

# Implementation of Driver Drowsiness Alert and Automatic Vehicle Control System Using Arduino

Akash gowda BR<sup>1\*</sup>, Dharmesh Bhavda<sup>2</sup>, MD Ismail zabi ulla<sup>3</sup>, Swetha N<sup>4</sup>

<sup>1,2,3,4</sup>Department of Computer Science, East West Institute of Technology, Bangalore, India

DOI: <https://doi.org/10.26438/ijcse/v7si15.220225> | Available online at: [www.ijcseonline.org](http://www.ijcseonline.org)

**Abstract**—The model is about making cars more intelligent and interactive which may notify or resist user under unacceptable conditions, they may provide critical information of real time situations to rescue or police or owner himself. Driver fatigue resulting from sleep deprivation or Drunken driving is prevented is an important factor in the increasing number of accidents on today's roads. In this paper, we describe a real -time safety prototype that stop vehicle ignition. The purpose of such a model is to advance a system to detect fatigue symptoms in drivers and control the vehicle to avoid accidents. In this paper, we propose a driver drowsiness detection and alcohol detection system in which alcohol detection sensor and image processing are used for detecting the driver. If the driver is found to have sleep, or driver in an alcoholic condition then buzzer will start and then turns the vehicle ignition off.

**Keywords**—Component, Formatting, Style, Styling, Insert (key words)

## I. INTRODUCTION

The number of deadly traffic accidents are increasing every year with the increase in number of vehicles, thereby increasing the damage in properties. One of the main causes for these ever-increasing accidents is driver drowsiness and alcohol consumption. According to the global status report on road safety given by WHO which reflects information from about 180 countries has indicated that worldwide the total number of road traffic death has plateaued at 1.25 million per year, with India reporting about 1.34 lakh fatalities in road accidents every year, a vast 70 percent of them being due to drunken driving. A way to minimize this vast number is to use advanced techniques for driver assistance. Driver monitoring can be done by using two ways: direct and indirect driver related measures. Direct driver related measures include head movement; facial expressions obtained using camera sensors. Indirect driver related measures consist of driver activities, response to specific situation. The drowsiness of driver can be encountered by detecting series of actions performed by driver which includes eye activities, the amount of time and frequency of closing eyes, head displacement measured by considering centre of gravity. While looking for pedal and steering actions to estimate drowsiness of driver, delay in steering, lateral shifting of vehicle and standard deviation of distance to a leading vehicle are more often used. However, most of these techniques are less reliable. Due to difference in physical properties from person to person, driver cannot be correctly judged by physiological measures. Applications

based on these methods could be impractical in real world due to difficulty in a driver monitoring system. This study focuses on continuous detection of the driver's current state using image processing. A driver is given an alert as soon as symptoms of drowsiness are detected. This can be done by using an alarm. Alcohol consumed can be detected using a sensor

## II. RELATED WORK

Yuichi Saito, Makoto Itoh, Toshiyuki Inagaki [1] have described an assistance system which effectively prevents sleep related accidents. They presented a multi-layered assistance with dual control scheme which can help in reducing sleep related accidents. This assistance system interacts with driver to determine driver's state in a multi-layered way. They used driving simulator which was equipped with assistance system for investigating the effectiveness of finding out drowsiness of driver and avoiding accidents due to lane departure.

[2.Anjali K U, Athiramol K Thampi, Athira Vijayaraman, Franiya Francis M, Jeffy James N, Bindhu K Rajan] Proposed a driver alertness detection system depending on fatigue detection at the given instant. They used image processing algorithms to identify the position of eyes. They obtained visual cues by observing eye blink rate using camera which characterize the alertness level of a driver.

## III. METHODOLOGY

### **Methodology**

It consists of Arduino

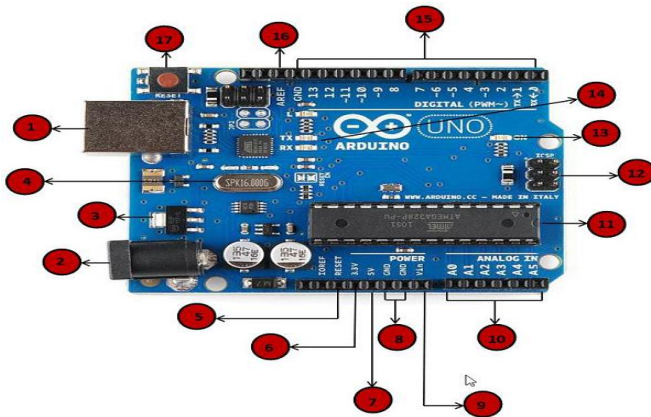


Fig 1: ARDUINO

### POWER SUPPLY

Arduino board can be powered by using the USB cable from your computer. All you need to do is connect the USB cable to the USB connection (1).

