Iran University of Science and Technology

International Journal of Automotive Engineering

Journal Homepage: ijae.iust.ac.ir

ISSN: 2008-9899

Driver Drowsiness Detection by Identification of Yawning and Eye Closure

Mina Zohoorian Yazdi[1], Mohsen Soryani [2*]

[1] School of Computer Engineering, Iran University of Science & Technology
E-mail: Mina.zjy93@gmail.com
[2]School of Computer Engineering, Iran University of Science & Technology
E-mail: soryani@iust.ac.ir

ARTICLE INFO

Article history
Received: 25 May 2019
Accepted: 28 Aug 2019
Published: 1 Sept , 2019

Keywords:
Active Contour
Driver Drowsiness
Deep Learning
Depth Information
Eyes State
RGB_D
Yawning Detection

A B S T R A C T

Today most accidents are caused by drivers' fatigue, drowsiness and losing attention on the road ahead. In this paper, a system is introduced, using RGB-D cameras to automatically identify drowsiness and give warning. In this system two important modules have been utilized simultaneously to identify the state of driver's mouth and eyes for detecting drowsiness. At first, using the depth information, the mouth area and its state are identified. Then using CNN networks, to predict whether the eyes are open or closed, a semi-VGG architecture is used .The results of yawning and eyes states detection are integrated to decide whether an alarm should be issued. The results show an accuracy of about 90% which is encouraging.

1. Introduction

Driving is a common activity in most people's lives and securing it is an important matter in daily life. Though safety on roads and car design are improving, the total number of accidents is rising at the same time [1]. Recently, new methods, specially based on artificial intelligence, have been used to improve car's safety. In [2] an adaptive modified fuzzy-sliding mode controller has been designed for vehicles equipped with ABS system. In [3] a new version of multi-objective differential evolution with dynamically adaptable mutation factor has been used for Pareto optimization of a 5-degree of freedom vehicle vibration model. In [4] obstacle avoidance for autonomous vehicles using force field method has been investigated. As people get older, they have slower reactions and responses to sudden events. Also drivers' distractions are increasing due to the added facilities now available in cars such as mobile phones and the internet. Among the main factors which result in driver's loss of attention to the road ahead, fatigue and drowsiness have been reported in most accidents [5]. The evaluation of drivers' attention loss has become a very active area of study in smart transportation systems. In [6], an efficient module for identifying the distraction of drivers has been made based on active Kinect sensor and computer vision tools. Using the color and depth data of Kinect camera, the system consists of 4 modules: 1-considering the eye behavior (identifying gazing or blinking) 2-determining hands position 3-detetmining head position 4-identifying the face state. Each module produces information for evaluating drivers' distraction and finally all the information is

* Mohsen Soryani
Email Address: soryani@iust.ac.ir
DOI: 10.22068/ijae.9.3.3033

[DOI: 10.22068/ijae.9.3.3033]

integrated using two different classification strategies (Adaboost, Hidden Markov Model). In [7] color and depth outputs of the Kinect sensor are used in the following stages: 1-locating head and detecting face, 2-extracting face and head features, 3-locating eyes 4- gaze estimation 5-identifying blinks 6-measuring head movements. Head movements around 3 axes are considered to determine roll, yaw and pitch angles in 3-D coordinates system. Face features including locations of eyebrow, mouth, nose and other components are calculated. Pupils and their angles are used to estimate gaze direction and eyes patterns are used to identify blinking. [8] Presents a real-time system to identify fatigue and drowsiness of the driver using RGB-D cameras. The introduced system consists of two important parts: estimating head position and identifying eyes states. Identifying the head position is performed using optical flow and depth constraint. Then a new feature descriptor, WLBP, is used to identify eyes states. Experiments have shown that WLBP feature is more resistant and more efficient than WLD and LBP features in presence of noise and changes in illumination. It has been shown that the combination of head position and eyes states can improve the effectiveness and robustness of the system. [9] Has presented an algorithm for identifying gaze direction based on learned features using a convolutional neural network (CNN). A 3-layer CNN for extracting features of the identified face area has been used. The output of the last layer is input to a trained classifier to identify drowsiness.

In this paper an algorithm is presented, using Kinect camera's RGB-D data, to monitor driver's fatigue and drowsiness by identifying the eyes state and yawning. A warning will be issued if the driver is found to be drowsy. The rest of this paper is structured as follows: in Section 2 the proposed approach is explained. The results are presented in Section 3 and finally Section 4 concludes the paper.

2. Approach

Fig.1 shows different stages of the suggested algorithm. Identifying drowsiness is done through integrating yawning and eyes closure data (logical OR). Yawning detection involves various stages for analyzing the condition of the mouth. These stages are as follows and each one will be introduced in the next sections:

1-nose tip detection

2-estimation of the mouth area and detecting its openness or closure

3-yawning detection

2.1. Nose tip detection

When head rotation is limited, the nose tip pixel has the minimum depth value in the depth map. When the head rotation exceeds a specified interval, for example, the rotation angle of the head is greater than +20 or less than -20 degrees, glasses or cheeks will have the lowest depth. When the pitch angle is greater than +15 or less than -15 degrees the chin or forehead has the smallest depth.

