



Drowsy Driver Detection and Monitoring System Using Support Vector Machine

Haruna Bassi, Hakeem A. Sulaimon, Muhammad Aminu Ahmad
Computer Science Department
Kaduna State University
Nigeria.

ABSTRACT

The current methods for detecting a drowsy driver and taking percussive measure in reducing road accidents are subjective. Many researchers have adopted different methods to address how road accident could be reduced. Existing methods such as measuring brainwave activity using electroencephalography (EEG), electrodes wrapped on a drivers scalp will make the driver uncomfortable during driving, wires crossing all over the driver's body. This paper looks at the use of principal component analysis, local binary pattern and support vector machine to classify the degree of change in the eye lid. These method places a safe and secure way of detecting drowsiness and reducing accidents at minimal cost. To measure these work we loaded a training set and a test set of different faces and feature (the eye) on Mat lab 8.0 .During the evaluation phase we compared linear, quadratic and polynomial kernels of the support vector machine. The true positive, false negative, true negative and false positive values were placed on a confusion matrix to obtain the giving result. The polynomial support vector machine had the highest accuracy of 99.4%

ARTICLE INFO

Article History

Received: May, 2020

Received in revised form: June, 2020

Accepted: July, 2020

Published online: August, 2020

KEYWORDS

Drowsy, monitoring, support vector machine, local binary pattern, and principal component analysis.

INTRODUCTION

Roads are pathways on which people move or travel, either wheeled or on foot. Roads can also be described in rural terms as less travelled way, while street describes roadways in urban terms. Highway refers to express ways travelled through major rural areas; it's also controlled points of entrance and exit from one city to another.

Accidents expressed by Odugbemi, (2010), a road traffic accident can be defined as that, which happens by chance, occurring unexpectedly and undersigned. According to the National Highway Traffic Safety Administration, driver fatigue was

responsible for 72,000 accidents that lead to more than 800 deaths in 2015. By 2020, it is estimated that traffic accidents will be the fifth leading cause of death in the world, reaching 2.4 million deaths per year (WHO, 2013).

The possible techniques for detecting drowsiness in drivers can be broadly divided into three major categories.

1. Based on driver's current state.
2. Driver's performance.
3. A Combination of the current state and driver's performance.



DROWSINESS DETECTION, FACIAL RECOGNITION, FEATURE EXTRACTION (OPENED/CLOSED EYE)

Face recognition is essentially the task of recognizing someone supported by his or her facial features which involves extracting, cropping, resizing and frequently converting pictures to grey scale, Alioua et al., (2016), carried out drowsiness measure using head pose, Yawning, and the Geometric method, Steerable filters, Histogram of oriented gradient (HOG), they used Haar features and support vector machine for Classification. Mandal et al., (2017), also used per mint eye closure (PERCLOS) as drowsiness detection measure; at the detection phase they used Harr features, Adaboost, Histogram of oriented gradient (HOG), Maximum Likelihood algorithm Spectral Regression; Threshold based and support vector machines as classifier. D'Orazio et al., (2007), measured Eye closure duration, Frequency of eye closure, Hough Transform for detection technique and used artificial neural network as Classifier. Mbouna et al., (2013), classified Pupil and Head pose estimation using Support Vector Machine as classifier, the lucas-kande optical flow and geometric constraints Viola-Jones Algorithm was used for detection and Feature Extraction. Danisman et al., (2010), discussed Drowsiness measure and Blink duration, using Viola-Jones algorithm, Neural network based eye detector, and Classifying drowsiness using Threshold based. Xie et al., (2012), also discussed Drowsiness measure base on PERCLOS, Head pose, and they used Viola-Jones

Algorithm in detection and feature extraction. Saradadevi et al., (2008), measured Yawning, using support vector machine as classifier and Viola-Jones Algorithm as Detection techniques. Sacco et al., (2012), carried out Drowsiness measure like PERCLOS, AECS, Mouth opening, and Viola-Jones Algorithm as detection technique and Correlation coefficient for template matching. The PERCLOS is found by the quantitative relation in overlapped time windows of one (1) minutes and a notification is given once it's over a threshold

Objectives

The objective of this study is detecting and monitoring drowsiness during driving situations using principal component analysis, local binary pattern and support vector machine in monitoring changes that occurs within the eye region that is the eye lids and contours, based on visual information obtained from camera assumed mounted on dashboards.

