Face detection based on multilayer feed-forward neural network and Haar features

Article *in* Software Practice and Experience · October 2018

DOI: 10.1002/spe.2646

3 authors, including:

Ebenezer Owusu Owusu
University of Ghana
7 PUBLICATIONS **105** CITATIONS
SEE PROFILE

Jamal-Deen Abdulai
University of Ghana
45 PUBLICATIONS **366** CITATIONS
SEE PROFILE

Received: 1 June 2018 | Revised: 28 August 2018 | Accepted: 17 September 2018

DOI: 10.1002/spe.2646

RESEARCH ARTICLE

WILEY

Face detection based on multilayer feed-forward neural network and Haar features

Ebenezer Owusu[1] | Jamal-Deen Abdulai[1] | Yongzhao Zhan[2]

[1]Department of Computer Science, University of Ghana, Accra, Ghana

[2]School of Computer Science and Communication Engineering, Jiangsu University, Zhenjiang, China

Correspondence
Ebenezer Owusu, Department of Computer Science, University of Ghana, Accra, Ghana.
Email: ebeowusu@ug.edu.gh

Summary

Fast and accurate detection of a facial data is crucial for both face and facial expression recognition systems. These systems include internet protocol video surveillance systems, crime scene photographs systems, and criminals' databases. The aim for this study is both improvement of accuracy and speed. The salient facial features are extracted through Haar techniques. The sizes of the images are reduced by Bessel down-sampling algorithm. This method preserved the details and perceptual quality of the original image. Then, image normalization was done by anisotropic smoothing. Multilayer feed-forward neural network with a back-propagation algorithm was used as classifier. A detection accuracy of 98.5% with acceptable false positives was registered with test sets from FDDB, CMU-MIT, and Champions databases. The speed of execution was also promising. An evaluation of the proposed method with other popular detectors on the FDDB set shows great improvement.

KEYWORDS

anisotropic smoothing, Bessel down-sampling, face detection, Haar features, multilayer feed-forward neural network (MFNN)

1 | INTRODUCTION

Face detection is an important stage in face recognition, facial expression, and emotion recognition applications. These technologies have received massive attention for various reasons. A typical instances is in the area of security. An example is combatting crime such as in retail organizations, where known organized retail criminals and shoplifters are monitored and arrested. Another example is found in transportation. In airports, it might be a watch list of terrorists and fugitives wanted by Interpol.

Lots of techniques are employed in face detection[1-4] but many of them have serious drawbacks. Most of the procedures employed in this work are known in other applications and they have drawbacks too, but we focused on how to minimize the challenges and maximize the advantages to enhance performance. The essential methodology includes facial feature extraction, image reduction, normalization, and classification by multilayer feed-forward neural network (MFNN).

Let us take a close look at facial feature extraction. Several methods are advanced in recent studies. These include distance-based methods and patch-based methods. Distance-based feature extractions are one of the most applied techniques for both two-dimensional (2D) and three-dimensional (3D) static faces. Among the popular ones are

feature extraction by Bhattacharyya distance[5] and Kullback Leibler[6] divergence. Despite some advantages of this method, the computational complexity is too high, resulting to misclassifications and consequently poor accuracy. The matching of even a small model shape with a normal image can take half an hour on an eight-processor Sun SPARCServer 1000.[7,8]

The patch-based methods that extract point clouds are also popular but have numerous drawbacks. Particular representations cannot be applied to other solutions without major modifications. The majority of the techniques have only been utilized to a single class. In addition, most methods do not exploit the large amounts of available training data.[9] Instead, they have mostly been examined using limited databases, some of which contain images that are not typical of the real environment. It has been revealed that the performance of most methods drop sharply when tested on images captured in uncontrolled environments.[10]

The Haar transform is the simplest of all wavelet transforms, however, it is very appealing for digital image feature extraction.[11] The most outstanding object of its desirable features is its robustness and nonrigorous computing requirements. A closer assessment of the local feature patterns in face images indicates that, typically, they contain fairly simple patterns with strong contrast. The 2D source images of Haar wavelets match very well with these patterns, making them very attractive for signal representation. The simple form of the Haar wavelets is another great advantage for real-time implementation. Additional benefit realized from applying the Haar wavelet transformation lies in its capacity to compress images as well as its applications to progressive transmission.[12] The Haar wavelets have been applied for facial feature extraction in face recognition, facial expression recognition, and other pattern recognition and image processing techniques[13-15] and we find it very useful. The Haar wavelet extraction is very fast.[16,17] However, the main problem with it is that the wavelets are too huge to result to effective classification when used as input data for facial expression classifiers[18] and its related applications that involve the extraction of huge facial data.