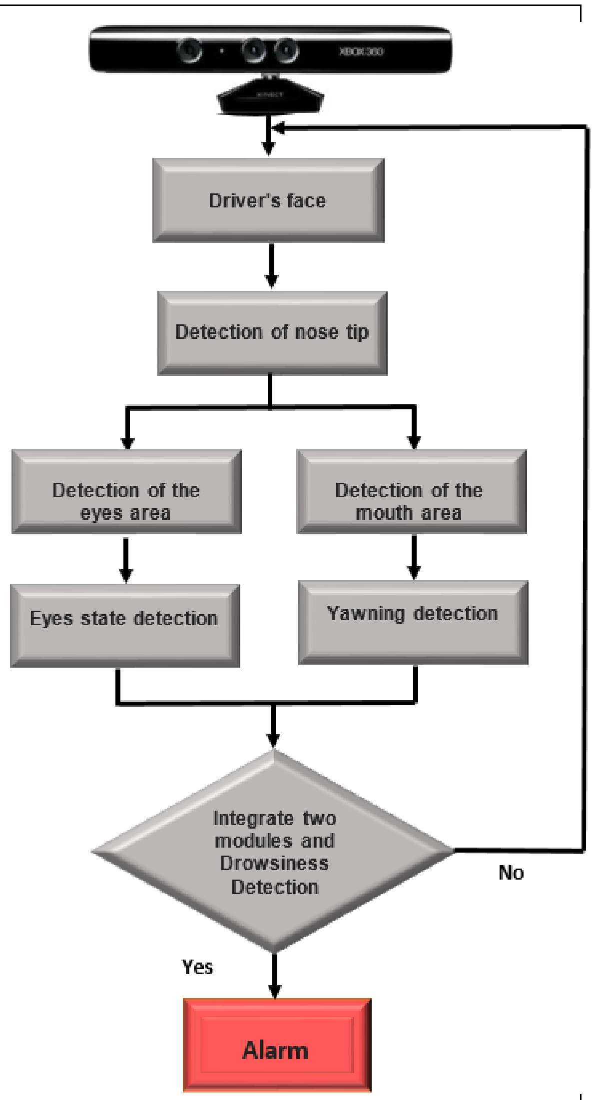

Figure 1: Block diagram of the driver's fatigue and drowsiness detection.

A method for detecting nose tip has 2 steps; using the "out of range" filter to remove invalid depth data and determining the location of pixel with the lowest depth. This method can accurately detect the nose tip but due to small changes that occur in depth information because of noise, at times, the minimum depth value may be mistaken as the glasses or hair. To eliminate this unwanted change, a Border Mask with dimensions chosen according

[DOI: 10.22068/ijae.9.3.3033]

to the number of rows and columns of the depth image is multiplied by the image to remove the hair area around the face (Fig.2). Then a median filter (7 * 7) is applied to the depth image. Now, in the new image, the minimum depth value is obtained and is used as a threshold and a binary image is created. To ensure that the minimum value of the original depth image is not discarded after convolving the median filter, the surroundings of the minimum depth is also considered. A squared structural element with the same size as the median filter is used to dilate the binary image that was created. The result of the expanded binary image is multiplied by the initial depth image element by element so that a wider search area is obtained. Now, the minimum depth pixel is found on this binary image. If more than one minimum point is found, the average of the column values is obtained and the maximum row value is considered. Now, as shown in Fig.3, the bottom half of the face image will be used to estimate the location and state of the mouth.

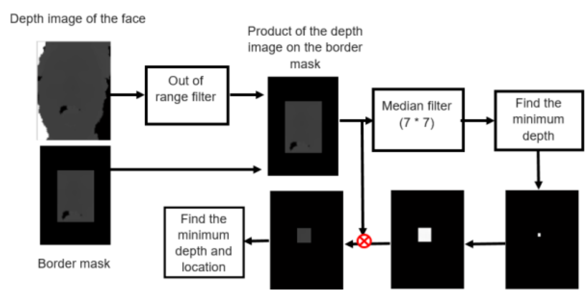

Figure2: Block diagram of the nose tip detection.

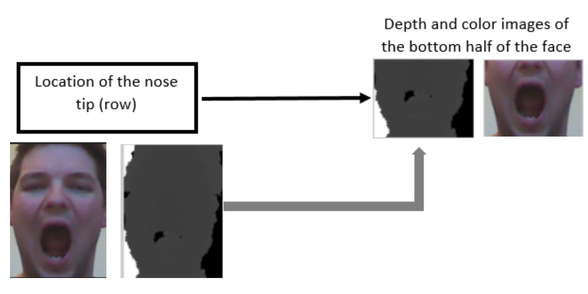

Figure3: Block diagram for separating the bottom half of the face.

2.2. Mouth area detection

In order to identify the location of lips and mouth in the lower face area, an adaptive mask as shown in Fig.4 is applied on the depth image. The depth value of Pixel (i, j) is compared with the average depth of its local neighborhood. The neighborhood size is measured by relation (l):

$$S = (i,j) = \frac{(f * \bar{s})}{d(i,j)} \tag{1}$$

Where f is the focal length of the camera, d (i,j) is the depth value in millimeter for pixel (i,j) and $\bar{S}$ is the average height of the mouth and is considered as 50 mm.

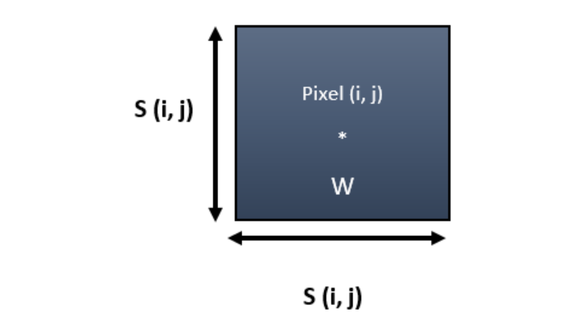

Figure4: Adaptive mask used to detect the mouth area using depth information.

