JOIN (Jurnal Online Informatika)
p-ISSN: 2528-1682, e-ISSN: 2527-9165
Volume 9 Number 2 | December 2024: 238-248
DOI: 10.15575/join.v9i2.1076

Detection of Drowsiness in Drivers Using Image Processing and Support Vector Machine (SVM) Classification

Sugeng[1], Hendri Praminiarto[2]
[1,2] Computer Systems Study Program, Faculty of Engineering and Computer Science, Indonesian Computer University, Bandung, Indonesia

Article Info

Article history:

Received May 31, 2023
Revised May 08, 2024
Accepted June 06, 2024
Published December 27, 2024

Keywords:

Artificial Intelligence
Classification
Drowsiness Detection
Machine Learning
Support Vector Machine

ABSTRACT

Accidents can be caused by external factors on the road, vehicle conditions, or internal factors such as drowsiness. Drowsiness while driving poses risks to the driver and others. An early detection system is crucial to alert drivers to stop or rest if they show signs of drowsiness. Physical signs of drowsiness include a lethargic facial expression, frequent eye blinking, continuous yawning, or nodding off. A detection system utilizing image processing and machine learning can observe these signs by detecting facial landmarks and analyzing activities such as eye blinking, yawning, and head tilt. This study aims to classify the drowsiness condition based on these three factors. The classification process is conducted using machine learning with the Support Vector Machine (SVM) method to determine whether a person is drowsy or not. The dataset consists of the number of eye blinks, head tilts, and yawns. Conditions are classified into two classes, drowsy and not drowsy. In this study, the SVM classification method can predict drowsiness with an accuracy of up to 77% in the conducted tests.

Corresponding Author:

Sugeng,
Computer Systems Study Program, Faculty of Engineering and Computer Science, Indonesian Computer University, Bandung
Jl. Dipatiukur 112-114, Bandung, Indonesia. 40614
Email: sugeng@email.unikom.ac.id

1. INTRODUCTION

Accidents that occur claim 1.3 million lives each year, accidents often occur not only due to external factors such as improper vehicles, damaged roads or inadequate street lighting, but there are internal factors, namely humans [1] itself, one of which is drowsiness . Drowsiness is one of the things that commonly occurs in motorists, especially when the driver has traveled a long way causing fatigue and drowsiness while driving. Momentary drowsiness usually occurs in motorists or what is often referred to as *microsleep.* This condition occurs suddenly and usually lasts for 1 second to 2 seconds and not everyone cares about this condition. Even though for motorists, if they continue their journey with these conditions, it will certainly be very dangerous because it can result in an accident [2]. The condition of the driver who is experiencing drowsiness can be seen from the increasing number of continuous blinks [3].

There are many cases of accidents caused by drowsiness that claim the most victims in terms of internal factors [4] about 80% of accidents are caused by human error[5], data from the Republic of Indonesia Police, 61% is caused by driver behavior[6]. As solution is needed to overcome them, one of which is by using facial recognition and seeing facial activity whether you are sleepy or not. Sleepy conditions can be seen from the closed eyes and the condition of someone who yawns frequently. Through face detection and activity detection from the face it will help in determining whether a person is sleepy or not. Face detection can be done with machine learning by quoting important information contained in faces [6].

Face recognition is a technology used for *biometric recognition* based on human facial features [7]. Basically, facial recognition in a computer is a process in which faces match an image with a face that has been indexed in the system [8]. By using facial recognition, it can detect whether the driver is sleepy or not.

Face recognition is used to map faces based on points that have special characteristics [9]of the driver so that the correct facial position is found. There are various methods of face detection. One of the popular methods is the *blazeface method* with an accuracy of up to 98.61% [10]. With a high level of precision and light weight, *the blazeface* is very suitable for a drowsiness detection system for motorists. The *blazeface* method can run very fast reaching 200 – 1000+ FPS for certain devices.

To be able to detect eye blinking and yawning mouth activity, an EAR (Eye Aspect Ratio) method can be used [12], in which this method measures the ratio between the width and height of a shape. The shape of the eyes and mouth when open and closed will certainly be different. The difference in this ratio will be used to determine whether the eyes are blinking or not.