In this study, the extracted Haar features are reduced in dimensions by Bessel down sampling;[19] a method that is effective in preserving images' perceptual quality. Notwithstanding resizing, we performed illumination normalization through anisotropic smoothing.[20] The combined procedure enhanced the robustness to illumination variations.

The processed facial features are then input into a feed-forward neural network classifier. The MFNN remains the most popular and most extensively utilized models in many realistic applications. The technique is supplied with both sample inputs and expected outputs. The expected outputs are compared against the actual outputs for a specified input. By means of the expected outputs, the back-propagation training algorithm obtains a computed error and adjusts the weights of the different layers backwards from the output layer to the input layer. The remaining part of this work explains in detail how the study is performed.

Section 2 discusses some related studies conducted in this field. Image acquisition and preprocessing are presented in Section 3. In Section 4, the facial feature extraction by Haar feature extraction is examined. Down sampling of images is discussed in Section 5. The detail discussion of classification by MFNN is presented in Section 6. The experimental results and analysis are discussed in detail in Section 7. Experimental findings is summarized in Section 8. Finally, the conclusion of the study is presented in Section 9.

2 | RELATED WORK

There have been lots of studies about face detection and image processing techniques. Though space will not permit a thorough review, a few popular methods are worth reviewing.

Voila-Jones descriptor[21] introduced Haar wavelet image representation and AdaBoost algorithm for a 2D frontal face detection. Throughout this method, special attention was paid to increase the speed of the detector, hence a cascade of classifiers was designed to perform that function. However, the detection accuracy of 95% is not scientifically sound for successful implementation in today's technology.

Rowley et al[22] also presented a neural network classifier to distinguish facial image from nonfacial image. This system relies on multiple networks to increase performance on a CMU test set of 130 images but the detection accuracy was just 90.5%.

Gupta et al[23] developed face detection based on Gabor features and neural network. In that method, the authors resorted to manual deletion of some images to reduce time complexity. Mechanical deletion of images has a potential of removing better training samples and therefore compromising accuracy.

Neural networks have been used extensively in various forms of facial image applications[1,24,25] but what basically makes the difference is the image processing techniques. In this proposed method, superior and robust imaging processing techniques are employed to ensure better results.

Finally, unlike previous works, special caution is taking to train the neural network classifier with several types of images from different databases to enhance performance.

3 | IMAGE ACQUISITION AND PREPROCESSING

Several images from different databases were assembled for this work. We used 800 positive examples from MultiPIE and 1100 negative images from the Inria Person database[26] to train the classifier. The negative Inria Person images are outdoor scenes that do not contain human objects. The CMU MultiPIE face database[27] contains about 750 000 images of 337 subjects under multiple viewpoints, different expressions, and illumination conditions. To avoid biases in testing of the system, we collected unknown datasets from CMU-MIT database and 236 colored images from Champions database. Each image from Champions datasets involves one face. We also used 2845 samples from the FDDB to test for the performance of the system. The CMU-MIT images contain a total of 507 frontal faces and require the networks to examine 83 099 211 25×15 pixel windows. We adopted the evaluation protocol for the FDDB dataset[28] to compare our results with others. All queried images were automatically resized to a window of size 25×15 and converted into grayscale images. Figure 1 shows sample database images used for experiment.

FIGURE 1 Sample database images for experiment. A, Sample images from Champion database; B, Sample images from FDDB database; C, Sample images from CMU MultiPIE database; D, Sample images from Inria Person database [Colour figure can be viewed at wileyonlinelibrary.com]

4 | FEATURE EXTRACTION METHOD

The Haar wavelet transformation involves the computation of the averages of two values as well as the differences of the elements in rows and columns of a matrix with respect to other adjacent elements.[12] The feature extraction procedure is presented as follows.

Step 1: Let the integral of image with pixels P be defined as follows:

$$I_p(u,v) = \sum_{x-1}^{u}\sum_{y-1}^{v} P(x,y), \tag{1}$$

for block B situating at the top-left corner (x_1, y_1) and bottom-right corner.

Step 2: The total pixel intensity in this block is computed by the expression

$$S(B) = I(x_1, y_1) + I(x_2, y_2) - I(x_2, y_1) - I(x_1, y_2). \tag{2}$$

Step 3: A "squared" integral image I_q is attained by interchanging $P(x,y)$ in Equation (1). This facilitates first computation of the block variance σ_B.