If the absolute difference between the average depth of the window W and depth d(i,j) is higher than a threshold, pixel(i,j) is set to 1, otherwise it is set to 0. Convolving the mask with the depth image acts as an edge detector and can identify edges of the mouth. The average depth of the window W is calculated as:

$$\mu_W(i,j) = \frac{\sum_{(u,v)\in W} V(u,v) * d(u,v)}{\sum_{(u,v)\in W} V(u,v) + \varepsilon} \tag{2}$$

Here $V(u,v)$ is a masking parameter in order to consider only valid depth values and ε is a small constant. If the depth value of a pixel(u,v) is in the range of valid depth values, then $V(u,v)$ is set to one, otherwise it is considered zero:

$$V(i,j) = \begin{cases} 1 & if \;\; d(i,j) \; is \; valid \\ 0 & Otherwise \end{cases} \tag{3}$$

Now by comparing d(i,j) (depth of the window center) with the average window depth, a mask(m) is created:

$$m(i,j) = \begin{cases} 1 & if \; |\mu_W(i,j) - d(i,j)| > \tau \\ 0 & Otherwise \end{cases} \tag{4}$$

Where $\tau = 5mm$ was determined empirically. Figures 5 and 6 show examples of the mouth area detection using this method for both open and closed mouths.

[DOI: 10.22068/ijae.9.3.3033]

Figure5: Results of detecting open mouths (second row) by applying the adaptive mask on the depth image (first row) for 4 different individuals.

Figure6: Results of detecting closed mouths (second row) by applying the adaptive mask on the depth image (first row) for 4 different individuals.

2.3. Open mouth detection

To determine if the driver is yawning, the mouth cavity is analyzed in two ways. In the first method, the maximum depth of the mouth area will be found. An experimental value, according to the amount of data in the depth of the mouth area, is reduced from the maximum depth to produce a threshold. The depth values which are greater than the threshold are set to 1 and all other values will be set to 0. The result is a binary image as shown in Fig.7. Brighter values of depth data which are larger than the threshold demonstrate the mouth cavity. An experimental threshold for this method was chosen 20 (depth of 2 centimeters).

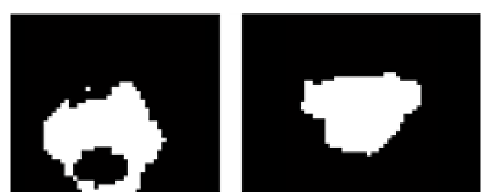

Figure7: Detection of the mouth cavity using depth information for 2 different individuals

The second method used to determine the mouth cavity, is applying the active contour model to the mouth area extracted from the depth image. In this way, the initialization of the contour must be done automatically. The pixel with the maximum depth in the mouth area is considered as the center of a 7*7 window.

[DOI: 10.22068/ijae.9.3.3033]

This window will act as the initial active contour and in the following iterations it will expand until it adapts to the boundaries of the curvature of the driver's mouth. Fig.8 shows results of using active contour model for 2 individuals.

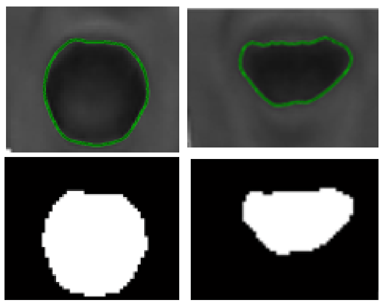

Figure8: Application of the active contour model on the mouth area for 2 individuals.

To find the yawning state, intersections of the binary images obtained from the two previous methods will be determined. The first and the last row of this binary image are found and if the distance between two rows is greater than a threshold, the yawning state is declared, otherwise the mouth is considered closed or in a talking state. In Fig.9 the distance between the first and the last row of the mouth cavity is greater than the threshold and yawning mode is detected.

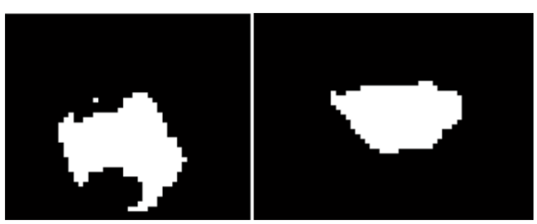

Figure9: Combining the results of two methods to obtain mouth cavity and mouth state for 2 individuals.

2.4. Eyes state detection

There are two default eyes states: open and closed. The eyes are in the open state when the iris and pupil are visible. Classifying an open or closed eye is complicated due to many factors such as the change in shape of the eyes, position and rotation of the face, blinking and illumination. All these factors make the detection of eyes' state difficult. In existing methods, identifying open or closed eyes are done through two stages: 1-extracting the distinguishing features of the eyes, 2-calculating distance criteria between extracted features and open and closed reference eyes. In traditional methods efforts are made to improve suitable features which are extracted manually. Deep learning and convolutional neural networks (CNN) has recently been the center of attention by researchers because of their high strength in automatic extraction of features. Extracting features and measures of learning in deep networks are two fully resolved issues in a way that feature extraction is done by CNN and then features' comparison is performed by classifiers. Fig.10 shows a simple model of extracting features by CNN followed by open/closed eyes classification.

