Drowsy Driver Assistant System

M. ElSabrouty, A. Hamdy, A. Fawky and S. Khalil[1]

1 Faculty of Information Engineering and Technology
Communication Department
German University in Cairo
New cairo,Cairo, Egypt.
E-mail: maha.elsabrouty@guc.edu.eg
ahmed.elsayed@student.guc.edu.eg
abdelfattah.soliman@student.guc.edu.eg
sherif.khalil@student.guc.edu.eg

***Abstract*— This paper proposes a new approach to help the drowsy drivers on the roads by developing a real time computerized system which has the ability to detect the face and to know the status of the eyes, if the eyes are closed the system strikes out an alarm to alert the driver. We used PCA and skin tone for face detection. For eye detection we used edge detection, wavelets transform and YCbCr transformation. The algorithm performs well regardless the face position, lighting varieties and immune to any distracting noise. The designed algorithm is able to attain 90% accuracy when it was tested on different databases.**

***Keywords*—Drowsy driver, edge detection, eye detection, face detection, Principal Component Analysis, skin tone , YCbCr, wavelets transform**

I. INTRODUCTION

Sleeping while driving happens more often more than most of the people think. According to the United State Department of Transportation (USDT) [11] & [14], drowsiness is responsible for 1% to 10% percent of the 20 million car accidents which occur each year in the US, therefore, developing an alarm system is a crucial task. It is thus important to develop a fast alarm system that will notify drivers upon drowsiness and yet produce as few false alarms as possible.

The human face can be utilized as a preprocessing step to limit the search for eyes within a limited space and thus reduce false detection result. Face detection is considered as the backbone of many systems as in security systems, face recognition and human interface applications. In Drowsy Driver Application, face detection has to be combined with an eye detection algorithm to get the desired performance.

Many applications were developed for face detection which is Neural Networks (NN), Principle Component Analysis (PCA), wavelets, edge detection, Hough transform, and template matching and geometrical matching are some techniques taken for major approaches. Neural networks and matching algorithms needs a huge training set of face and non-face images and take a long time in execution as discussed in [2], [7] & [18]. Principle component analysis takes only gray scale images as well as it is used for recognition as discussed in [5] [6], [10], [13], [15], [20] & [21]. Edge detection is severely affected by light condition as in [3], [4] & [8]. As for wavelets it uses a big size window so it cannot detect small face and fail in extreme lighting condition as discussed in [10] & [22]. In the other hand, eye detection techniques are not as popular as face detection techniques or it only works with certain conditions. So we had to introduce a new algorithm that could detect the eyes within the face region.

As our application is online and thus is governed by execution deadlines, we have proposed a system to improve the driver's safety by detecting the status of the eye using as PCA & skin-tone for face detection to minimize the searching region for the eyes. For eye detection, we have proposed using wavelets transformation and edge detection with the addition of YC_bC_r transformation. At last, we use Kalman filters for tracking the eyes to continuously know the eyes position and status. The whole alarm system is implemented on the DSP processor, ADSP-BF561 Blackfin Symmetric Multi-Processor for Consumer Multimedia. The microprocessor contains 512 MB of RAMs and 8 MB of flash memory.

The rest of the paper is organized as follows. Section II, presents the background on the used algorithms. We present our modifications and implementation on the used algorithms in Section III. Section IV presents the simulation and implementation results. Finally, we conclude the paper in Section V.

II. THEORETICAL BACKGROUND

A. Face Detection Algorithms

1) Principle Components Analysis

As discussed in [13], PCA starts with a random vector **x** with *n* elements, which contain a sample as second order statistics (SOS). No probability density of the vectors are made, it only matters that the first and second order statistics are known or can be estimated for vector **x**, so the elements of **x** are measurements like pixel gray levels or values of a signal at different time instants. It is important that the elements are mutually correlated and there is some redundancy in **x** which makes the compression possible.

PCA transformation starts with centering the vector **x** by subtracting its mean:

$$\mathbf{x} \leftarrow \mathbf{x} - E\{\mathbf{x}\} \tag{1}$$

This mean is estimated from the available sample **x**(1),…,**x**(T). After centering the vector, the mean E{**x**} will be zero. Then, **x** is linearly transformed to another vector **y** with *m* elements such that $m < n$, that leads to induce the redundancy as the correlations is removed. This is done by finding a rotated orthogonal coordinate system such that the elements of **x** in the new coordinates become uncorrelated. However, the variances of the projections of **x** on the new coordinate axes are maximized so that the first axis corresponds to the maximal variance; the second axis corresponds to the maximal variance in the direction orthogonal to the first axis and so on.