Step 4: For pixel feature block B, the subsequent Haar-wavelet coefficient can be determined by the expression

$$H(u,v) = \frac{\sum_{i=1}^{N_a}\left\{\mathrm{Sgn}(B_i)\sum_{x}^{M_{B_i}}\sum_{y=1}^{M_{B_i}}[B_i(x,y)-\mu_B]\right\}}{N(u,v).\sigma_B} = \frac{1}{N(u,v).\sigma_B}\sum_{i=1}^{N_B}\left[\mathrm{Sgn}(B_i).S(B_i)\right], \tag{3}$$

where B_i represents the subblocks corresponding to nonzero coefficients areas. N_B is the number of subblocks and $\mathrm{Sgn}(B_i)$ represents the sign of the coefficient to the subblock B_i. The size of B_i is denoted by $M_{B_i} \times M_{B_i}$. Due to the simple fact that the coefficients $H(0,0)$ only hold the average intensity value of the block, it is zero for all illumination-corrected block images and therefore can be ignored during matching.

For all residual basis images, the sum area of +1 signed subblocks is equivalent to the area of −1 signed subblocks. $H(u,v)$ is the normalization factor.

The whole subblock data can be determined by using the integral image, making $S(B_i)$ occurs in the second expression of Equation (3).

5 | DOWN SAMPLING AND NORMALIZATION OF IMAGE

5.1 | Down-sampling of image

Considering the large sizes of images, we noted that it would take a lot of time for the classifier to complete the training process. Hence, for the second time, we scaled down the size of the images to a reasonable size. Methods like bilinear interpolations have long been used by several authors but considering the fact that interpolations are not very suitable in image reductions as they are prone to aliasing problems we resorted to Bessel down sampling. This method reduced the size of the image and also preserved the details and perceptual quality of the original image. The method presented by Ganga et al[29] was applied to this work. Assuming the size of an image I is $(a \times b)$ and its new size after reduction is $((a\text{-}m) \times (b\text{-}n))$, where m and n are positive integers such that $m < a$ and $n < b$, the Bessel coefficient $c(m_1,m_2)$ is computed as follows:

$$c(m_1,m_2) = \frac{4\int_0^a\int_0^b t_1 t_2 x(t_1,t_2) J_0\left(\frac{\alpha_{m_1}}{a}t_1\right) J_0\left(\frac{\alpha_{m_2}}{b}t_2\right) dt_1 dt_2}{(ab)^2\left[J_1\left(\alpha_{m_1}\right) J_1\left(\alpha_{m_2}\right)\right]^2}, \tag{4}$$

where $x(t_1,t_2)$ is a finite duration signal in the interval $0 \le t_1 \le a$ and $0 \le t_2 \le b$. The signal $x(t_1,t_2)$ is represented by the zero-order Bessel functions of the first kind in an infinite Bessel series expressed as follows:

$$x(t_1,t_2) = \sum_{m_1=1}^{a}\sum_{m_2=1}^{b} c(m_1,m_2) J_0\left(\frac{\alpha_{m_1}}{a}t_1\right) J_0\left(\frac{\alpha_{m_2}}{b}t_2\right). \tag{5}$$

The variables are the roots of $J_0(t_1) = 0$ and $J_0(t_2) = 0$. $m_1 = \{1, 2, 3, \ldots a\}$ and $m_2 = \{1, 2, 3, \ldots, b\}$. The parameter $J_1(\alpha_{m_1})$ and $J_1(\alpha_{m_2})$ are first-order Bessel functions. The first-order Bessel function $J_v(z)$, is the solution of the Bessel's differential

equation of the form

$$z^2\frac{d^2y}{dz^2}+z\frac{dy}{dz}+(z^2-v^2)=0, \tag{6}$$

where z is a real constant. The solution is defined by its Taylor series expansion around $z=0$, hence the function can be expressed as

$$J_v(z)=\left(\frac{1}{2}z\right)^v\sum_{k=0}^{\infty}\frac{\left(\frac{-1}{4}z^2\right)^v}{k!\Gamma(v+k+1)}. \tag{7}$$

The method of reducing an image of size $a\times b$ to a size of $(a-m)\times(b-n)$ is down sampling, where m and n are any positive integers such that $m<n$ and $n<b$. In this study, the size of images are reduced to one-fourth of original size. The final down-sampling image is expressed as follows:

$$x_d(t_1,t_2)=\sum_{m_1=1}^{a}\sum_{m_2=1}^{b}c(m_1,m_2)\,J_0\left(\frac{\alpha_{m_1}}{a-m}t_1\right)J_0\left(\frac{\alpha_{m_2}}{b-n}t_2\right) \tag{8}$$

$$0\le t_1\le a-m, 0\le t_2\le b-n.$$

5.2 | Normalization of image

Region normalization followed down sampling. Each image is normalized to unity to provide uniform illumination. In general, an image $I(x, y)$ is assumed to be the product of the reflectance, ie, $R(x, y)$, and the luminance, ie, $L(x, y)$. This follows that $I(x, y) = R(x, y).L(x, y)$. Computing the reflectance and the luminance fields of real images is, in general, an ill-posed problem. Hence, numerous hypotheses are proposed to calculate L and R or both. According to Gross and Brajovic,[30] the solution for $L(x, y)$ is determined by the following:

