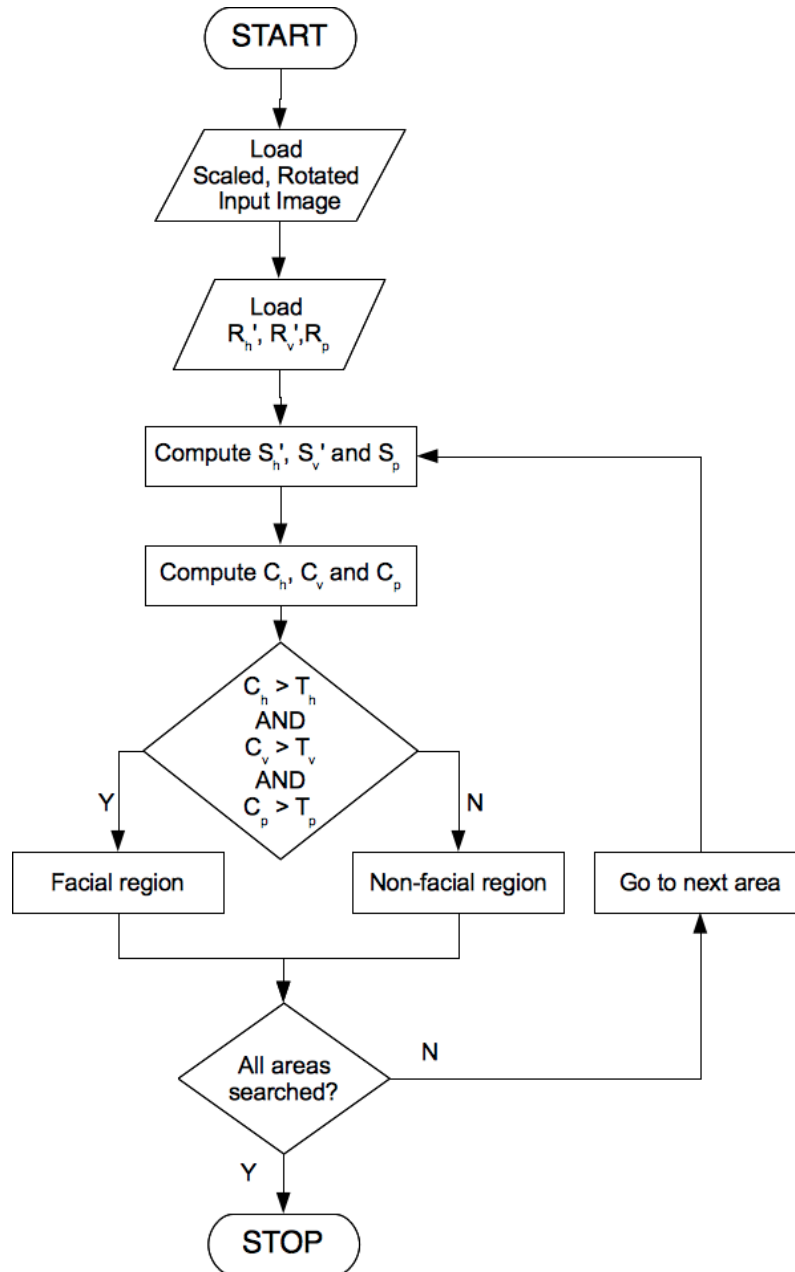


Figure 3 Flowchart of the proposed face detection algorithm.

4 Experiment Setup and Results

In this Section we shall discuss the setup of our experiments and the results we obtained. First, we shall discuss the training face samples we used to build the reference vectors as explained in Section 3.2. Then, we shall proceed to discuss the testing images and procedures.

4.1 Training Face Samples

In our experiments, we construct the reference sample from the VISiO (Video, Image and Signal Processing Laboratory) Multiview Face Database [13]. Examples of the images contained in the database is presented in Figure 4.



Figure 4 Some examples of the images in the VISiO Multiview Face Database.