METHODOLOGY

System Flow chart Diagram

The flow diagram of the system is depicted in Fig 1 shows schema of image detection, capturing, to illustrate that a support vector machine with its function, algorithm and using local binary function and principal component analysis can effectively be used to predict a drowsy driver using images. In this study the input values (pictures frames) would be considered for prediction of drowsiness.

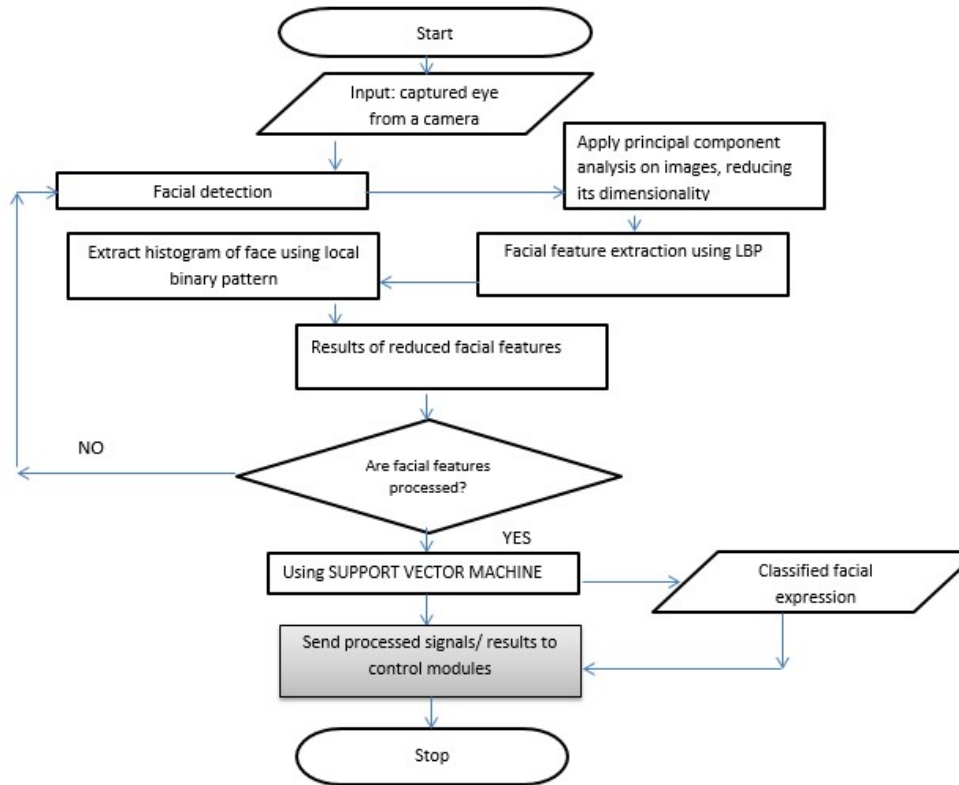


Figure 1: Flow chart classifying drowsy eyes using support vector machine.

Local Binary Pattern

The local binary pattern Local Binary Patterns (LBP) in other term viewed and termed with the function of describing image texture be a texture descriptor which could be also used to represent individual faces; an individual face image comprises of micro-texture-patterns used to find and classify an image region and the variation of its native intensity. The local binary pattern (LBP) operator is carried out in form of a gray-scale invariant texture, which is gotten from the native neighborhood of the image texture. The local binary pattern operator also assigns values to the components features of an image by thresholding the neighborhood of every pixel and the results obtained are known as a binary variety. The

unique feature of LBP is that it does not change the illumination of an image (Ojala et al., 2002).

Texture Descriptor

Traditionally Local Binary Pattern has been viewed to be a generic texture descriptor. Here every pixel of the images is assigned a value by the operator by thresholding a 3x3 neighborhood with the middle pixel value which is either a number zero or one these resulting values are read either clockwise. Starting from the top left neighbor which is a 1 and moving towards the upper right corner 0 in circular movements so that we have values, such as 10001101 as can be seen in the following figure below.