$$J(L)=\iint_{\Omega}\rho(x,y)(L-1)^2dxdy+\lambda\iint_{\Omega}\left(L_x^2+L_y^2\right)dxdy, \tag{9}$$

where the first term drives the solution to follow the perception gain model and the second term imposes a smoothness constraint. Ω refers to the image. The differential equation can be solved by the application of the Euler-Langrange method. Our illumination normalization was based on anisotropic smoothing proposed by Kim et al.[20] The space varying permeability weight, ie, $\rho(x, y)$, controls the anisotropic nature of the smoothing constraint. The image $I(x, y)$ is normalized to form the new image $I(x, y)^{\text{new}}$. This normalized image $I(x, y)^{\text{new}}$ is expressed as follows:

$$I(x,y)^{\text{new}}=\text{scale}\left(\alpha R(x,y)+Heq\left(L(x,y)\right)\right), \tag{10}$$

where $R(x,y) = I(x,y)/L(x,y)$ and Heq denotes histogram equalization, ie, $\alpha > 1$. In this work, we chose $\alpha = 2$. The final normalized image is shown in Figure 2.

6 | FACE DETECTION BY MULTILAYER FEED-FORWARD NEURAL NETWORK

The most challenging component of a robust face detection system is the efficiency of the classifier.

The system accepts an input of 25×15 pixel region of the image and generates an output of 1 for a face and -1 for a nonface. The initiative of making the classifier fast is addressed by decreasing the searching space but preserving the details and perceptual quality of the image. The architecture is a multilayer neural network with one hidden layer and a back-propagation algorithm to train the network. The input layer which served as receiving the Haar features has nodes equal in dimension as the feature vectors. The output layer has one node. The number of epochs for this experiment was 1000 and the goal was 0.001. We assume that the training samples are represented by $\{p_1,t_1\},\{p_2,t_2\},\ \ldots,\{p_Q,t_Q\}$, where $p_q=[p_{q1},p_{q2},\ \ldots,p_{qr},\ \ldots,p_{qR}]^T\in\mathfrak{R}^R$ is an input to the network. All the pixel values are read line by line to form an input pattern, which is a column vector in $\mathfrak{R}^R$ dimensional space, R represents the total face image, p_{qr} represents the intensity value of rth feature value in the qth face image. The network is trained to produce an output of 1 for the face samples and -1 for the nonface samples. The training algorithm is standard error back-propagation. The process of training involves weight initialization, calculation of the activation unit, weight adjustment, weight adaptation, and

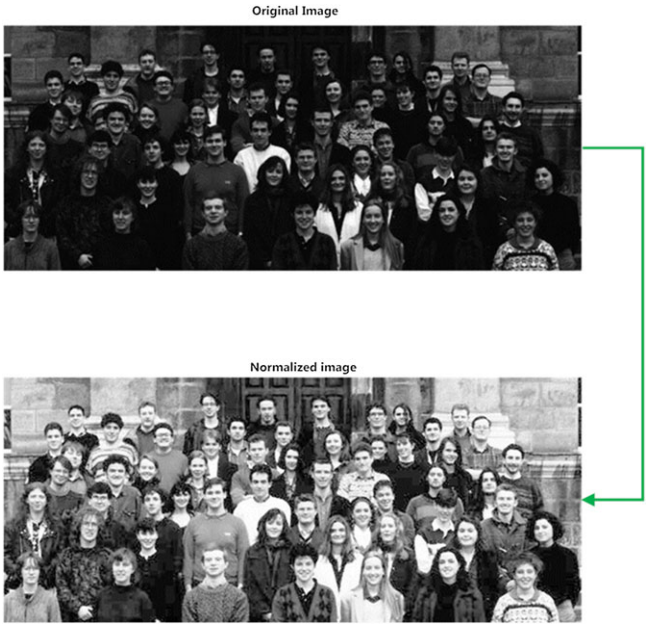

FIGURE 2 Original image (top), normalized image (down) [Colour figure can be viewed at wileyonlinelibrary.com]

convergence testing. All weights were initially set to small random values. Assume v_{ji} represents the weight between the jth hidden unit and ith input unit; and w_{kj} representing the weight between the kth output and the jth hidden unit. The activation unit is calculated sequentially, starting from the input layer. The activation of hidden and output unit is calculated as follows:

$$y_j^{(p)} = f_{yj}^{(p)}\left(\sum_{i=1}^{I} v_{ji} z_i - v_{jo}\right) \tag{11}$$

$$o_k^{(p)} = f_{ok}^{(p)}\left(\sum_{j=1}^{J} w_{kj} y_j - w_{ko}\right), \tag{12}$$