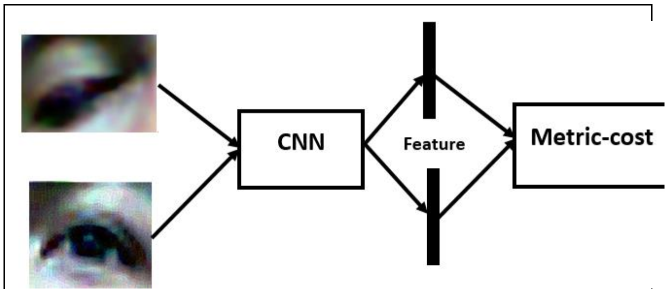

Figure10: A simple model for feature extraction by CNN followed by open/closed eyes classification.

In order to extract eye features from the sequence of input images a semi-VGG model is used. In this model, the last 2 fully connected layers and the softmax layer have been trained for classification of open and closed eyes. The architecture and parameters of the suggested model have been summarized in Fig.11.

[DOI: 10.22068/ijae.9.3.3033]

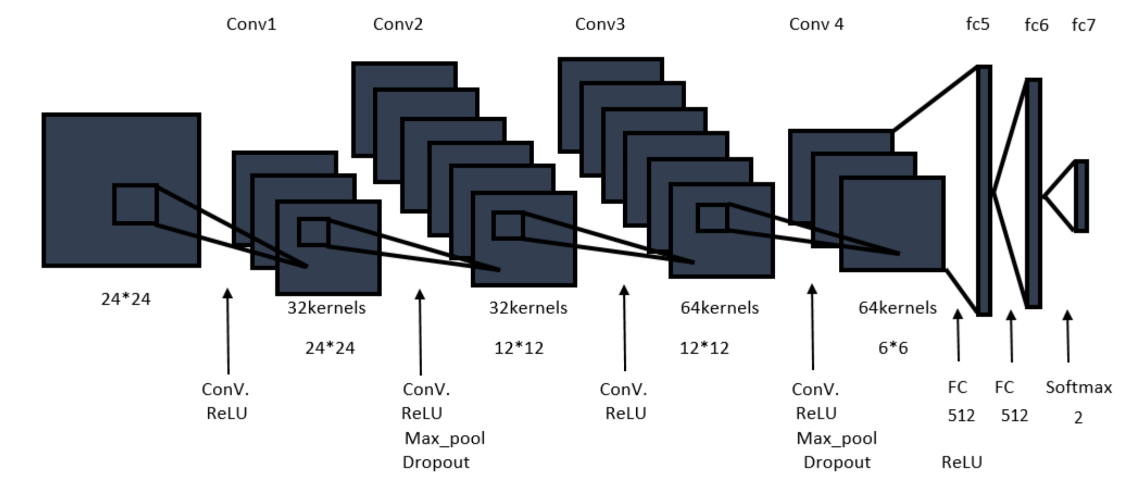

Figure11: Layout and specifications of the semi-VGG network architecture

Each input image is 24×24 pixels. The first convolutional layer with 32 cores of size 24×24 filters the input image. After normalization of the output of the first convolutional layer, it is given as input to the second convolutional layer with 32 cores of size 12×12. The third convolutional layer has 64 cores of size 12×12. The last convolutional layer has 64 cores of size 6×6. The fully-connected layers have 512 neurons. The output of the last fully-connected layer is directed to a two-dimensional Softmax layer which produces labels for open and closed eyes classes.

2.5. Detection of driver's fatigue and drowsiness

In the previous sections, methods for identifying yawn and closed or open eyes were discussed. Identifying driver's fatigue and drowsiness in the suggested work is done by considering states of mouth and eyes simultaneously. The action of yawning and/or eye closure, when feeling drowsy, continues in several frames. By investigating the data set it was found that when a driver yawns, his mouth is open for at least 38 frames. The average number of frames for yawning was found to be 85. We chose 45 frames as the threshold value for identifying driver's yawn. Therefore, looking at the input images, if the mouth is found open in more than 45 successive frames, the system identifies yawning. Another sign of drowsiness, considered here, is closure of eyes in consecutive frames. Human eye blinks about 10 to 15 times per minute. The average duration of blinking is 150 to 250 ms. If eye closure takes long, that is, the interval between eyes opening and closing last longer than a threshold value, driver drowsiness can be confirmed. If we consider the closed eye threshold as 0.5 second, supposing the imaging rate of 30 frames per second, the eyes should be closed in at least 15 successive frames so that drowsiness is authenticated. Detection of yawning and/or eye closure will cause the system to warn the driver for his lack of attention to the road. This is shown in Fig.12. However, in some conditions, identifying the state of eyes and mouth may fail when the driver turns his head or wears glasses or has covered his mouth with his hand. These situations have not been addressed in this paper. To compensate for false negative errors in the eye closure detection stage of the system, when input frames are investigated for finding at least 15 successive frames with closed eyes, small number of detected open eye states are ignored. That is, if at most in 20% of the 15 frames (i.e 3frames) open eyes were detected, they are ignored and a closed eyes incident is declared. We use the same error compensation for the yawning sequences.