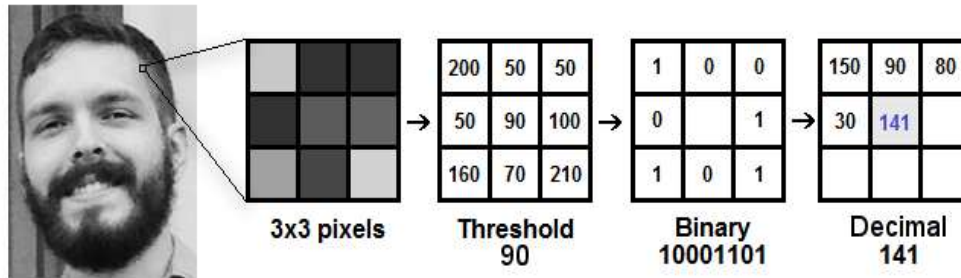


Figure 2 Binary block obtained from Salton (2017).

Fig 2 above gives an Example of texture primitives Calculation of LBP feature. To find the local binary pattern of a giving image we will carry out the following:

1. Divide a picture window in to cells where each cell make up a pixel say a 16x16 pixel for each cell; compare the element to its eight neighbours upper left, left-middle, left-bottom, right-top and lower right. Follow the pixels in a circular pattern, in a clockwise or anti-clock-wise pattern.
2. During computation the value of the centre pixel may be greater than the value of its neighbours we will write 1 else write 0 this give an 8 digits binary variety; this sometimes could be rewritten in to decimals form for easy computation.
3. From the result obtained above we can find the frequency of each number in terms of zeros and one with that we will create a bar chat where every combination of that pixels square measures smaller and that square measure larger

than the centre.. From the result obtained above we can find the frequency of each number in terms of zeros and one with that we will create a bar chat where every combination of that pixels square measures smaller and that square measure larger than the centre..

4. Grey scale image, LBP image is shown in Fig 2 above, the grey scale and corresponding LBP image Fig 4 Block-LBP bar chart is displayed below
5. Below is a sample of local histograms and picture element level values to urge the feature vector using matlab tool.

If there is a texture T in a local neighborhood of a grey scale image as the joint distribution of gray levels of $P(P>1)$ image pixels. Circularly bilaterally symmetric neighbor sets for various (P, R) For grey scale un-changeability the grey worth of the center constituent is subtracted from the circularly symmetric neighborhoods $g_p (P= 1,2, \dots p-1)$

$$\text{Giving } T = (g_0 - g_c, g_1 - g_c, \dots, g_{p-1} - g_c) \dots \dots \dots (3.3)$$

Now we assume the differences $g_p - g_c$ are independent,

$$\text{so } T = t(g_c) t(g_0 - g_c, g_1 - g_c, \dots, g_{p-1} - g_c) \dots \dots \dots (3.4)$$

Where a gray value g_c corresponds to the gray value of the center pixel of the local neighborhood and $g_p (P= 1, 2 \dots p-1)$ correspond to the gray value of P equally spaced pixels of a circle of radius ($R > 0$) that form a circularly symmetric neighbor set.

If the coordinates of g_c is $(0, 0)$ then the coordinates of g_p are given by $(-R \sin 2\pi P/p, R \cos 2\pi P) \dots \dots \dots (3.2)$

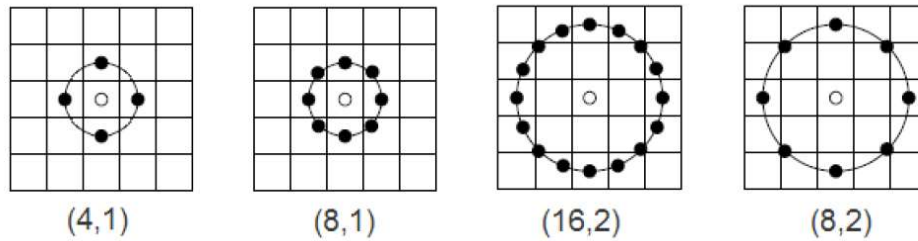


Figure 3: Circularly symmetric neighbor sets for different (P, R) , Sánchez (2010)

The Figure 2 shows circularly Centro symmetric neighbor sets for numerous (P, R) . The grey values of neighbors that don't fall precisely within the center of pixels are calculable by

interpolation. The occurrences of patterns within the neighborhood of every picture element are going to be encoded in an exceedingly P -dimensional bar chart.