where $y_j^{(p)}$ is the activation of the jth hidden unit and $o_k^{(p)}$ is the activation of the kth output unit for the pattern, ie, p. f is a sigmoid function. In this work, we scaled the inputs to $[-1, 1]$. k is the total number of output units, I is the total number of input units, and J is the total number of hidden units. v_{jo} is the weight connected to the bias unit in the hidden layer. We adjusted the weights, starting at the output units, and recursively propagated error signals to the input layer. The detected output $o_k^{(p)}$ is compared with the corresponding target value $t_k^{(p)}$, which is a face image over the entire training set using the sigmoid function to express the approximation error in the network's target functions

$$E^p = \frac{1}{2}\sum_{k=1}^{K}\left(t_k^{(p)} - o_k^{(p)}\right)^2. \tag{13}$$

The minimization of the error $E^{(p)}$ requires the partial derivative of $E^{(p)}$ with respect to each weight in the network to be computed. The change in weight is proportional to the corresponding derivative

$$\Delta v_{ji}(t+1) = -\eta\frac{\partial E^{(p)}}{\partial v_{ji}} + \alpha\Delta v_{ji}(t) \tag{14}$$

$$\Delta w_{kj}(t+1) = -\eta\frac{\partial E^{(p)}}{\partial w_{kj}} + \alpha\Delta w_{kj}(t), \tag{15}$$

where η is the learning rate between 0 and 1, we set it to 0.9. The function α is also set to 0.9. The last term is a function of the previous weight change

$$\frac{\partial E}{\partial v_{ji}} = \frac{\partial y_j}{\partial v_{ji}}\sum_{k=1}^{K} -(t_k - o_k)\, o_k\,(1 - o_k)\, y_j w_{kj}, \tag{16}$$

$\frac{\partial y_j}{\partial v_{ji}} = y_j(1 - y_j)z_i$. Therefore,

$$\Delta v_{ji} = \eta \sum_{k=1}^{K} (t_k - o_k)o_k(1 - o_k)y_j w_{kj} y_j(1 - y_j)z_i. \tag{17}$$

The weights are updated by

$$w_{kj}(t + 1) = w_{kj}(t) + \Delta w_{kj}(t + 1) \tag{18}$$

$$v_{ji}(t + 1) = v_{ji}(t) + \Delta v_{ji}(t + 1), \tag{19}$$

where t is equal to the current time step. Δv_{ji} and Δw_{kj} are the weight adjustments. We repeated the process once from step (10) to achieve the desired output. The output of this network has been just one so we have just one neuron at the output. In this work, we added neuron bias to the network to offset the origin of the activation functions so as to make room for rapid convergence of the training process. The bias has a constant activation value −1. We trained the bias weights the same way as we did for other weights. The functions that trained the hidden and the output units are given in (17) and (18), respectively,

$$y_j = f\left(\sum_{i=0}^{I} v_{ji} z_i\right) \tag{20}$$

$$o_k = f\left(\sum_{j=0}^{I} w_{kj} y_j\right), \tag{21}$$

$z_o = -1$ and $y_o = -1$. v_{jo} is the weight to the bias unit y_o in the hidden layer. In this work, we scaled the inputs to $[-1, 1]$. A green bounding box of size 25×15 is drawn around a detected face. Figure 3 shows the processes involved in searching for the potential face region.

7 | EXPERIMENTAL RESULTS AND ANALYSIS

The network is trained with 800 positive sets from the MultiPIE and 1100 negative sets from the IN-RIAPerson databases. Testing is performed with 2845 images from FDDB, 507 images of the CMU-MIT database, and 236 colored images from the Champions database. The network relies on the selected features. Vectors of extracted facial features consisting of