Furthermore, an *inverse kinematics method is used* to determine the degree of tilt of the head. The position of the rider's head can be observed by measuring the degree of tilt of the rider's head. When you are sleepy, of course the position of the tilt of the head will be different from the condition when you are still fit.

Classification is used to determine whether a person is drowsy or not using the *Support Vector Machine* (SVM) classification. SMV works more effectively than deep learning on smaller datasets because the program is not complex so it does not burden machine performance. The SVM classification method is a classification method that classifies an object by finding the largest [12]where the larger *the hyperplane* , the better the results, this method is the best method in determining classification results with a high degree of precision, and high accuracy [13].

In previous research in determining sleepiness limited to the eye and mouth area [14], by D. H. Fudholi et al, fatigue detection using deep learning only uses the number of eye blinks [15]. R.T Puteri and F Utaminingrum, drowsiness detection using haar sequence and convulutional neural network where data is only taken from how long the eyes are closed [16]. D Amalia and F Utamaningrum conducted research on sleepiness detection using facial landmarks to detect only the number of blinks [17]. C. A. Saputra, D Erwanto and P.N. Rahayu conducted research on drowsiness detection based only on the length of time the eyes are closed, if it is more than three seconds then the driver is said to be sleepy [2]. Dean C. J. M. Pardede, et al, detects sleepiness through Raspberry Pi-based eye tracking, the study only calculates the aspect ratio of the eyes as an indicator of whether sleepy or not [12]. No research was carried out based on the number of blinks or the condition of the tilt of the head. In this study the authors added additional parameters in detecting drowsiness to get more accurate results than previous studies.

This research was conducted to determine the condition of the driver who is sleepy or not by detecting eye blinks, the amount of yawning and the number of nods or tilts of the head so that accurate results will be obtained in determining whether the driver is sleepy or not, by using blazeface, SVM and inverse kinematics it *will* be *very* suitable used because the process runs fast and also a high level of accuracy.

It is hoped that this research can be helpful and useful for readers, and can also be useful in reducing the number of accidents due to drowsiness and fatigue.

2. METHOD

The research method used in this research is to start by collecting datasets through the camera and then carry out the data *training process*. The model that has been successfully built is then used to detect drowsiness on drivers in real time. In Figure 1 is a block diagram that explains the entire system created.

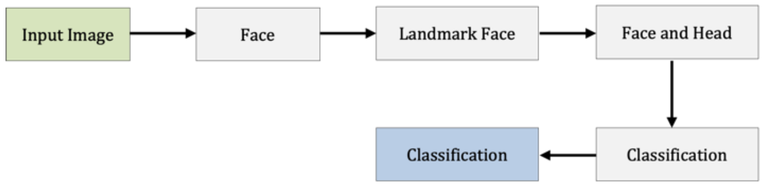


Figure 1. Block diagram of the sleep detection system

p-ISSN: 2528-1682
e-ISSN: 2527-9165

The sleep detection process starts with entering an image, then continues with the face detection process in the image. After the face is detected, the next process is the process of detecting *face landmarks* which are points on the face that form faces. Points of *face landmarks* are used to detect the movement of the eyes, mouth and tilt of the head. The collected activity data is then processed with machine learning, namely the SVM classification to determine whether a driver is experiencing drowsiness or not.

2.1 Face Detection

The face detection process is an image processing process in detecting a person's face. Face detection is the same as other object detection that utilizes object features in the image. Object features are extracted with a certain method and then a classification process is carried out to determine what objects are in an image. One of the most popular face detection methods is face detection with the haar-cascade method.

Other methods in face detection such as Mediapipe which is a *machine learning development model* that can work on *multiple platforms* or can work on various devices and various programming languages used[18] [20], mediapipe provides *end to end acceleration* which makes *machine learning* be faster to use. Due to its ability to be developed using various programming languages, Mediapipe is suitable for mobile and desktop devices. Mediapipe can work in *real time* and not only detect faces but mediapipe can work for detecting objects, hand movements, bodies, and other things related to *machine learning.*

2.2 *Face* detection of landmarks

After the system is able to detect faces in the image field, the next process is to detect the face landmad key point on the face. *Face landmark* key points are connecting points that compose a face shape in detail based on the shape of a person's face. In addition, *the face landmark* key point also accurately maps the position of facial parts, such as the eyes, nose, mouth and eyebrows. On the mediapipe there are 468 *face landmark* key points that can be used to detect movement or activity of the eyes, mouth and tilt of a person's head show on Figur 2. The points on key point landmarks can be used to determine changes in activity on the face. When the eyes blink or the head changes position, these points can be used as a reference for changes in activity.