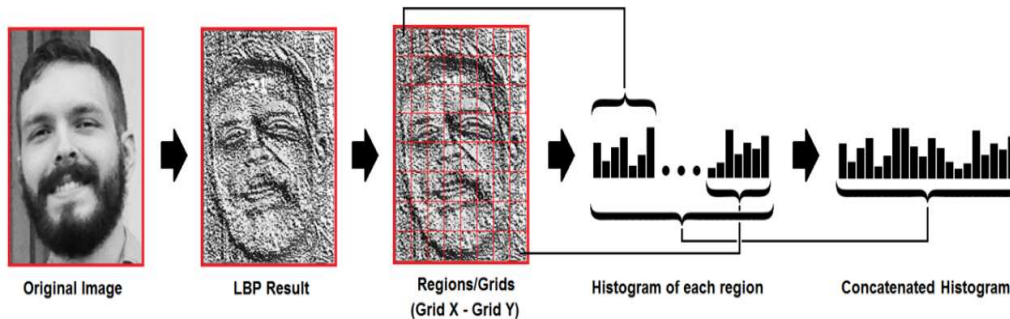


Figure 4: Histogram obtained from Salton (2017).

Training Algorithm

A full system using Local Binary Patterns and the principal component Analysis for face detection and recognition of face features stages will be detailed and evaluated as follows

- i. Open eyes of size 50×40 are designated and every image is divided in to sub-blocks of size 5×4 a pair of. The LBP feature values of the sub-blocks are found.
- ii. Sixteen bin bar chart of every block is calculated.

- iii. The bar chart of every block is organized to make a worldwide feature descriptor.
- iv. Principal Component analysis is finished on the feature vectors obtaining the Eigen vectors with highest Eigen values and designating them.

Detection Phase

Detection and recognition stages has been detailed, evaluated as follows.

- i. Region of Interest from face detection stage is obtained.

- ii. For every sub window the Block-LBP bar chart is found and is projected to Eigen area.
- iii. The sub window with minimum reconstruction error is found
- iv. If the reconstruction error is a smaller amount than a threshold it's thought-about as a positive detection.

Support Vector Machine

The classification between open and closed eye is completed and supported

using geometrician distance that is the Euclidean distance. The intuition of comparing one data point to the other data point and try to find how close is that certain point says '+' or '-', is to the k's closest point and whatever the majority of that k- closest point is then the new point is that class. We will break down the algorithm neighbors; this algorithm hinges on a distance called the Euclidean distance. To accomplish our task, on drowsiness detection we used the distance Euclidean as giving by equation below.

$$d = \sqrt{\sum_{i=1}^v (P_{1i} - P_{2i})^2}$$

Vega et al., (2017) maintains that support vector machine takes input as set of n feature vectors $x_i \rightarrow$ together with their labels $y_i \in \{1, -1\}$. The fact as to support Recall the distance from a point (x_0, y_0) to a line: $Ax+By+c = 0$ is: $|Ax_0 + By_0 + c|/\text{sqrt}(A^2+B^2)$, so,

The distance between $wx+b=1$ and $wx'+b = 0$ is then: $|w \cdot x + b|/||w||=1/||w||$, so The total distance between $wx+b =1$ and $wx' + b = -1$ is thus: $2/||w||$. In order to maximize the margin, we thus need to minimize $||w||$.

vector machines is to find the hyper plane that maximizes the distance between two classes $\{1, -1\}$. Finding a solution to the optimization problem does this