The VISiO Multiview Face Database currently contains face images of 100 subjects. These subjects are evenly distributed in gender (i.e., 50 subjects are female and 50 subjects are male). The age varies between 19 and 69 years old. The images in the database are taken under controlled condition in our laboratory. Each subject is photographed against a uniform white background using a single camera and identical settings. We used a single fluorescent light source that is placed above, and in front of, the subject. The camera's automatic white balance is used for all subjects. Each image is taken as a colour image. For each subject, we take 105 photographs. Each photograph has a different combination of viewpoint, facial expression, and accessories. In total, the VISiO Multiview Face Database contains 10500 different images. Each image in the VISiO multiview face database is manually cropped around the facial area. Then the images are resampled into a 64×64 pixel 8-bit grayscale image.

For the experiments presented in this paper, we only use a subset of the images from the database. Specifically, we only used samples from the frontal face pose without accessories and neutral facial expression. Furthermore, we performed a pre-processing to the image samples, as follows. First, the original facial region samples in the VISiO Multiview Face Database shows the whole head of the subject, including hair, neck, etc. Our preliminary results show that these areas (i.e., the extra information of the subjects) can actually degrade the performance

of the system. Therefore, we decided to crop the original face samples to the area containing prominent facial features, namely the eyes, nose and mouth. Secondly, in the original image samples there are slight variations in the position and size of the facial region. These variations can be disadvantageous during the creation of the average face sample, yielding an almost “featureless” sample. Therefore, we manually align the faces in the database. To do so we have to slightly rotate or resize some of the faces. The alignment is based on the position of the subjects’ eyes. The result of the pre-processing are face samples that have a resolution of 48×48 pixels. These samples are used to calculate the average face sample used to construct the reference feature vectors. An example of the pre-processed sample and the average face sample is shown in Figure 5.



Figure 5 Example of the pre-processed training sample (left) and the calculated average face sample (right).

4.2 Test Images and Procedure

In order to test the performance of the proposed system, we use input images from various sources. Some images are made specifically for this purpose. The other images are taken from the Combined MIT/CMU Test Images [6] or downloaded from various sites on the Internet. The test images are all grayscale images (or are converted to grayscale from colour images) but have various spatial resolutions. The test images are also of various qualities, in term of sharpness and contrast. Most of the test images used in the test contain only one facial region. However, we also use test images that contain as many as nine faces. Examples of the test images are shown in Figure 6.

It should be noted that the majority of images used in the experiments does not contain the reference facial region samples. Although some subjects in the test set appear both in the reference samples and the test images, the pictures are taken at completely different conditions (different background, pose, lighting conditions, expressions, etc.). It can be said, therefore, that our system has to deal mostly with images that are “new” to it. Finally, since we only use frontal face images as reference, we mainly use test images that contain frontal face images. In total, we use 154 test images containing 200 faces.



Figure 6 Examples of the test images used in our experiments.

Figure 7 shows some examples of the the result of our proposed face detection system. In each image, the white square indicates the area in which a human face has been detected. Note that the faces in these examples are detected at different scaling factors (ζ) and rotation angles (θ). The examples presented in Figure 7 show that our proposed face detection system is capable of detecting the facial regions present in the test images. In test images containing two or more faces, as in Figure 7(e) and 7(g), our face detection system also manages to detect the faces correctly. It can also be noted from Figure 7(d) and 7(e) that the faces in the input image are slightly tilted but the system can still detect the face correctly without having to rotate the input image. This shows that the proposed system is not very sensitive to the rotation angle parameter (θ). Finally, it can be seen from these examples that the faces in the test images have different expressions but our system can detect these faces correctly. In our experiments, the proposed system still produced some missed detections and false alarms. Figure 7(f) is an example of an image in which a missed detection occurred, since the face of the subject on the left is not detected. Figure 7(g) shows an example of the occurrence of false alarm, in which a non-facial region is detected as a face. Generally, non facial regions with a lot of texture (particularly vertical patterns) tend to be misclassified by the algorithm.