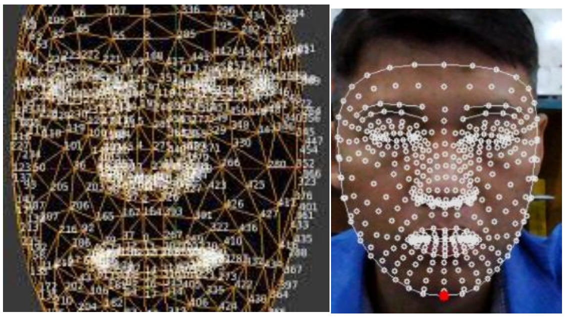


Figure 2. 468 Face Landmark Key Point

2.3 Collecting Datasets

A dataset is a collection of reference data that is used to train a model with machine learning or to generate an artificial intelligence model. Datasets can be obtained through sites that provide special datasets in bulk or from institutional and company websites related to the research being carried out. Datasets can also be made independently by taking into account the needs of the system being built. In machine learning, the dataset is made as much as possible in order to produce better machine accuracy.

2.4 Data Training and Model Testing

The next process in the research after dataset collection is the classification system training process which is built using the dataset that has been collected. The created algorithm system is trained with a dataset with a total of 70% of the entire dataset. The remaining 30% is used to carry out the process of testing models that have been successfully trained. There are no restrictions on the division between data for train and data for test. In general, datasets can be divided into 80:20, 70:30 and even

50:50. 70:30 was chosen based on the condition of the dataset which is not too large so it is hoped that this composition is sufficient to represent all the data held.

2.5 SVM classification

Support Vector Machine (SVM) is a supervised learning method commonly employed for classification (like Support Vector Classification) and regression (Support Vector Regression). In classification modeling, SVM exhibits a more mature and mathematically clear concept compared to other classification techniques. SVM can handle both linear and non-linear classification and regression problems.[19]

SVM seeks the best hyperplane by maximizing the distance between classes. The hyperplane is a function used to separate classes. In 2-D, the function used for inter-class classification is called a line, whereas in 3-D, it's referred to as a plane. Similarly, in higher-dimensional class spaces, the function used for classification is termed a hyperplane. Figure 3 shows an example of a Hyperplane that separates two classes positive (+1) and negative (-1)[20].

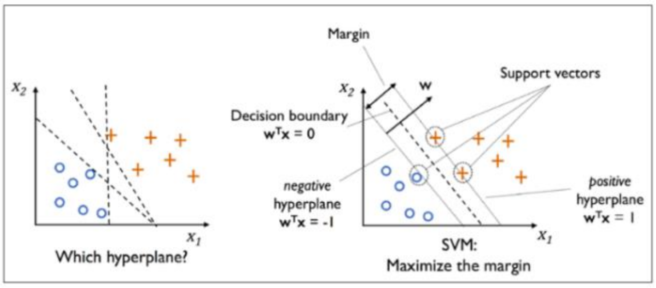


Figure 3. Sample Hyperplane that separates the two classes positive (+1) and negative (-1)

2.6 System Testing

The last process is the process of testing the system as a whole to detect the drowsiness of a car driver. The system is tested to be able to detect the activity of eye blinking, the amount of yawning, and to detect the tilt of the driver's head. After that, the system must also be able to estimate or predict the drowsiness that occurs in the driver based on the activity of blinking, yawning, and head tilt.

3. RESULT AND DISCUSSION

The drowsiness detection system for motorists works according to *the flowchart* in Figure 4. By entering the video from the webcam attached to the system. The system will retrieve data for each frame from the video input, and then carry out the preprocessing process and the face detection process. After the facial data has been successfully detected, the next process is the process of looking for landmark points on the face.