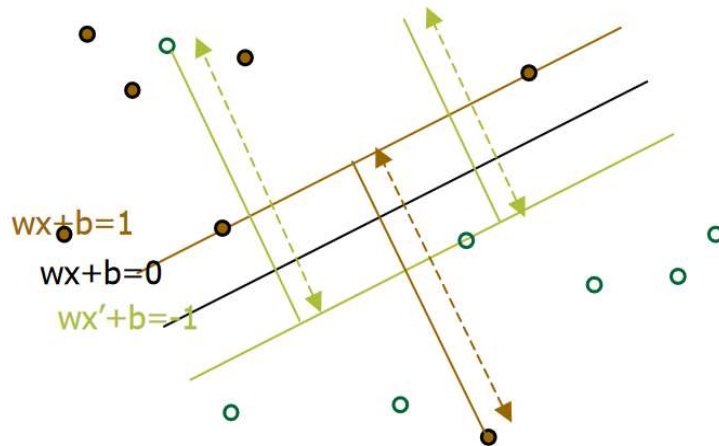


Figure 5: Support Vector Machine hyper planes. Jakkula (2009)

$$x_i \cdot w + b \geq +1 \text{ where } y_i = +1 \dots\dots\dots (1)$$

$$x_i \cdot w + b \leq -1 \text{ where } y_i = -1 \dots\dots\dots (2)$$

(1) And (2) can be combined into: $y_i(x_i \cdot w) \geq 1$

Minimize $\|w\|$, standard discrimination boundary is obeyed, i.e.,
 $\min f(x), g(x)=0$

Which can be rewritten as: $\min f: \frac{1}{2} \|w\|^2 \quad g: y_i (w \cdot x_i) - b = 1 \text{ or } [y_i (w \cdot x_i) - b] = -1.$

SIMULATIONS, RESULT, AND ANALYSIS

This section provides the system simulation with a snapshot of data for extracting information on Mat lab and sum-link of our system; also, we have data collected from IntelliEye dataset and had an accuracy of 99.4% using polynomial Support Vector Machine.

Data Set

The dataset was obtained from www.intellifaces.com, collected data has 37 different persons (33 men and 4 women) the dataset consists of 84,898 images, dataset contains the information about gender for each image (man, woman), the information if the eye image contains glasses is also provided for each image (with and without the glasses) also information about two eye states (open, close). At this moment, the dataset contains the images captured by three different sensors (Intel Real Sense RS 300 sensor with

640 x 480 resolutions, IDS Imaging sensor with 1280 x 1024 resolution, and Aptina sensor with 752 x 480 resolutions). In other to achieve our set objectives we carried out training of the Support Vector Machine using 460 images and during the testing stage 1700 samples were tested. The training images are extracted using MATLAB tool 8.0.

Then Principal component Analysis (PCA) is assigned on each feature vectors. Binary Pattern bar graph feature was completed using different kernels as shown below we obtained different accuracy level using the linear, quadratic and polynomial Support Vector Machine. The vector space in line with the weights and bottom truth are used for Support Vector Machine. The carried out training is completed with completely different kernels and different accuracy levels were obtained. The results are shown below.



Figure 6: Images obtained from Intellifaces (Eye), dataset of opened and closed eyes

Corresponding author: Bassi, H. ✉ bassi.haruna1@gmail.com ✉ Computer Science Department, Kaduna State University, Kaduna. © 2019 Faculty of Technology Education, ATBU Bauchi. All rights reserved

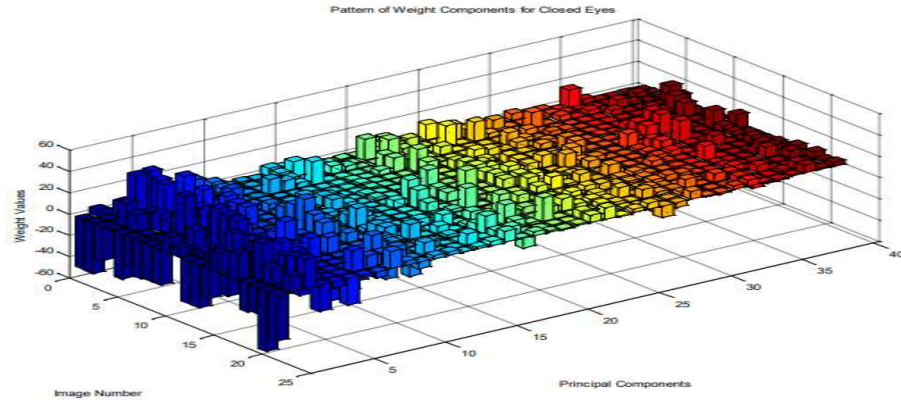


Figure 7: Pattern weights of closed eyes on a mat-lab (3-d)

From the below table, the rate of true positive and false negative was calculated using the formula.