Overall, our system correctly detected 183 faces out of 200 faces in the test image set for a detection rate of 91.5%. In our experiments, we encountered a total of 17 missed detections and 38 false alarms. The number of false alarms is quite low (approximately 17%), despite the fact that the proposed system is biased towards false alarms as explained in Section 3.2. These results show an improvement compared to our previous work [12]. In our previous work, the system gives a detection rate of 86.3% and a false alarm rate of 39.7%. While the detection rate shows modest improvement, the performance of the system with respect to the false alarm rate shows significant improvement. This is due to the fact that in the work presented in this paper, we add the local histogram correlation as a differentiating feature.

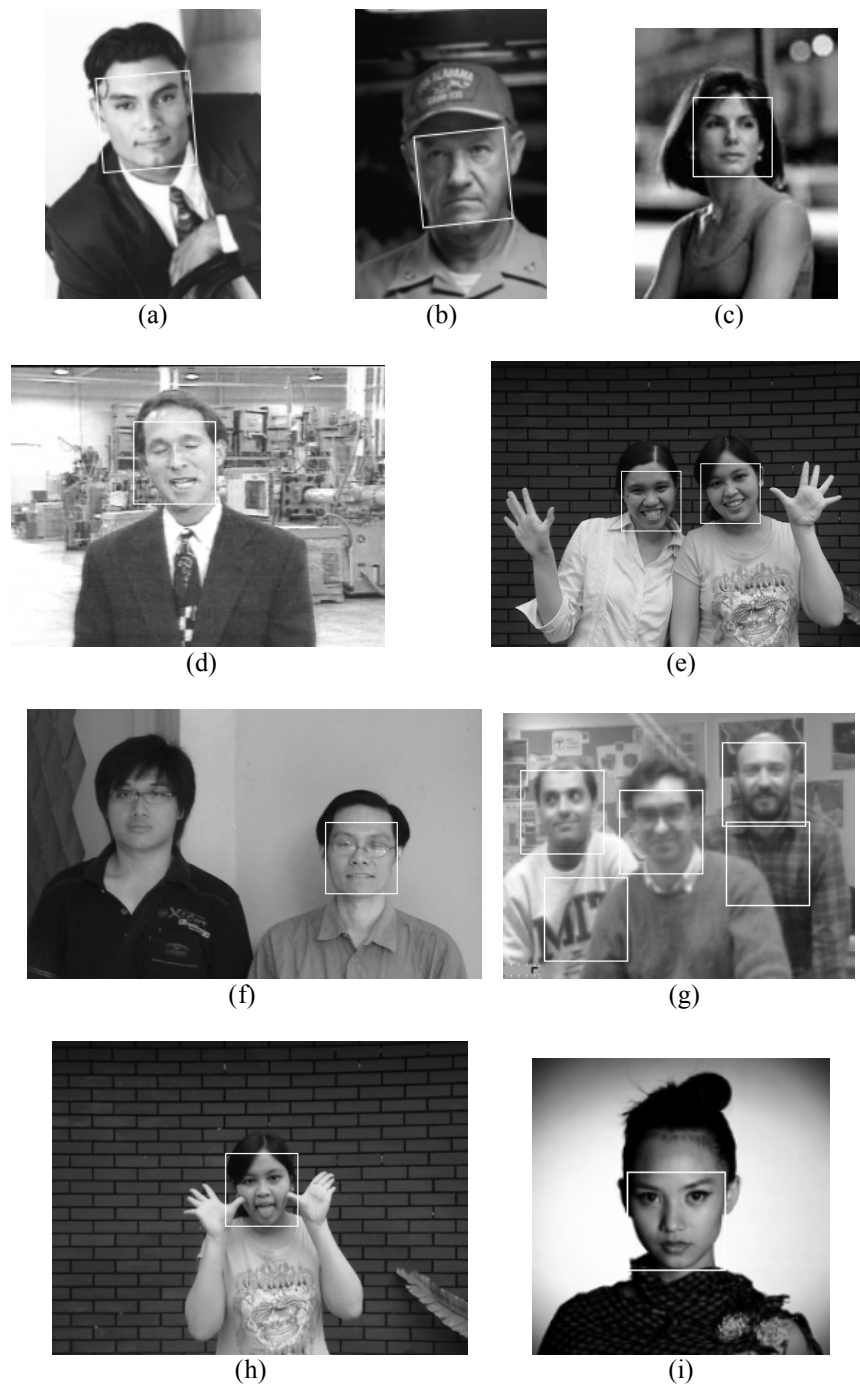


Figure 7 Examples of the detection results.