The next stage is to calculate the activity of eye blinking, mouth opening or yawning, and head tilt activity. All these conditions are recorded by the system within 10 seconds. All conditions that have been recorded will then be classified using machine learning with the SVM method. The output of this classification is a sleepy state or not. If it turns out that the recorded data is in a drowsy state, the system will turn on the buzzer as an indicator that the user is experiencing a drowsy state.

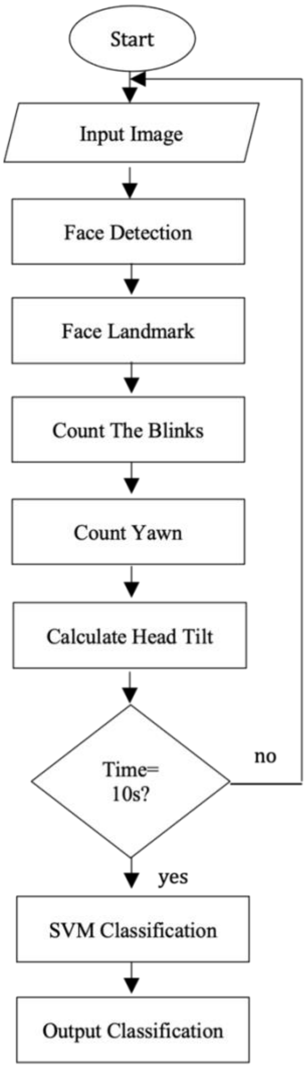


Figure 4. Drowsiness Detection System Process Flow

3.1. Detect eye activity, mouth and head tilt

Eye Aspect Ratio method (EAR**)** is a method that can be used to determine the condition of the eyes in a closed or open state. The aspect ratio of the eye can be determined through simple mathematical calculations by utilizing the points on the *face landmarks*.

Face landmarks on the eyes which are divided into 16 points for the left eye and 16 points for the right eye. *Landmarks* on the eye are only used as many as 6 points to be able to calculate EAR. EAR is calculated by formula (7) which consists of points p1-p6 where p1-p6 of the left eye are points with numbers *[362, 385, 387, 263, 373, 380]*. while the right eye side has p1-p6 with index numbers *[33, 160, 158, 133, 153, 144]*. The p1 – p6 values on *the face landmarks* of the eyes appear as shown in Figure 5 below.

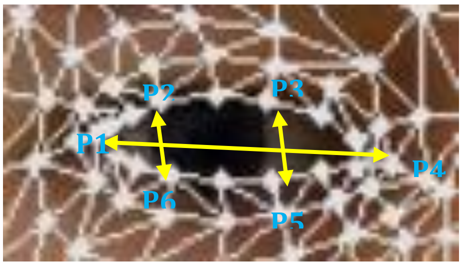
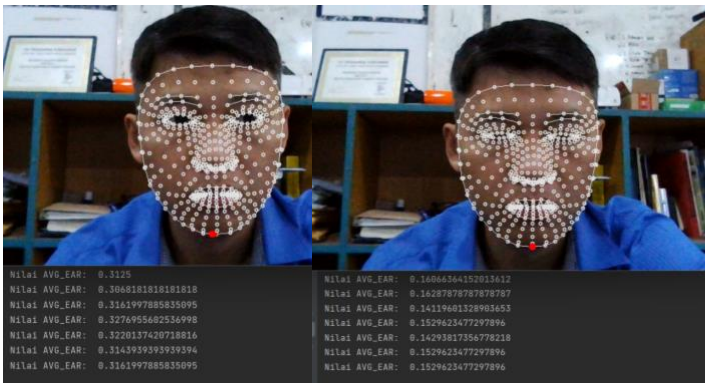


Figure 5. Determination of Aspect Ratio in the Eyes

Eye Aspect Ratio formula (EAR**)** can be seen in the following equation:

$$EAR = \frac{|p2-p6|+|p3-p5|}{2|p1-p4|} \quad (7)$$

The results of testing the system in detecting eye activity are shown in Figure 6. When the eyes are open, the system records an average EAR value of 0.31, while the average EAR value when the eyes are closed is 0.15.