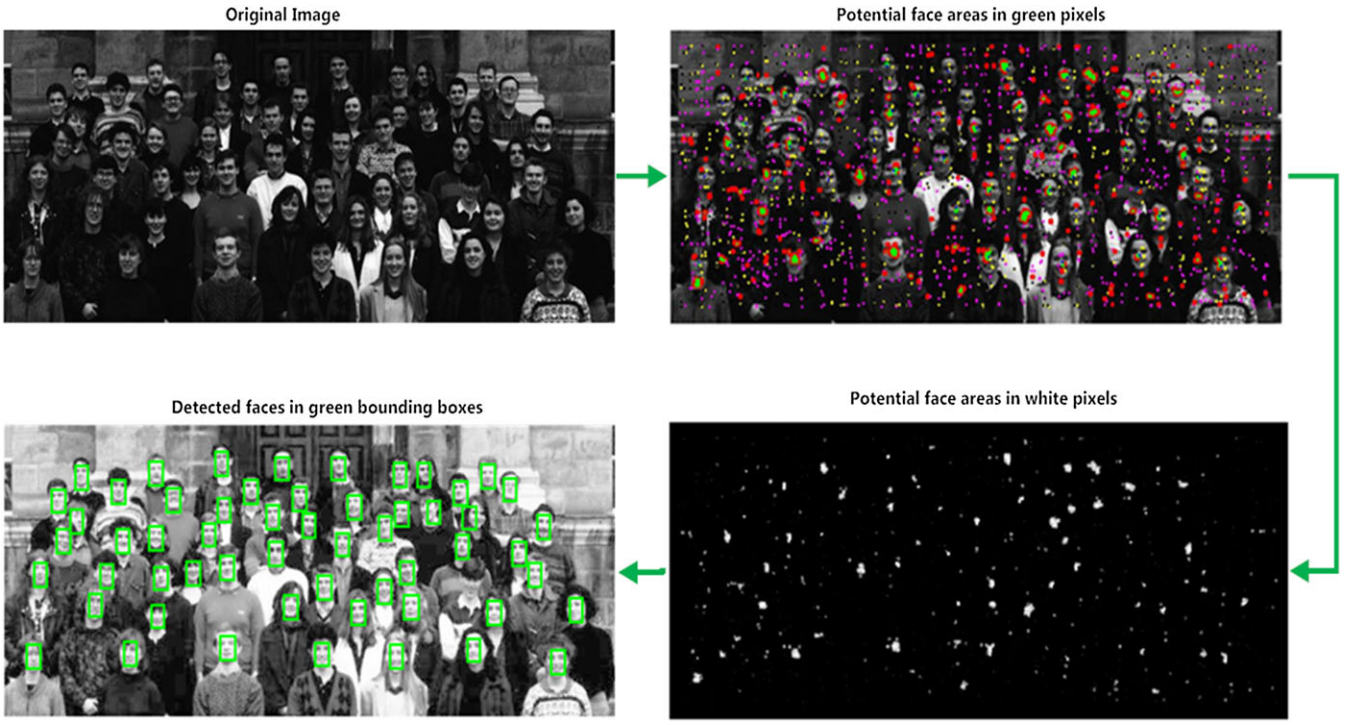

FIGURE 3 Facial image detection process [Colour figure can be viewed at wileyonlinelibrary.com]

TABLE 1 The detection accuracy of different methods on the MIT-CMU dataset

Detector	Detection Rate (%)	False Detection
Viola-Jones[31]	88.4	31
Lin Huan[32]	88.3	57
Jiang et al[33]	87.6	6
Schneiderman-Kanade[34]	94.4	65
Viola-Jones[21]	93.1	78
Roth et al[35]	94.8	78
Owusu et al[3]	97.6	53
Proposed method	98.5	47

TABLE 2 The execution time of different methods

Author	Machine Specification	Image Size	Processing Time/s
Jiang et al[33]	2.67 GHz CPU NVidia GTX-465	640 × 480	0.0593
Rowley et al[22]	200 MHz R4400 SGI Indigo 2	320 × 240	7.2
Owusu et al[3]	2 GHz Intel Pentium IV	320 × 240	0.0285
Proposed method	2 GHz Intel Pentium IV	320 × 240	0.0201

FIGURE 4 Sample results of detected faces [Colour figure can be viewed at wileyonlinelibrary.com]

eyes, nose, peaks, and hollows are fed to the input layers of the network. Correct detection rates and false positives are computed from the testing results of MIT-CMU samples. Table 1 compares the detection accuracy of different methods on the MIT-CMU dataset. Table 2 gives indications of the speed of execution in different studies though they do not indicate direct comparison due to different set-ups except with our previous study.[3]

The best results are observed with frontal faces and a maximum of 15° yaw rotations. We intend to incorporate head rotation algorithms to investigate detections in head rotations of 45° in future study. Figure 4 shows sample detections of images and Figure 5 shows the receiver operating characteristic (ROC) curves of different detectors in the FDDB sets. The plot clearly indicates that the proposed method outperforms some popular detectors.

8 | DISCUSSION OF RESULTS

The outcome of the experiment demonstrates unequivocally that the proposed method outperforms the popular descriptors in terms of processing time and detection accuracy. It is established that the windows around the cheeks provide good grounds for nonfrontal face detection. Furthermore, if we intend to increase the speed of the classifier, it is important we focus more on how to decrease the searching space and preserve details of perceptual quality of the image. Unlike other descriptors, the method is trained and tested with different types of images of which some depict real-life scenarios. It is clear that, with effective and robust image processing technique, neural network offers one of the best classification approach. An observation of the ROC curve in Figure 5 clearly indicates this point.