[DOI: 10.22068/ijae.9.3.3033]

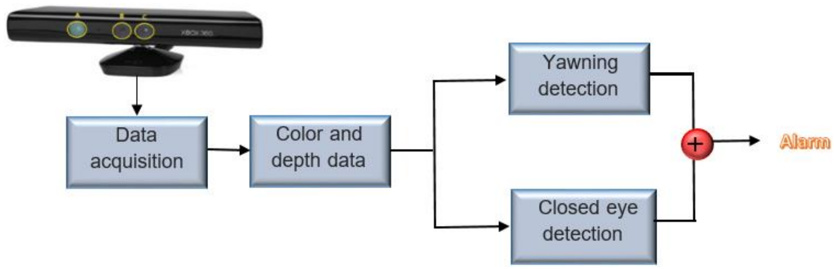

Figure12: Block diagram of the driver fatigue and drowsiness detection system.

3. Experimental results

The suggested algorithms in this paper have been evaluated on the VAP_RGBD dataset [20]. The dataset contains faces of 31 individuals, each individual has been pictured in 17 different positions, and each state has been repeated 3 times. As a result, there are 3 * 17 * 31 = 1581 RGB images and 1581 depth images in the data set. Color images have 32 bit standard bitmap format with resolution of 1280 x 960 pixels. Depth images are text files where each pixel of the depth image is represented by its depth value. The value of depth is measured using a millimeter-scale distance by the Kinect camera. Depth images have a resolution of 480 x 640 pixels where valid values for depths range from 400 to 4000 mm. Some outlaws in depth images are: undefined depth values -1, depth values that are too close 0 and depth values that are too far 4095.

Another dataset prepared by authors, using an Xbox One 360 Kinect camera, includes 10 sequences of images from different people with 10000 frames. People were located in the distance of 50 to 100 cm from the camera. Videos were recorded with 30 frames per second. RGB images were saved using Bitmap format and depth images were saved as text files with 480*640 resolution. Valid values for depth range from 50 to 3000 mm.

3.1. Nose tip detection

As previously stated, the yawning detection algorithm is run in 3 separate stages; nose tip detection, estimation of the mouth area and yawning detection. As noted in Section 2, the nose tip is determined by finding the lowest depth value in the depth image. The use of depth information eliminates the algorithm's dependency on illumination variations and reduces errors in the following stages. Figs.13 and 14, show the results of nose tip detection and finding the bottom half of the face respectively, for four different individuals.

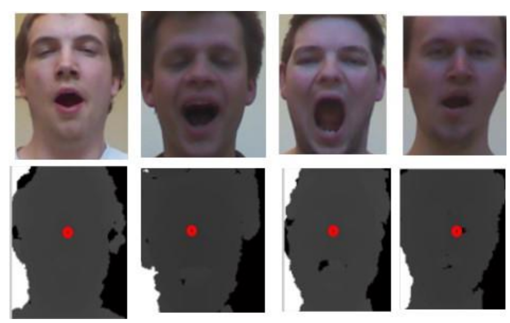

Figure13: The minimum depth location in the face depth image.

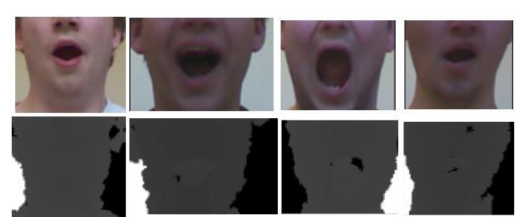

Figure14: Using the nose tip to separate the bottom half of the face.

3.2. Mouth area detection

[DOI: 10.22068/ijae.9.3.3033]

Using the mouth detection algorithm explained in section 2, experiments were conducted on 140 images of open mouths and 40 images containing semi-open or closed mouths. The algorithm successfully identified 136 out of the 140 cases of open mouths and 31 out of the 40 cases in which the mouth is semi-open or closed. Figs.15 and 16 and Table.1 represent the results obtained from these experiments. Results of the Viola Jones algorithm [21] are also included for comparison. The proposed mouth area detection algorithm detected 167 out of 180 cases correctly with an accuracy of 91% while Viola Jones ' algorithm has only detected 106 cases correctly with 58% accuracy. Viola Jones ' algorithm has high accuracy in detecting closed mouths. The superiority of the proposed algorithm in detecting both closed and open mouths is evident.