We also observe that while our system is not very sensitive to the parameter θ , it is particularly sensitive to the choice of the scaling factor, ζ . Incorrect choice of ζ may result in the face(s) in an input image not being detected or other areas of the input image not containing faces to be misclassified. The sensitivity of the system to this parameter is shown by the following experiment. We choose an input image and then proceed to detect the face in the image. We vary the values for the parameter ζ from 0.5 to 2.0. We observe that the best performance of the system can only be achieved for $1 \leq \zeta \leq 1.2$. This is shown in Figure 8(a). When we set $\zeta = 0.9$, the detector does not detect the whole facial area correctly, as shown in Figure 8(b). For $\zeta < 0.9$, the detector misses the face entirely. When we set $\zeta > 1.2$, the detector still detects parts of the face area but it also produces a high number of false alarms. This is shown in Figure 8(c). Similar behaviour is observed in other test images, although the exact value of the parameter ζ may be different in each case.

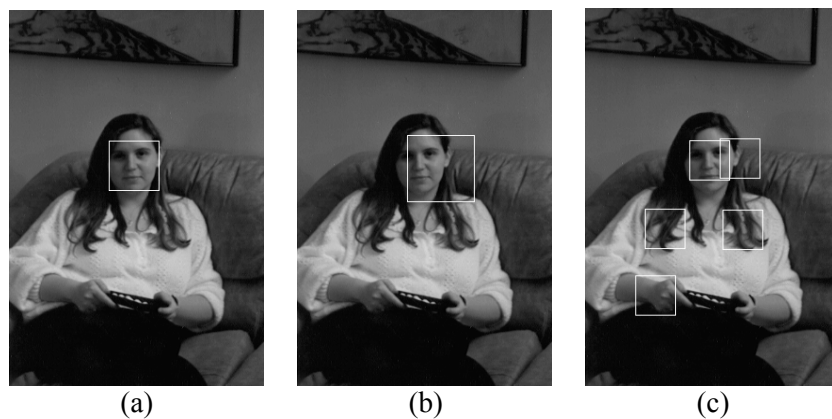


Figure 8 Detection result using different scaling factors: (a) $\zeta = 1$, (b) $\zeta = 0.9$ and (c) $\zeta = 1.4$.

Another example of the sensitivity of the system to the scaling factor is observed when the input image contains two faces that, due to the position of the subjects, have different apparent sizes. In this case, the system cannot detect both faces simultaneously, since different scaling factors must be used for each face. These are shown in Figure 9(a) and 9(b). In Figure 9(a), we set $\zeta = 0.8$ to correctly detect the subject at the front (larger apparent size). In this example, the system misses the subject at the back. To detect the subject at the back (smaller apparent size), we use $\zeta = 1.3$. This is shown in Figure 9(b). In this case, the system cannot correctly detect the subject at the front. The system only detects part of the face of the subject at the front. The area covered by the “detection box” does not really represent a face, so we can declare this a false detection.

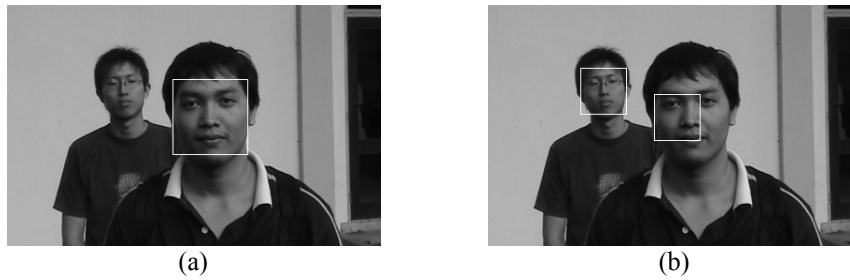


Figure 9 Images containing faces of different apparent sizes: (a) detection using $\zeta = 0.8$ and (b) detection using $\zeta = 1.3$.