(a) EAR eyes open (b) EAR eyes closed

Figure 6. Calculation of the eye EAR value

3.1.1 Yawning Detection

The process for detecting yawning is the same as detecting eyes closed. By utilizing the EAR equation, the aspect ratio when the mouth opens (yawns) and closes will show the difference. At the mouth of the points used to calculate the aspect ratio are at points [96, 82, 312, 325, 317, 87]. if converted to position coordinates in the image it looks like in the following example program:

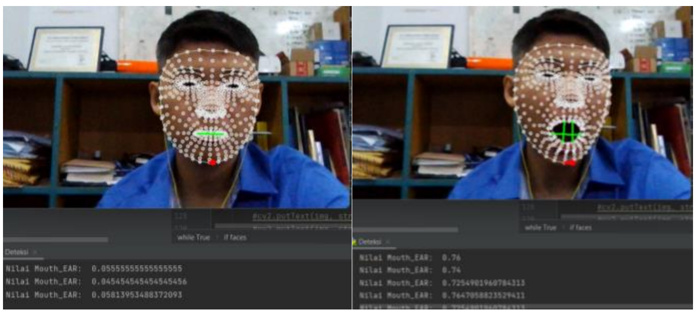
```
M_p1x , M_p1y = faces[ 0 ][ 96 ] # M-P1
M_p2x , M_p2y = faces[ 0 ][ 82 ] # M_p2
M_p3x , M_p3y = faces [ 0 ][ 312 ] # M_p3
M_p4x , M_p4y = faces[ 0 ][ 325 ] # M_p4
M_p5x , M_p5y = faces[ 0 ][ 317 ] # M_p5
M_p6x , M_p6y = faces[ 0 ][ 87 ] # M_p6
```

From the coordinates of the image pixels, the ratio value of the mouth can then be calculated with the previous equation, if it is made in the program it looks like this:

```
EAR_Mouth =
( abs ( M_p2y - M_p6y)+ abs (M_p3y - M_p5y)) / ( 2 * abs (M_p1x - M_p4x))

print ( "Value of Mouth_EAR: " , EAR_Mouth)
```

Figure 7 is the output of the program to see the results of the open and closed mouth ratio values. When closed the value of the mouth ratio is around 0.05. While when the mouth is open the value of the ratio is around 0.75.



(a) Closed mouth ratio (b) Aspect ratio open mouth

Figure 7. Mouth Aspect Ratio Calculation

3.1.2 Head Tilt Detection

The tilt of the head of the driver is one of the indicators that is calculated to determine whether a person is sleepy or not, the tilt of the head is calculated using the angular change formula in the inverse kinematic equation. In the program to be able to retrieve the middle value from advance can be done with the following command:

```
fx11, fy11 = faces[ 0 ][ 10 ]
fx12, fy12 = faces[ 0 ][ 152 ]

fx_imaginer = fx11fy_imaginer = fy12imaginer_length = 0
```

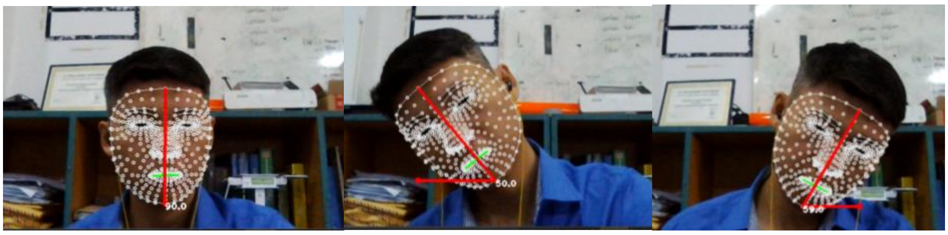
Calculation of kinematic inverse as shown in the following sample program snippet:

```
face_length = math.sqrt ( pow ((fx12 - fx11) , 2 ) + pow ((fy12 - fy11) , 2 ))
```

```
imaginer_length = math.sqrt ( pow ((fx12 - fx_imaginer) , 2 ) + pow ((fy12 - fy_imaginer) , 2 ))
```

```
face_degree = math. degrees (math. acos((imaginer_length / face_length)))
```

The test results to determine the tilt angle of the head can be seen in Figure 8. When the head is straight, the angle formed is 90 degrees (Figure 8a). In figure 8b, the head can be seen tilted to the left at an angle of 50 degrees, finally in figure 8c. Head tilted to the right at an angle of 59 degrees. This slope value is used to determine whether a person is sleepy or not.