### Power (Barrel Jack)

Arduino boards can be powered directly from the AC mains power supply by connecting it to the Barrel Jack (2).

### Voltage Regulator

The function of the voltage regulator is to control the voltage given to the Arduino board and stabilize the DC voltages used by the processor and other elements.

### Crystal Oscillator

The crystal oscillator helps Arduino in dealing with time issues. How does Arduino calculate time? The answer is, by using the crystal oscillator. The number printed on top of the Arduino crystal is 16.000H9H. It tells us that the frequency is 16,000,000 Hertz or 16 MHz.

### Arduino Reset

You can reset your Arduino board, i.e., start your program from the beginning. You can reset the UNO board in two ways. First, by using the reset button (17) on the board. Second, you can connect an external reset button to the Arduino pin labelled RESET (5).

### Pins (3.3, 5, GND, Vin)

- 3.3V (6) – Supply 3.3 output volt
- 5V (7) – Supply 5 output volt
- Most of the components used with Arduino board works fine with 3.3 volt and 5 volt.
- GND (8)(Ground) – There are several GND pins on the Arduino, any of which can be used to ground your circuit.

- Vin (9) – This pin also can be used to power the Arduino board from an external power source, like AC mains power supply.

### Analog pins

The Arduino UNO board has five analog input pins A0 through A5. These pins can read the signal from an analog sensor like the humidity sensor or temperature sensor and convert it into a digital value that can be read by the microprocessor.

### Main microcontroller

Each Arduino board has its own microcontroller (11). You can assume it as the brain of your board. The main IC (integrated circuit) on the Arduino is slightly different from board to board. The microcontrollers are usually of the ATMEL Company. You must know what IC your board has before loading up a new program from the Arduino IDE. This information is available on the top of the IC. For more details about the IC construction and functions, you can refer to the data sheet.

### ICSP pin

Mostly, ICSP (12) is an AVR, a tiny programming header for the Arduino consisting of MOSI, MISO, SCK, RESET, VCC, and GND. It is often referred to as an SPI (Serial Peripheral Interface), which could be considered as an "expansion" of the output. Actually, you are slaving the output device to the master of the SPI bus.

### Power LED indicator

This LED should light up when you plug your Arduino into a power source to indicate that your board is powered up correctly. If this light does not turn on, then there is something wrong with the connection.

### TX and RX LEDs

On your board, you will find two labels: TX (transmit) and RX (receive). They appear in two places on the Arduino UNO board. First, at the digital pins 0 and 1, to indicate the pins responsible for serial communication. Second, the TX and RX led (13). The TX led flashes with different speed while sending the serial data. The speed of flashing depends on the baud rate used by the board. RX flashes during the receiving process.

### Digital I/O

The Arduino UNO board has 14 digital I/O pins (15) (of which 6 provide PWM (Pulse Width Modulation) output. These pins can be configured to work as input digital pins to read logic values (0 or 1) or as digital output pins to drive different modules like LEDs, relays, etc. The pins labeled "~" can be used to generate PWM.

**AREF**

AREF stands for Analog Reference. It is sometimes, used to set an external reference voltage (between 0 and 5 Volts) as the upper limit for the analog input pins.

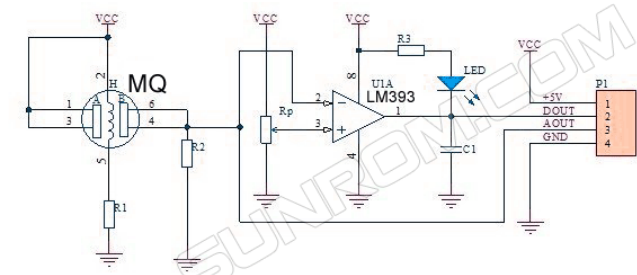
**Arduino Uno to ATmega328 Pin Mapping**

When ATmega328 chip is used in place of Arduino Uno, or vice versa, the image below shows the pin mapping between the two.

*1) The Microcontroller*

It is important to understand that the Arduino board includes a microcontroller, and this microcontroller is what executes the instructions in your program. If you know this, you won't use the common nonsense phrase "Arduino is a microcontroller" ever again.