2) Skin-Tone Extraction

Many algorithms dealt before with the skin color and how to detect faces by finding the skin such in [9], [12], [17] & [19]. Most of these techniques were based on finding a face pattern by comparing it with a certain mask. However, they all start by finding the suitable color space such as YC_bC_r, HUV or others. YC_bC_r is the most used color space as most of the previously mentioned algorithms operate on the chrominance of the skin to extract some facial features such as the mouth or the eyes.

B. Eye Detection Algorithms

1) Edge detection

It may be one of the easiest ways for face or eye detection [16] that comes into our minds at first as it has been discussed before by many researchers. The main aim of the edge detection techniques is to find points in an input digital image at which the light intensity changes suddenly (sharply). These sharp changes in the image can reveal some important things from an image like:

- Non-uniform surface.
- Changes in the property of a material.
- Luminance change.
- Discontinuities in surface orientation.

The edge detection for an image consequently reduces the size of the image and the amount of data in it; it filters out some unnecessary information. It is easy to be implemented, because less processing bandwidth is needed, but, it is very affected by the luminance conditions, works only on grayscale images.

2) Discrete wavelets transform

In this method [1] we are suggesting a new method for eye recognition using discrete wavelets transform with 'Haar' as the mother wavelet. In spite of many researchers opinions although it is being computational intensive but it is powerful as it is based on strong mathematical concepts and relations that ensures and proves its reliability and its expected efficiency as it is robust against different lighting condition, noise and low computational coast. Wavelets are used not only in eye recognition but also in many other fields as it facilitates the analysis of a specific signal. We experienced some difficulties in dealing with wavelets and in applying some algorithms to recognize the eyes in an image. Due to the variation in luminance, different environmental conditions, visual angle and facial expressions; facial organs and features are recognized by this order: eyes, nose and mouth. Factors that must be taken in consideration: eyelashes, eyelids movements, head tilting, makeup, glasses.

a) Discretised wavelet transforms

The main reason for using this type of wavelets transformations is to analyze a specified signal to pick just a discrete subset of the upper half plane to be able to reconstruct a signal from the corresponding wavelet coefficients. The corresponding discrete subset of the half plane consists of all the points $(a^m, n\, a^m b)$ with integers while $m, n \in Z$. The corresponding baby wavelets are known as:

$$\Psi_{m,n} = a^{-\frac{m}{2}} * \Psi\,(a^{-m}t - nb) \tag{2}$$

a>1, b>0

For a reconstruction of any signal x of finite energy by the formula:

$$x(t) = \sum_{m\in\mathbb{Z}} \sum_{n\in\mathbb{Z}} \langle x, \psi_{m,n}\rangle \cdot \psi_{m,n}(t) \tag{3}$$

b) Mother wavelet

In real-life practical applications, the designers and the researchers prefer the most efficient ways, so they use continuously differentiable functions with compact support as mother (prototype) wavelet (functions). Therefore, to satisfy some analytical requirements we should choose the wavelet functions from a subspace of the space. $L^1(\mathbb{R}) \cap L^2(\mathbb{R})$, in this space the functions are both absolutely and square integrable:

$$\int_{-\infty}^{\infty} |\psi(t)|^2\, dt < \infty \tag{4}$$

$$\int_{-\infty}^{\infty} |\psi(t)|\, dt < \infty \tag{5}$$

It can be ensured that we can formulate the conditions of zero mean and square norm one:

$$\int_{-\infty}^{\infty} \psi(t)\, dt = 0 \tag{6}$$

$$\int_{-\infty}^{\infty} |\psi(t)|^2\, dt = 1 \tag{7}$$

In the case of continuous wavelet transform, the Ψ must be a wavelet. The mother wavelet must satisfy an admissibility criterion to get a stably invertible transform. In the case of discrete wavelet transform, the multi-resolution analysis is used in many constructions, which defines the wavelet by a scaling function.

III. SYSTEM ALGORITHM

Our algorithm contains four main techniques: 1) we uses a skin-tone extraction as a preprocessing step. 2) Then, we use principle component analysis to successfully detect the face using Yale Face Image Database B. 3) After detecting the face edge detection is preformed to detect the edges in the face. 4) The output of the edge is taken as an input to the wavelet transformation.

A. Face Detection Algorithm

1) Initialization

The first step is initialization, by estimating the eigenface from the database to get the mean position and size of the faces. First, we load the name of the training set from a text file. Then, we get the principle component analysis of the eigenfaces resulting from the basic eigenface algorithm. At last by manual processing we identify the mean position and size of the face.