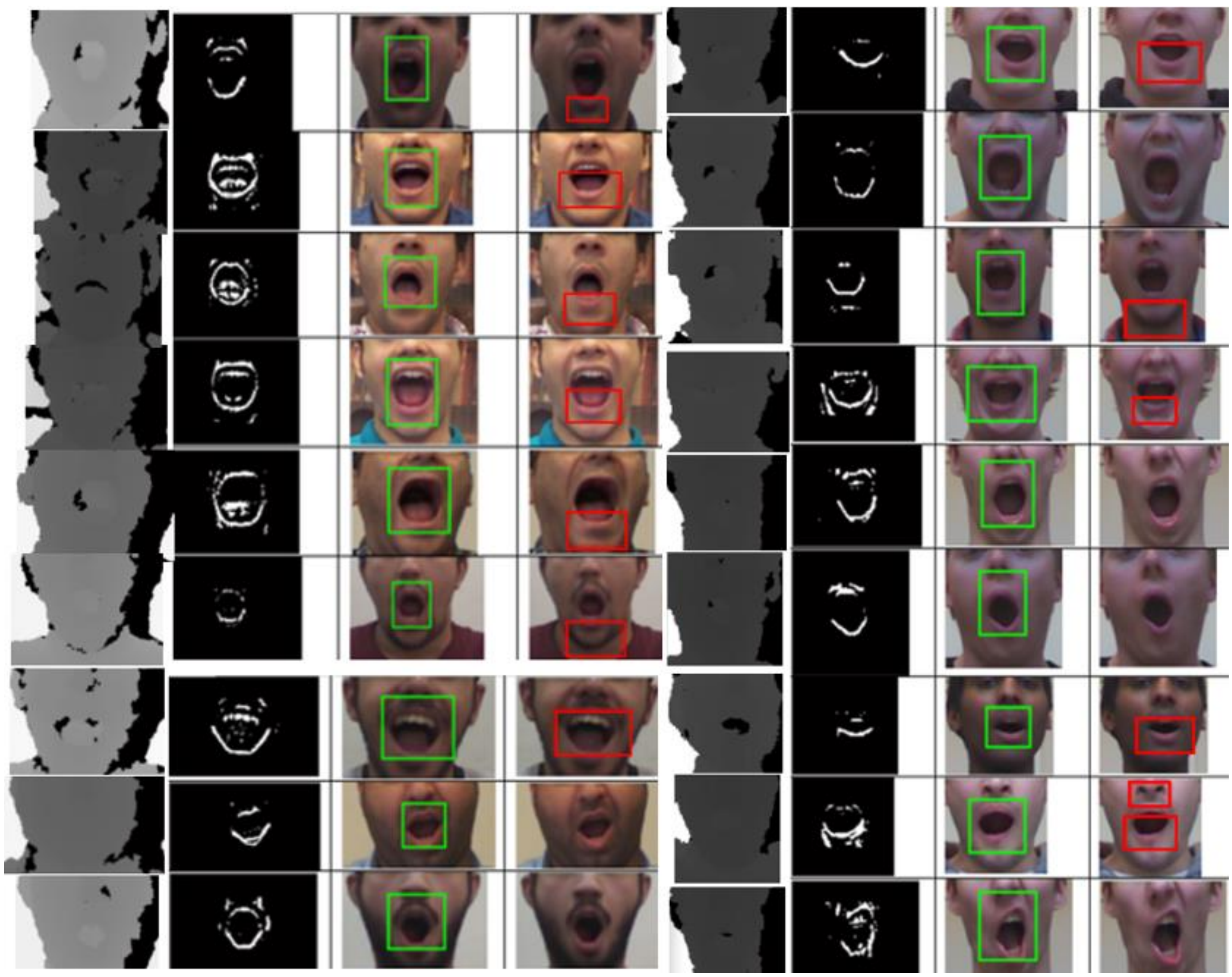

Figure15: A set of depth images containing open mouths (columns 1 and 5), binary images identifying mouth area using the algorithm (columns 2 and 6), color windows placed on the mouth area (columns 3 and 7), detecting mouth area using Viola Jones algorithm (columns 4 and 8).

[DOI: 10.22068/ijae.9.3.3033]

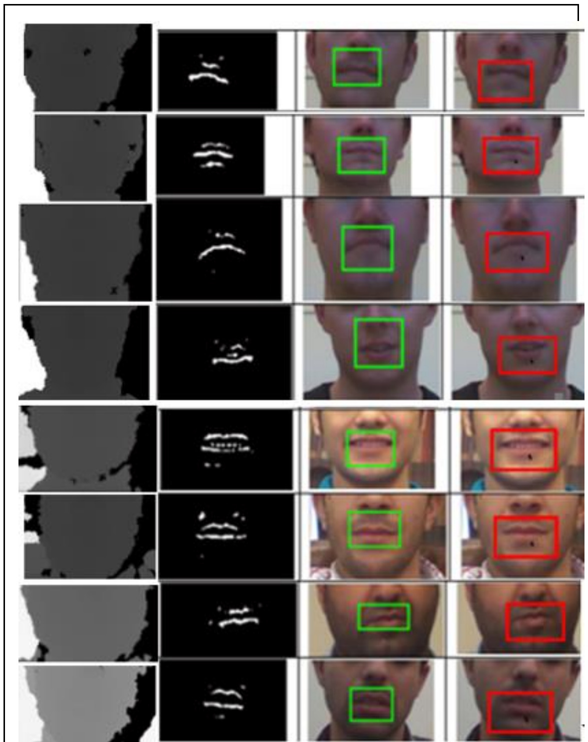

Figure16: A set of depth images containing closed mouths (columns 1), binary images identifying mouth area using the algorithm (columns 2), color windows placed on the mouth area (columns 3), detecting mouth area using Viola Jones algorithm (columns 4).

Table 1: Comparison of the results of Viola Jones mouth detection algorithm and the proposed algorithm

Total samples=180	Viola Jones algorithm		Proposed algorithm	
	Closed mouth	Open mouth	Closed mouth	Open mouth
Total No. of Closed mouth=40	35	5	31	9
Total No. of Open mouth=140	69	71	4	136

3.3. Open mouth detection

After detecting the mouth area, the final step is to detect whether the mouth is open or closed. The mouth state is determined by measuring the height of the open mouth and comparing it with a threshold. By examining open mouth images of several individuals, it was determined that a suitable threshold for the height of an open yawning mouth (HM)1 is 15 lines. We used 100 open mouths and 80 semi-open or closed mouth images for this experiment. Results showed that the algorithm is able to correctly identify 94 out of 100 open mouth cases and 77 out of 80 semi-open or closed cases. Table 2 and Figs 17, 18 and 19 show the results of this experiment.

Figure17: Results of finding the mouth cavity using maximum depth information (first method) for 4 different individuals.