Table 1: Actual /predictive values obtained are classed into the table below

		Actual Values	
		Positive (1)	Negative (0)
Predicted Values	Positive (1)	TP	FP
	Negative (0)	FN	TN

The true positive rate (TPR), (sensitivity) is giving as --- $TPR = \frac{TP}{TP+FN}$

The True negative rate (TNR) is giving as ----- $TNR = \frac{TN}{TN+FP}$

The false positive rate (FPR) is giving as ----- $FPR = \frac{FP}{TN+FP}$

Accuracy = $\frac{TP + TN}{TP + TN + FN + FP}$

Table 2. Shows the result of this work on a confusion matrix

Kernel function	True positive	True negative	False positive	False negative	True positive rate	False positive rate
Linear/straight line Support Vector Machine	738	708	44	14	98.10%	5.8%
Quadratic Support Vector Machine	727	665	87	25	96.6%	11.5%
Polynomial Support Vector Machine	748	707	45	4	99.4%	5.9%



CONCLUSION

Based on our findings, polynomial SVM has the highest accuracy of about 99.4% it is clear that through the process of principal component analysis and local binary function algorithm we had effectively predicted drowsiness using support vector machine (polynomial) on mat-lab. Further works it is recommended that other machine learning tool and models to predict drowsiness should be employed.

REFERENCE

- Alioua N., Amine A., A. Rogozan, A. Benshair, and M. Rziza, (2016), "Driver head pose estimation using efficient descriptor fusion," *Eurasip J. Image Video Process*
- D'Orazio T., Leo, Guaragnella C., and Distanti A., (2007), "A visual approach for driver inattention detection," *Pattern Recognit.*, vol. 40, no. 8, pp. 2341–2355.
- Danisman T., Bilasco I. M., Djeraba C., and N. Ihaddadene, (2010), "Drowsy driver detection system using eye blink patterns," *2010 Int. Conf. Mach. Web Intell. ICMWI 2010 - Proc.*, pp. 230–233,
- Laura Sánchez López (2010) *Local Binary Patterns applied to Face Detection and Recognition Directors*: Francisc Tarrés Ruiz, Antonio Rama Calvo. *Signal Theory & Communication Department*.
- Mandal B, Li L., Wang G. S., and Lin J., (2017), "Towards Detection of Bus Driver Fatigue Based on Robust Visual Analysis of Eye State," *IEEE Trans. Intell. Transp. Syst.*, vol. 18, no. 3, pp. 545–557.
- Mbouna R. O., Kong S. G., and Chun M. G., (2013), "Visual analysis of eye state and head pose for driver alertness monitoring," *IEEE Trans. Intell. Transp. Syst.*, vol. 14, no. 3, pp. 1462–1469,
- Sacco M. and Farrugia R., (2012), "Driver fatigue monitoring system using support vector machines," *5th Int. Symp. Commun. Control Signal Process.*
- Saradadevi M. (2008), "Driver Fatigue Detection Using Mouth and Yawning Analysis," *Int. J. Comput. Sci. Netw. Security*, vol. 8, no. 6, pp. 183–188,
- Xie J. F., Xie M., and Zhu W, (2012), "Driver fatigue detection based on head gesture and PERCLOS," *Int. Conf. Wavelet Act. Media Technol. Inf. Process. ICWAMTIP 2012*, pp. 128–131, 2012.
- Vega R., Sajed T., Mathewson K. W., Khare K., Pilarski P. M., Greiner R., (2017). Assessment of feature selection and classification methods for recognizing motor imagery tasks from electroencephalographic signals. *Artif. Intell. Res.* 6:37 10.5430/air.v6n1p37 [CrossRef] [Google Scholar]
- Vikramaditya Jakkula (2009), *Tutorial on Support Vector Machine (SUPPORT VECTOR MACHINE)* School of EECS, Washington State University, Pullman 99164. www.intellifaces.com