2) Training

We load the training set from the list file. Then, the training set in cropped to meet the criteria from phase 1. Then, we apply the algorithm discussed in [6]. At the end, we reduce the dimension of the output by choosing the principle components.

1. The first step was getting the mean of the training set.

$$\Psi = \frac{1}{M} \sum_{n=1}^{M} \Gamma_n \tag{8}$$

2. Then, we centers the training set around zero by subtracting the mean from it.

$$\Phi_i = \Gamma_i - \Psi \tag{9}$$

3. Afterwards, we calculate the eigenvectors and the eigenvalues of the covariance the matrix.

$$\begin{aligned} C &= \frac{1}{M} \sum_{n=1}^{M} \Phi_n \Phi_n^T \\ &= AA^T \end{aligned} \tag{10}$$

Where $A = [\ \Phi_1\ \Phi_2\ \dots\ \Phi_M\ ]$

4. Then finally, we get the principle components of the eigenfaces depending on its eigenvalues.

3) Image Preprocessing

We insert any color image as an input image. This image is resized to a 256×256 pixel image. Then, the image is normalized to adjust the lightening condition. Afterwards, we apply a skin detection algorithm to eliminate the background from the resized image. At the end, the image is transformed into a grayscale image of PGM format.

4) Face Candidate Projection

The gray scaled image is scanned for collecting samples. These samples are subtracted by the training set mean. And then, the output samples are projected on the reduced dimension matrix from the training phase.

5) Estimation

For the final phase, the projection matrix is sorted in ascending order and we get the smallest projection as our final output of the face and draw a rectangle of fixed size 150×180 pixel.

B. Eye Detection Algorithm

1) Normalization

Due to the undesirable effects of the luminance the YCbCr code gives us undesirable output, so we decided to use a normalization code to normalize the luminance of the input picture.

2) Edge Detection

Edge detection as mentioned was our first approach but as we witnessed it was not the best approach when it comes to eye detection due to luminance and other factors but after all it gave us the edges of the eye so we decided that it will be our input to the wavelet transform, so that it will make the eye socket detection easier.

3) YCbCr transformation

We developed an algorithm that has the ability to identify the human skin color and it turns it to blue based on certain thresholds. Its main idea that it converts the input RGB image to YCbCr space, using for loops and conditional if-statements it could set the skin color into blue (value: 255), and it cancels any objects at the background (value: 0).

4) DWT

The main aim of this function is to compute the second order DWT of an input matrix (image). We did it on two steps, first we calculated the DWT coefficients using 'haar' method, second, we recomputed the second order DWT of the computed coefficients, then we took the absolute value the second coefficients in a matrix (Z).

5) Performing IDWT.

Computing the inverse DWT of the image. We calculated the addition of the selected node from the wavelet tree and its maxima.

6) Combination Between YCbCr & Wavelets.

Given the output of the wavelet we combined the wavelet to YCbCr by adding the two images together.

7) Cropping the eye region

After combining the YCbCr with wavelet transform all possible eye position is found but that was it not enough we still had to chose from the many eye position selected .To do that we cropped 10x10 image of the YCbCr of every selected eye position, then we calculated the percentage of white color in the cropped region and we took the highest 10% of them to lower the number of eye locations

8) Locating the eyes

After getting 10% of the cropped images we then make sure that there is no eye position close to the other (we choose 5 pixels) then we cropped these position 20X20 ,of the eye-position in 512X512 colored image ,to look for white region in this area which will be identified as the eye if not found an alarm will sound.

IV. Experimental Results

As we can see in the below output image, acceptable eye detection is performed despite some facial conditions like status of the eyes, the existence of glasses, different skin colors, and head positions. This algorithm has proven flexibility and reliability. If we use perfect background and perfect face detection the efficiency will increase up to 95%

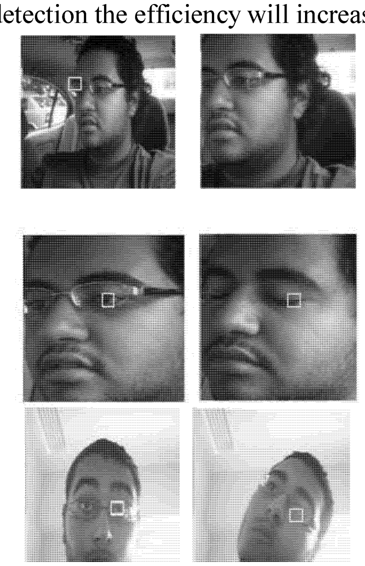

Fig. 2 Successful eye detection

V. Conclusion and Future Work

We have presented an algorithm for Drowsy Driver Assistant System that detect skins as a pre-processing step and then apply PCA rules to detect faces successfully. The algorithm begins with initialization and then training which are applied only once. Then, we take the input image are preprocess it to get a skin only image in grayscale. The image is then sampled and project on the training space. Finally, an estimation phase is applied to detect the face. After getting the face, the algorithm performs edge detection and wavelet transformation to detect the eye position. At last, we zoom into the eye position to know the status of the eye by finding the white components in it.