(a) Head tilted to the left (b) Head tilted to the right (c) Head straight

Figure 8. Testing the tilt of the head.

3.2. Collection of Data Sets

Data collection was carried out by finding respondents to collect data on whether they were sleepy or not by counting the number of blinks, the number of mouth openings or yawns, and the number of head nods, which will then be used as a dataset.

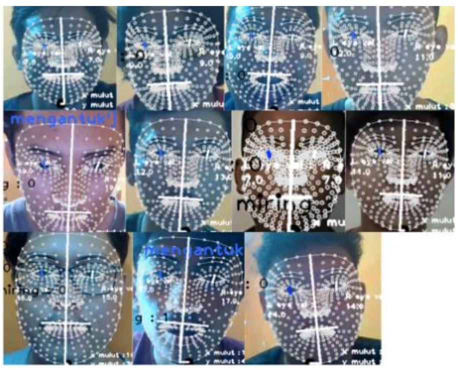


Figure 9. Drowsiness Dataset Collection

Figure 9 is a respondent whose data is taken for being sleepy or not using a method that has been stored in the system. In taking the dataset, the respondents were made sleepy and bored by being given a boring show to find out the number of blinks and drowsiness, and data were collected during the day and at night to find out whether the respondent was sleepy due to sleeping hours or indeed due to boredom due to eye fatigue. The data obtained from the respondents was then used as a sleepy dataset with a total of 110 data as follows.

Table 1. Respondents' eye blinking, yawning and head tilt activity data

Number of flashes	The number of tilted heads	Amount of evaporation	Condition
5	1	2	sleepy
5	1	1	sleepy
3	1	2	sleepy
....	....	....	....
2	1	0	not sleepy
1	0	0	not sleepy
2	0	0	not sleepy

From table 1, a simple data set is produced that can be used to determine whether a person is sleepy or not according to the number of eye blinks, the number of yawns and the tilt of the head.

3. 3. Dataset Testing

After collecting a dataset of sleepy and not sleepy conditions, then this data can be used as training data on the SVM algorithm that has been created. The data is divided into training data and test data. The training data consists of 70% of the total data, the rest is used as test and validation data. From the training process to the created algorithm, an accuracy of 93%, *94% recall,* and 94% precision is obtained. The program output resulting from the machine learning process looks like in Figure 10.

```
Accuracy  : 0.93939394
Precision : 0.94736842
Recall    : 0.94736842
```

Figure 10. Output Of Training Process

In testing the dataset, it was found that the dataset used had an accuracy rate of up to 93%, the precision level reached 94% and *the recall* reached 94%.

After the dataset is obtained, then the SVM model is created using *a pickle* where the training data model will be made separately from the test data. In this study, the test data is *real-time* data from the condition of the driver, while the training data is made using the SVM model.

3. 4. Drowsiness Detection System Testing

Furthermore, to prove the drowsiness detection tool, a test is carried out on the system that has been built using several test parameters including:

Tool testing with ideal light conditions:

Drowsiness detection test will test whether the system can detect drowsiness and fatigue in drivers. Then do the system test in the following way:

a. The camera distance to the face is 25 – 40 cm.
b. Using a webcam with 720p resolution and 720p IR camera
c. The lighting conditions of the room are bright and clear
d. The condition of the activity of the eyes, mouth and head is detected every 10 seconds.

The test results are shown in Table 2. The input data from the system is the number of eye blinks, the number of head tilts and the number of yawns.

Table 2. The results of the sleep detection test

No	Photo	Input Data	Information
1		1, 1, 2	Sleepy (Correct)

No	Photo	Input Data	Information
2		2, 0, 2	Sleepy (False)
3		0, 1, 0	Not Sleepy (Correct)
4		1, 0, 3	Sleepy (Wrong)
5		1, 1, 0	Not sleepy (Correct)
6		2, 1, 0	Not Sleepy (Correct)
7		2,1,0	Not sleepy (Correct)
8		2,2,0	Not sleepy (Correct)
9		1,3,1	Sleepy (Correct)

From table 2 it can be seen that there were two errors out of a total of nine trials in detecting a person's drowsiness. So that can be said. The sleep detection system has an accuracy of 77% in detecting drowsiness

Testing the face distance to the camera:

In this test it was found that the system could detect distances between 10 cm to 60 cm. The test results are shown in Figure 11.