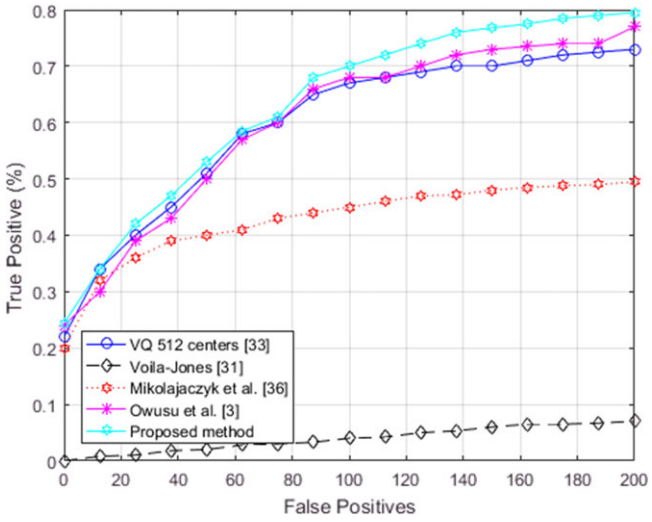

FIGURE 5 The receiver operating characteristic (ROC) curves of different detectors on FDDB set [Colour figure can be viewed at wileyonlinelibrary.com]

9 | CONCLUSIONS

In this paper, we have proposed a face detection method based on Haar feature extraction and classification by MFNN with a back-propagation algorithm. The images were normalized by anisotropic smoothing and down-sampled by Bessel methods. The feature vectors were input to an MFNN with a back-propagation algorithm for training the network. The training was done with 800 positive sets from the MultiPIE and 1100 negative sets from the Inria Person. Testing was performed with some 2845 images in FDDB, 507 images from CMU-MIT, and 236 colored images from Champions database. The result of the study indicates that the proposed method outperformed popular methods built with the same datasets. The FDDB samples were used as the benchmark for comparison through ROC assessments.

This approach has outstanding strengths when compared with others. First, the image processing technique is competent enough to maintain the perceptual quality of the digital images. Second, the network is trained and tested with wide variety of images, from both Caucasian and non-Caucasian and from both premeditated and real backgrounds, making it more robust for practical implementation. However, the limitations of the study has to do with detection in severe head rotations of the magnitude of 45°. Moreover, the most threats to validity has to do with the old problem facing face detection and recognition study, that is, lack of a common standard face detection database for testing. Lack of common testing benchmark makes direct comparison difficult to execute.

In order to further reduce computational complexity and increase the speed of detection, future improvements of the system will focus on the possibility of integrating skin detection algorithms to reduce searching complexity. Another consideration is the variation of the window size of the detector to enhance accuracy.

ORCID

Ebenezer Owusu http://orcid.org/0000-0002-4670-1342

REFERENCES

1. Owusu E, Zhan Y, Mao QR. A neural-AdaBoost based facial expression recognition system. *Expert Syst Appl*. 2014;41(7):3383-3390.
2. Owusu E, Zhan Y, Mao QR. An SVM-AdaBoost facial expression recognition system. *Appl Intell*. 2014;40(3):536-545.
3. Owusu E, Zhan Y, Mao QR. An SVM-AdaBoost-based face detection system. *J Exp Theor Artif Intell*. 2014;26(4):477-491.
4. Parke FI, Waters K. *Computer Facial Animation*. Boca Raton, FL: CRC Press; 2008.
5. Choi E, Lee C. Feature extraction based on the Bhattacharyya distance. *Pattern Recognit*. 2003;36(8):1703-1709.
6. Polani D. Kullback-Leibler divergence. In: *Encyclopedia of Systems Biology*. New York, NY: Springer; 2013:1087-1088.
7. Huang H-F, Tai S-C. Facial expression recognition using new feature extraction algorithm. *ELCVIA: Electron Lett Comput Vis Image Anal*. 2012;11(1):0041-0054.
8. Rucklidge WJ. Efficiently locating objects using the Hausdorff distance. *Int J Comput Vis*. 1997;24(3):251-270.