For future work, the algorithm will be experimented in night condition images.

ACKNOWLEDGMENT

We would like to thank Eng. Mohamed El Mahdy with the Computer Science department, faculty of Media Engineering and Technology, German University in Cairo for all the support and help that he had shown for us. We would also like to thank all the researches who published their papers on the internet for access.

REFERENCES

[1] Addison, P. S., *The Illustrated Wavelet Transform Handbook*, Institute of Physics, 2002.

[2] Bianchini, M., and Sarti, L. "An Eye Detection System Based on Neural Autoassociators". Università degli Studi di Siena, Siena.

[3] Claypoole, R., Lewis, J., Bhashyam, S., and Kelly, K. "Laplacian Edge Detection". *Digital Image Processing at Rice University*, 1997. Available:: http://www.owlnet.rice.edu/~elec539/Projects97/ morphjrks/laplacian.html

[4] Claypoole, R., Lewis, J., Bhashyam, S., and Kelly, K. "Image Processing: Edge Detection". *Digital Image Processing at Rice University*, 1997. Available: http://www.owlnet.rice.edu/~elec539/Projects97/morphjr ks/moredge.html

[5] Dailey, M. "Matt's Matlab Tutorial Source Code Page". University of California, San Diego, 1997: http://www.cs.ait.ac.th/~mdailey/matlab/

[6] Delac K., Grgic M., and Grgic S. "Independent Comparative Study of PCA, ICA, and LDA". *International Journal of Imaging Systems and Technology*, 252-260, 1995.

[7] El-Bakry, H. M., and Zhao, Q. "Sub-Image Detection Using Fast Neural Processors and Image Decomposition". *World Enformatika transactions on engineering, computing and technology*, pp. 85-88, February 2001.

[8] Fröba, B., and Küblbeck, C. "Robust Face Detection at Video Frame Rate Based on Edge Orientation Features". Fraunhofer Institute for Integrated Circuits, Erlangen.

[9] Garcia, C., and Tziritas, G. "Face Detection Using Quantized Skin Color Regions Mergin and Wavelet Packet Analysis". *IEEE Transactions on Multimedia*, pp. 264-277, September 1999.

[10] Hamdy, A. "Drowsy Driver Assistant System: Principal Component Analysis". GUC, Cairo, 2007.

[11] Hampton, C., "DANGER: Drowsy Drivers Can Kill". Available: http://www.paturnpike.com/newsletters/summer96/page2-3.htm

[12] Hsu, R.-L., Abdel-Mottaleb, M., and Jain, A. K. "Face Detection in Color Images". *IEEE Transaction on Pattern Analysis and Machine Intelligence*, 696-706, 2002.

[13] Hyvärinen, A., Karhunen, J., and Oja, E. *Independent Component Analysis.* John Wiley & son, Inc., NY, 2001

[14] Knipling & Wang, "Drowsy Driving and Automobile Crashing", , 1995. Available: http://www.nhtsa.dot.gov/PEOPLE/INJURY/drowsy_driving1/drowsy.html

[15] Lee, Y. "Face Detection using PCA".

[16] Lindeberg, T., "Edge detection and ridge detection with automatic scale selection", *International Journal of Computer Vision*, 30, 2, pp 117--154, 1998.

[17] Ng, C., and Pun, T. "Statistical Approach to a Color-based Face Detection Algorithm", 2002.

[18] Rowley, H. A., Baluja, S., and Kanade, T. "Neural Network-Based Face Detection". *IEEE Transactions on Pattern Analysis and Machine Intelligence*, pp. 23-38, January 1998.

[19] Sandeep, K., and Rajagopalan, A. N. "Human Face Detection in Cluttered Color Images Using Skin and Edge Information". Indian Institute of Technology, Chennai.

[20] Tipping, M. E., and Bishop, C. M. "Probabilistic Principle Component Analysis". *Journal of the Royal Statistical Society*, 611-622, 1999.

[21] Turk, M., and Pentland, A. "Eigenfaces for Recognition". *Journal of Cognitive Neuroscience*, 71-86, 1991.

[22] Wu, B., Ai, H., and Liu, R. "Glasses Detection by Boosting Simple Wavelet Features". Tsinghua University, Beijing.