(a) Can work (b) Can work
Figure 11. Testing the face distance to the camera

Low light testing:

In low-light testing, the system runs with an accuracy rate of 100%, this is because the author uses an infrared camera so that it can operate in the dark. The test results are shown in Figure 12.

(a) Low Light Conditions (b) Total Darkness

Figure 12. Test results in dark light conditions

Testing using accessories:

In this test the system can run well when the face is wearing glasses and a hat or other head cover but cannot work when wearing a mask, show in Figure 13.

(a) Can Work (b) Unable to Work

Figure 13. Test results in dark light conditions

Testing with face position:

Another test is to test the angle of the face towards the camera. The user's face can be tilted left and right, bowed or turned upwards. The slope angle is then measured to determine the slope position that is still detected by the system. The test results are shown in Table 6. In this test it was found that the system could run up to a position of 10º to 20º

3. CONCLUSION

The conclusion of this study is that the sleep detection system can work by utilizing the methods used, namely the blazeface method, *invert kinematics,* and SVM classification to detect drowsiness and can run with a fast but accurate response with an accuracy rate of up to 77% in actual testing. While the SVM model developed has a level of accuracy that reaches more than 90%. Meanwhile, the sleep detection system can only be used at a distance of between 30cm and 60cm from the camera towards the face, with a tilt of the face to the camera of 20-30 degrees. For researchers who will conduct research related to sleepiness detection by utilizing sleepiness characteristics based on eye activity, mouth, and head tilt, they can create more datasets so that detection results are even better.

REFERENCES

[1] G. Gloria, 'Pakar UGM Sebut Empat Faktor Penyebab Kecelakaan di Jalan Tol | Universitas Gadjah Mada'. Accessed: May 19, 2023. [Online]. Available: https://ugm.ac.id/id/berita/21920-pakar-ugm-sebut-empat-faktor-penyebab-kecelakaan-di-jalan-tol

[2] C. Aj. Saputra, D. Erwanto, and P. N. Rahayu, 'Deteksi Kantuk Pengendara Roda Empat Menggunakan Haar Cascade Classifier Dan Convolutional Neural Network', *J. Electr. Eng. Comput. JEECOM*, vol. 3, no. 1, Art. no. 1, Apr. 2021, doi: 10.33650/jeecom.v3i1.1510.

[3] A. H. A. P. Perdana, S. T. Rasmana, and H. Pratikno, 'Implementasi Sistem Deteksi Mata Kantuk Berdasarkan Facial Landmarks Detection Menggunakan Metode Regression Trees', *J. Technol. Inform. JoTI*, vol. 1, no. 1, Art. no. 1, Oct. 2019, doi: 10.37802/joti.v1i1.1.

[4] A. Alfandianto, C. E. Suharyanto, and F. N. D. Prasasti, 'Analisis Regresi Linier Guna Mengetahui Pengaruh Tingkat Kematian Kecelakaan Lalu Lintas Terhadap Faktor Mabuk', *SAINTEK J. Ilm. Sains Dan Teknol. Ind.*, vol. 5, no. 1, Art. no. 1, Jul. 2021, doi: 10.32524/saintek.v5i1.246.

[5] PT VIVA MEDIA BARU- VIVA, 'KNKT: 80 Persen Kecelakaan di Tol Akibat Mengantuk dan Letih'. Accessed: May 23, 2023. [Online]. Available: https://www.viva.co.id/berita/nasional/1427758-knkt-80-persen-kecelakaan-di-tol-akibat-mengantuk-dan-letih

[6] P. KOMINFO, 'Rata-rata Tiga Orang Meninggal Setiap Jam Akibat Kecelakaan Jalan', Website Resmi Kementerian Komunikasi dan Informatika RI. Accessed: May 08, 2024. [Online]. Available: http:///content/detail/10368/rata-rata-tiga-orang-meninggal-setiap-jamakibat-kecelakaan-jalan/0/artikel_gpr

[7] S. Sugeng and A. Mulyana, 'Sistem Absensi Menggunakan Pengenalan Wajah (Face Recognition) Berbasis Web LAN', *J. Sisfokom Sist. Inf. Dan Komput.*, vol. 11, no. 1, Art. no. 1, Apr. 2022, doi: 10.32736/sisfokom.v11i1.1371.