9. Aghajanian J, Warrell J, Prince SJ, Li P, Rohn JL, Baum B. Patch-based within-object classification. Paper presented at: 12th IEEE International Conference on Computer Vision; 2009; Kyoto, Japan.
10. Mäkinen E, Raisamo R. An experimental comparison of gender classification methods. *Pattern Recognit Lett*. 2008;29(10):1544-1556.
11. Chang C-C, Chuang J-C, Hu Y-S. Similar image retrieval based on wavelet transformation. *Int J Wavelets Multiresolution Inf Process*. 2004;2(02):111-120.
12. Mulcahy C. Image compression using the Haar wavelet transform. *Spelman Sci Math J*. 1997;1(1):22-31.
13. Van Fleet PJ. *Discrete Wavelet Transformations: An Elementary Approach with Applications*. Hoboken, NJ: John Wiley & Sons; 2011.
14. Gonzalez RC. *Digital Image Processing*. London, UK: Pearson Education; 2016.
15. Moharir PS. *Pattern-Recognition Transforms*. Boston, MA: Research Studies Press; 1992.
16. Zalmanzon LA. *Fourier, Walsh and Haar Transforms and Their Application in Control, Communication and Other Fields*. Moscow, Russia: Science Publisher; 1989.
17. Satiyan M, Nagarajan R. Recognition of facial expression using Haar-like feature extraction method. Paper presented at: IEEE 2010 International Conference on Intelligent and Advanced Systems (ICIAS); 2010; Kuala Lumpur, Malaysia.
18. Sarode N, Bhatia S. Facial expression recognition. *Int J Comput Sci Eng*. 2010;2(5):1552-1557.
19. Chadha AR, Vaidya PP, Roja MM. Face recognition using discrete cosine transform for global and local features. Paper presented at: 2011 International Conference on Recent Advancements in Electrical, Electronics and Control Engineering (ICONRAEeCE); 2011; Sivakasi, India.
20. Kim S, Chung S-T, Jung S, Cho S. An improved illumination normalization based on anisotropic smoothing for face recognition. *Int J Comput Sci Eng*. 2008;3(2):100-106.
21. Viola P, Jones M. Rapid object detection using a boosted cascade of simple features. In: Computer Vision and Pattern Recognition (CVPR 2001). Proceedings of the 2001 IEEE Computer Society Conference, Vol. 1; 2001; Kauai, Hawaii.
22. Rowley HA, Baluja S, Kanade T. Neural network-based face detection. *IEEE Trans Pattern Anal Mach Intell*. 1998;20(1):23-38.
23. Gupta B, Gupta S, Tiwari AK. Face detection using Gabor feature extraction and artificial neural network. In: Proceedings of the International Symposium on Computer Engineering & Technology (ISCET); 2010; Punjab, India.
24. Khatun A, Bhuiyan M. Neural network based face recognition with Gabor filters. *Int J Comput Sci Netw Secur*. 2011;1(11):71-74.
25. Shenoy A, Gale TM, Frank R, Davey N. Recognizing emotions by analyzing facial expressions. In: Proceedings of the UK Workshop on Computational Intelligence (UKCI); 2007; London, UK.
26. Dalal N, Triggs B. Histograms of oriented gradients for human detection. Paper presented at: IEEE Computer Society Conference on Computer Vision and Pattern Recognition; 2005; San Diego, CA.
27. Gross G, Matthews I, Cohn J, Kanade T. Multi-pie. *Image Vis Comput*. 2010;28(5):807-813.
28. Berg L, Berg AC, Edwards J, et al. Names and faces in the news. In: Proceedings of the 2004 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, Vol. 2; 2004; Washington, DC.
29. Mohan GP, Prakash C, Gangashetty SV. Bessel transform for image resizing. Paper presented at: 18th International Conference on Systems, Signals and Image Processing (IWSSIP); 2011; Sarajevo, Bosnia and Herzegovina.
30. Gross R, Brajovic V. An image preprocessing algorithm for illumination invariant face recognition. In: *Audio- and Video-Based Biometric Person Authentication*. Berlin, Germany: Springer; 2003.
31. Viola P, Jones MJ. Robust real-time face detection. *Int J Comput Vis*. 2004;57(2):137-154.
32. Huang L-L, Shimizu A, Kobatake H. Robust face detection using Gabor filter features. *Pattern Recognit Lett*. 2005;26(11):1641-1649.
33. Jiang F, Shi B, Fischer M, Ekenel HK. Effective discretization of Gabor features for real-time face detection. Paper presented at: 18th IEEE International Conference on Image Processing (ICIP); 2011; Brussels, Belgium.
34. Schneiderman H, Kanade T. A statistical method for 3D object detection applied to faces and cars. In: IEEE Proceedings of Computer Vision and Pattern Recognition; 2000; Hilton Head Island, SC.
35. Yang M-H, Roth D, Ahuja N. A SNoW-based face detector. In: NIPS'99 Proceedings of the 12th International Conference on Neural Information Processing Systems; 2000; Denver, CO.
36. Mikolajczyk K, Schmid C, Zisserman A. Human detection based on a probabilistic assembly of robust part detectors. In: *European Conference on Computer Vision*. Berlin, Germany: Springer; 2004:69-82.

How to cite this article: Owusu E, Abdulai J-D, Zhan Y. Face detection based on multilayer feed-forward neural network and Haar features. *Softw Pract Exper*. 2018;1–10. https://doi.org/10.1002/spe.2646