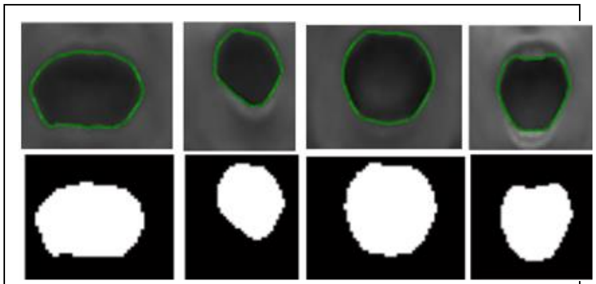

Figure18: Results of applying active contour to the mouth area (second method) for 4 different cases.

[1] Height of the Mouth

[DOI: 10.22068/ijae.9.3.3033]

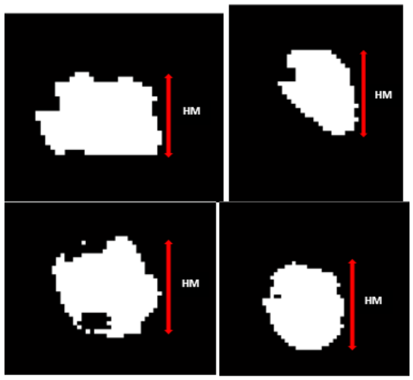

Figure19: Combining the results of Figs. 17 and 18 by obtaining their intersections for 4 different individuals.

Table 2: Accuracy of the results of open mouth detection

No. of cases=180	Predicted Closed mouth	Predicted open mouth
#Closed mouth: 88	77	3
#Open mouth: 100	6	94

$$Accuracy = \frac{TP + TN}{TP + FP + TN + FN} = 95\%$$

3.4. Eyes State Detection

Results of the CNN network used for identification of open and closed eyes are presented in this section. After detecting nose tip in the depth image, the upper part of the face is separated and the Viola Jones algorithm [21] is applied to RGB upper face images to identify the eyes location. Fig. 20 shows how the upper face is separated using the nose tip location. Fig. 21 shows the results of Viola Jones algorithm for detecting eyes location in the RGB upper face images. Then left and right eyes are separated to form input samples for the CNN network. 200 open eye samples and 200 closed eye samples were extracted from the dataset. All image samples were resized to 24*24 to be suitable as the CNN input. Figure 22 shows single eye samples produced for training and testing stages of the network. The deep neural network was trained with 90% of the samples and was evaluated with the rest. In the suggested semi-VGG network, weighing is done through back-propagation and the last layer has two output classes. The network was implemented using Keras package with Tensor flow Backend. Batch size was adjusted to 32, momentum was default Keras, Optimizer = RmsProp, loss function was equal to categorical cross entropy and trained by weight reduction of 10-6. The initial learning rate was adjusted to 10^{-4}. The eye state classifier resulted an accuracy of 90%.

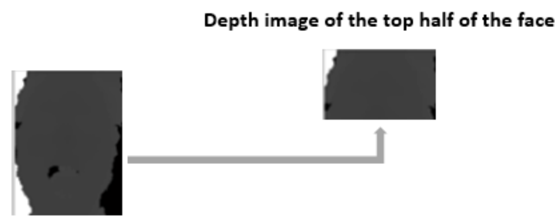

Figure20: Separating the upper half of the face after finding nose tip.

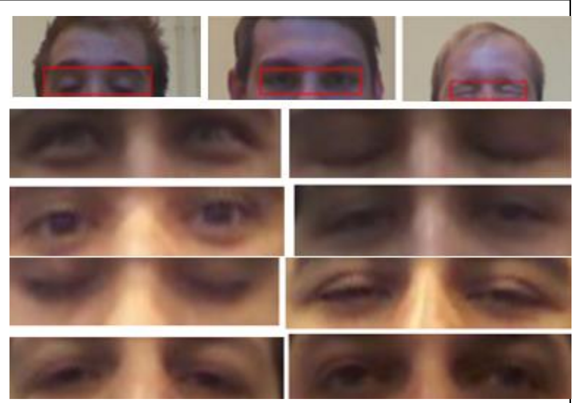

Figure21: Results of finding the location of eyes in RGB upper half face images using Viola Jones algorithm; up: VAP_RGBD database, down: our database.

[DOI: 10.22068/ijae.9.3.3033]

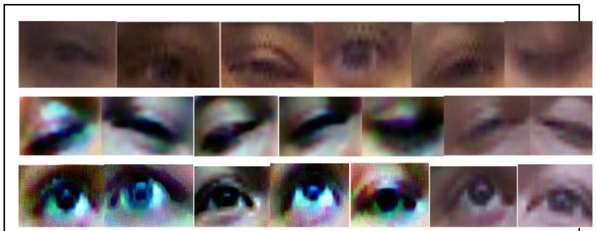

Figure22: Open and closed eye samples from VAP_RGBD and our database.

3.5. Driver Drowsiness and Fatigue Detection

Identifying deriver's fatigue and drowsiness in the suggested work is done through investigating the states of the mouth and the eyes. In these experiments, if 45 consecutive open mouth frames are found, the state of yawning is declared. To identify an eye closure state, the eyes should be closed in at least 15 successive frames. A logical OR is performed on the outputs of the yawning detection stage and the eye closure detection stage to warn the driver. As mentioned before, in order to compensate for the errors in identifying open mouth and closed eyes, the errors can be ignored up to 20% of the frames in each sequence.