5 Conclusions and Future Work

Based on the result of our experiments, we can conclude that the proposed face detection system gives a satisfactory result. The system can detect faces in test images that contain no part of the reference images. The proposed system also gives satisfactory performance with regard to the false alarm rate. The system shows a robustness against variations in background, illumination and subject expression. While the system is only trained to detect upright, frontal face images, it has shown an ability to detect faces that are slightly tilted or posed (i.e., not fully frontal). The proposed system is found to be quite sensitive to the choice of the scaling factor, ζ , but is not particularly sensitive to the rotation angle (θ). The addition of local histogram similarity as a differentiating feature in the system proposed in this paper has been shown to yield a significant performance improvement with regard to the false alarm rate compared to our previous work [12].

In our future work, we are going to improve the system in the following areas. First, we are going to add a new reference, namely the profile face reference. By including this reference, our system will be able to detect not only frontal faces, but also profile faces. We will also further improve the efficiency of the algorithm. This is particularly important to enable the proposed system to be used in a real time application. Since a facial region can appear anywhere in a test image, there is no choice but to perform an exhaustive search on the image. This approach is very computationally intensive. However, since it is safe to assume that facial regions do not overlap, we can develop a heuristic method to skip those areas that are unlikely to contain another facial region in order to increase the efficiency of the algorithm. Another possible strategy that we are going to investigate is to perform the search at a lower resolution than the original input image.

Acknowledgement

The work is supported by the Higher Education Directorate of the Ministry of Education under grant number: 461/SP2H/PL/E5.2/DITLITABMAS/IV/2011.

References

- [1] Li, S.Z. & Jain, A.K., *Handbook of Face Recognition*, Springer Science+Media, Inc., 2005.
- [2] Feris, R.S., et al., *Detection and Tracking of Facial Features in Video Sequences*, Lecture Notes in Artificial Intelligence 1793, Springer-Verlag Press, pp. 197-206, 2000.
- [3] Sao, A.K. & Yegnanaarayana, B., *Template Matching Approach for Pose Problem in Face Verification*, in Proc. MRCS 2006, Springer-Verlag Berlin Heidelberg, pp. 191-198, 2006.
- [4] Wang, L., et al., *Face Tracking using Motion-Guided Dynamic Template Matching*, in Proc. The 5th Asian Conference on Computer Vision, pp. 23-25, 2002.
- [5] Park, C.W. & Park, M., *Fast Template-based Face Detection Algorithm using Quadtree Template*, Journal of Applied Sciences, **6**(4), pp. 795-799, 2006.
- [6] Schneiderman, H. & Kanade, T., *A Statistical Method for 3D Object Detection Applied to Faces and Cars*, in Proc. IEEE Conf. Computer Vision and Pattern Recognition, pp. 746-751, 2000.
- [7] Liu, C., *A Bayesian Discriminating Features Method for Face Detection*, IEEE Trans. Pattern Analysis and Machine Intelligence, **25**(6), pp. 725-740, 2003.
- [8] Gonzales, R.C., & Woods, R.E., *Digital Image Processing*, 3rd ed., Pearson Education, Inc., 2010.
- [9] Russ, J.C., *The Image Processing Handbook*, 5th ed., CRC Press, 2007.
- [10] Khalid, M.S., et al., *Bhattacharyya Coefficient in Correlation of Gray-scale Objects*, Journal of Multimedia, **1**(1), pp. 56-61, 2006.
- [11] Buessler, J.L., et al., *Color Histogram Similarity for Robot-Arm Guiding*, in Proc. Int. Conf. Complex Systems Intelligence and Modern Technological Applications, 2004.
- [12] Setyawan, I., et al., *Face Detection System using 1-dimensional Haar Wavelet Coefficients*, in Proc. Int. Conf. Information & Communication Technology & Systems, pp. VI-35-VI-40, 2010.
- [13] Timotius, I.K., et al., *Face Recognition between Two Persons using Kernel Principal Component Analysis and Support Vector Machines*, International Journal on Electrical Engineering and Informatics, **2**(1), pp. 55-63, 2010.