[8] S. I. Lestariningati, A. B. Suksmono, I. J. M. Edward, and K. Usman, 'Group Class Residual ℓ1-Minimization on Random Projection Sparse Representation Classifier for Face Recognition', *Electronics*, vol. 11, no. 17, Art. no. 17, Jan. 2022, doi: 10.3390/electronics11172723.

[9] M. A. Prastya, 'SISTEM PENGENALAN WAJAH MANUSIA MENGGUNAKAN ALGORITMA VIOLA-JONES DAN PRINCIPAL COMPONENT ANALYSIS', *Ubiquitous Comput. Its Appl. J.*, vol. 2, no. 2, Art. no. 2, Dec. 2019, doi: 10.51804/ucaiaj.v2i2.85-92.

[10] D. Erwan, Y. Apridiansyah, E. D. Putra, and U. Juhardi, 'Algoritma Haar Cascade Deteksi WajahMenggunakan Phyton', *JUKOMIKA J. Ilmu Komput. Dan Inform.*, vol. 5, no. 2, Art. no. 2, Jan. 2023, doi: 10.54650/jukomika.v5i2.461.

[11] V. Bazarevsky, Y. Kartynnik, A. Vakunov, K. Raveendran, and M. Grundmann, 'BlazeFace: Sub-millisecond Neural Face Detection on Mobile GPUs'. arXiv, Jul. 14, 2019. doi: 10.48550/arXiv.1907.05047.

[12] D. C. J. M. Pardede, A. M. Rumagit, and S. T. G. Kaunang, 'Deteksi Pengendara Mengantuk Menggunakan Metode Eye Tracking Berbasis Rasberry Pi'. Accessed: May 08, 2024. [Online]. Available: http://repo.unsrat.ac.id/3581/

[13] V. I. Santoso, G. Virginia, and Y. Lukito, 'PENERAPAN SENTIMENT ANALYSIS PADA HASIL EVALUASI DOSEN DENGAN METODE SUPPORT VECTOR MACHINE', *J. Transform.*, vol. 14, no. 2, Art. no. 2, Jan. 2017, doi: 10.26623/transformatika.v14i2.439.

[14] R. Rizal, I. Girsang, and S. Prasetiyo, 'Klasifikasi Wajah Menggunakan Support Vector Machine (SVM)', *REMIK Ris. Dan E-J. Manaj. Inform. Komput.*, vol. 3, p. 1, Mar. 2019, doi: 10.33395/remik.v3i2.10080.

[15] D. H. Fudholi, R. A. N. Nayoan, M. Suyuti, and R. Rahmadi, 'Deteksi Indikasi Kelelahan Menggunakan Deep Learning', *J-SAKTI J. Sains Komput. Dan Inform.*, vol. 5, no. 1, Art. no. 1, Mar. 2021, doi: 10.30645/j-sakti.v5i1.292.

[16] R. T. Puteri and F. Utaminingrum, 'Deteksi Kantuk Menggunakan Kombinasi Haar Cascade dan Convolutional Neural Network', *J. Pengemb. Teknol. Inf. Dan Ilmu Komput.*, vol. 4, no. 3, Art. no. 3, Jun. 2020.

[17] Dewi Amalia and Fitri Utaminingrum, 'Deteksi Kantuk pada Pengemudi melalui Jumlah Kedipan Mata Menggunakan Facial Landmark berbasis Intel NUC | Jurnal Pengembangan Teknologi Informasi dan Ilmu Komputer'. Accessed: May 23, 2023. [Online]. Available: https://j-ptiik.ub.ac.id/index.php/j-ptiik/article/view/10283

[18] Mediapipe, 'MediaPipe', Google for Developers. Accessed: May 19, 2023. [Online]. Available: https://developers.google.com/mediapipe

[19] 'A comprehensive survey on support vector machine classification: Applications, challenges and trends', *Neurocomputing*, vol. 408, pp. 189–215, Sep. 2020, doi: 10.1016/j.neucom.2019.10.118.

[20] R. Berwick, 'An Idiot's guide to Support vector machines (SVMs)'.