4. Conclusion

In this paper, an algorithm was developed to identify driver's fatigue and drowsiness according to his yawning and closed eyes. Detecting open mouth and identifying yawning were initiated by nose tip identification through finding the minimum data in the face depth image. Having the nose tip location, the lower part of the face was separated from the whole image. In the resulting image, by applying an adaptive mask which acts as an edge detector the mouth area was identified with 91% accuracy. Two methods, one based on maximum depth and the other based on active contour model were applied on the mouth area and their results were combined to detect open mouths with 95% accuracy. Also by separating the upper part of the face, the eyes were located and separated using the Viola Jones algorithm. Then using a deep neural network, the open and closed eyes were classified with 90% accuracy. By combining the results of yawn detection and eye closure detection stages the decision to warn the driver is made. Using depth information which is not sensitive to illumination changes is the main advantage of his work. The approach may fail in occasions that the driver's mouth is covered by hand or he wears glasses. Dealing with driver's glasses should be considered in future works.

Acknowledgement:

We would like to express our gratitude to Dr. E. Koozegar for his useful comments and help.

References

[1] Etats-Unis., Healthy people 2010: Final review, Government Printing Office, Department of health, 2013.

[2] Majid Moavenian, Sina Sadeghi namaghi, "An adaptive modified fuzzy-sliding mode longitudinal control design and simulation for vehicles equipped with ABS system", International Journal of Automotive Engineering, IJAE 2019, 9(1): 2895-2907.

[3] M. Salehpour et. al., "Optimum Pareto design of vehicle vibration model excited by non-stationary random road using multi - objective differential evolution algorithm with dynamically adaptable mutation factor", International Journal of Automotive Engineering, IJAE 2018, 8(4): 2854-2867.

[4] B.Mashadi, M. A. Vesal and H. Amani, "Obstacle avoidance for an autonomous vehicle using force field Method", International Journal of Automotive Engineering, IJAE 2017, 7(3): 2166-2181.

[5] Craye, C., & Karray, F. (2015). Driver distraction detection and recognition using RGB-D sensor. arXiv preprint arXiv:1502.00250.

[6] Li, L., Werber, K., Calvillo, C. F., Dinh, K. D., Guarde, A., & König, A. (2014). Multi-sensor soft-computing system for driver drowsiness detection. In soft computing in industrial applications (pp. 129-140). Springer, Cham.

[7] Zhang, L., Liu, F., & Tang, J. (2015). Real-time system for driver fatigue detection by RGB-D camera. ACM Transactions on Intelligent Systems and Technology (TIST), 6(2), 22

[8] Choi, I. H., Tran, T. B. H., & Kim, Y. G. (2016, January). Real-time categorization of driver's gaze zone and head pose using the convolutional neural network. In Proceedings of HCI Korea (pp. 417-422). Hanbit Media, Inc.

[9] Dwivedi, K., Biswaranjan, K., & Sethi, A. (2014, February). Drowsy driver detection using representation learning. In 2014 IEEE International Advance Computing Conference (IACC) (pp. 995-999).

[10] Kim, Y. (2015). Detection of eye blinking using Doppler sensor with principal component analysis. IEEE antennas and wireless propagation letters, 14, 123-126.

[11] Nanni, L., Lumini, A., Dominio, F., & Zanuttigh, P. (2014). Effective and precise face detection based on color and depth data. Applied Computing and Informatics, 10(1-2), 1-13.

[12] Clavijo, G. L. R., Patino, J. O., & León, D. M. (2015, September). Detection of visual fatigue by analyzing the blink rate. In Signal Processing, Images and Computer Vision (STSIVA), 2015 20th Symposium on (pp. 1-5). IEEE.

[13] Sayeed, A., & Sadi, S. A. (2018). Driver Drowsiness Detection using Face Monitoring and Pressure Measurement. Journal of Embedded System & Applications, 5(3), 12-18.

[14] Alioua, N., Amine, A., & Rziza, M. (2014). Driver’s fatigue detection based on yawning extraction. International journal of vehicular technology, 2014.

[15] Ibrahim, M. M., Nur’Afifah, S. B and Manap, N. A., "Development of Yawning Detection Algorithm for Normal Lighting Condition and IR Condition", Journal of Telecommunication, Electronic and Computer Engineering (JTEC) 9, 2-13, 2017.

[16] Zhang, Weiwei, and Jinya Su, "Driver yawning detection based on long short term memory networks", Computational Intelligence (SSCI), IEEE Symposium Series on, IEEE, 2017.

[17] Galarza, E. E., Egas, F. D., Silva, F. M., Velasco, P. M., & Galarza, E. D. (2018, January). Real Time Driver Drowsiness Detection Based on Driver’s Face Image Behavior Using a System of Human Computer Interaction Implemented in a Smartphone. In International Conference on Information Theoretic Security (pp. 563-572). Springer, Cham.

[18] Fong, K. K. (2015). IR-Depth Face Detection and Lip Localization Using Kinect V2. Diss. California Polytechnic State University, San Luis Obispo.

[19] Lin, C. Y., Chang, P., Wang, A., & Fan, C. P. (2018, May). Machine Learning and Gradient Statistics Based Real-Time Driver Drowsiness Detection. In 2018 IEEE International Conference on Consumer Electronics-Taiwan (ICCE-TW) (pp. 1-2). IEEE.

[20] VAP_RG Database, http://www.vap.aau.dk.

[21] Viola P, Jones M. Rapid object detection using a boosted cascade of simple features. CVPR (1). 2001 Dec 8; 1:511-8.

[DOI: 10.22068/ijae.9.3.3033]