The ATmega328 microcontroller is the MCU used in Arduino UNO R3 as a main controller. ATmega328 is an MCU from the AVR family; it is an 8-bit device, which means that its data-bus architecture and internal registers are designed to handle 8 parallel data signals. ATmega328 has three types of memory:



digital pin 4	(PCINT20/XCK/T0) PD4	6	28	PC0 (ADC0/PCINT8)	analog input 0
VCC	VCC	7	22	GND	GND
GND	GND	8	21	AREF	analog reference
crystal	(PCINT6/XTAL1/TOSC1) PB6	9	20	AVCC	VCC
crystal	(PCINT7/XTAL2/TOSC2) PB7	10	19	PB5 (SCK/PCINT5)	digital pin 13
digital pin 5 (PWM)	(PCINT21/OC0B/T1) PD5	11	18	PB4 (MISO/PCINT4)	digital pin 12
digital pin 6 (PWM)	(PCINT22/OC0A/AIN0) PD6	12	17	PB3 (MOSI/OC2A/PCINT3)	digital pin 11 (PWM)
digital pin 7	(PCINT23/AIN1) PD7	13	16	PB2 (SS/OC1B/PCINT2)	digital pin 10 (PWM)
digital pin 8	(PCINT0/CLKO/ICP1) PB0	14	15	PB1 (OC1A/PCINT1)	digital pin 9 (PWM)

Digital Pins 11, 12 & 13 are used by the ICSP header for MOSI, MISO, SCK connections (Atmega168 pins 17, 18 & 19). Avoid low-impedance loads on these pins when using the ICSP header.

- **Flash memory:** 32KB nonvolatile memory. This is used for storing application, which explains why you don't need to upload your application every time you unplug arduino from its power source.
- **SRAM memory:** 2KB volatile memory. This is used for storing variables used by the application while it's running.
- **EEPROM memory:** 1KB nonvolatile memory. This can be used to store data that must be available

even after the board is powered down and then powered up again.

**UART Peripheral:**

A UART (Universal Asynchronous Receiver/Transmitter) is a serial interface. The ATmega328 has only one UART module.

The pins (RX, TX) of the UART are connected to a USB-to-UART converter circuit and also connected to pin0 and pin1 in the digital header. You must avoid using the UART if you're already using it to send/receive data over USB.

**L293D Motor Driver:**

*B. L293D Description*

L293D is a typical Motor driver or Motor Driver IC which allows DC motor to drive on either direction. L293D is a 16-pin IC which can control a set of two DC motors simultaneously in any direction. It means that you can control two DC motor with a single L293D IC. Dual H-bridge *Motor Driver integrated circuit (IC)*.

**Alcohol Sensor**

This module is made using Alcohol Gas Sensor MQ3. It is a low cost semiconductor sensor which can detect the presence of alcohol gases at concentrations from 0.05 mg/L to 10 mg/L. The sensitive material used for this sensor is SnO<sub>2</sub>, whose conductivity is lower in clean air. It's conductivity increases as the concentration of alcohol gases increases. It has high sensitivity to alcohol and has a good resistance to disturbances due to smoke, vapor and gasoline. This module provides both digital and analog outputs. This alcohol sensor is suitable for detecting alcohol concentration on your breath, just like your common Breathalyzer. It has a high sensitivity and fast response time. Sensor provides an analog resistive output based on alcohol concentration. The drive circuit is very simple, all it needs is one resistor. A simple interface could be a 0-3.3V ADC

**IV. ALGORITHMS USED**

**Viola Jones Algorithm using Haar classifier in OpenCV**

Viola Jones Algorithm is fast, efficient and gives level of accuracy. First step of viola jones algorithm includes training of the data set using any machine learning algorithm. The training is done using the two sets of positive and negative images. Basically, this training is done in order to develop predictive relationship between the data sets. In the Viola-Jones object detection framework, the Haar-like features are therefore organized in something called a classifier cascade to form a strong learner or classifier. The key advantage of a Haar-like feature over most other features is its calculation speed. Due to the use of Third is the method of combining the classifiers in a cascade structure to increase the speed of the detector by focussing on the promising regions of interest. a Haar-like feature of any size can be calculated in

constant time (approximately 60 microprocessor instructions for a 2-rectangle feature). After the classifier function is developed, AdaBoost algorithm [3] is used for selecting the required features and training the data set. This algorithm basically is used for enhancing the performance of the classifier function. AdaBoost is an effective procedure for searching out a small number of good “features” which nevertheless have significant variety. This algorithm is used for selecting the features like eyes or mouth region. After selecting the features of interest, the data sets are trained. Summary of the steps involved in this algorithm are:

### Integral Image

The integral image at location (x, y) contains the sum of the pixels above and to the left of x and y. It is the summation of all the pixel values in an original image

### Haar-Feature

Haar features are composed of either two or three rectangles. Face candidates are scanned and searched for Haar features of the current stage. The weight and size of each feature and the features themselves are generated by the learning algorithm-AdaBoost Each Haar feature has a value that is calculated by taking the area of each rectangle, multiplying each by their respective weights, and then summing the results. The area of each rectangle is easily found using the integral image. The coordinate of the any corner of a rectangle can be used to get the sum of all the pixels above and to the left of that location using the integral image. By using each corner of a rectangle, the area can be computed quickly.

### Classifier

A Haar classifier uses the rectangle integral to calculate the value of a Haar feature. The Haar classifier multiplies the weight of each rectangle by its area and the results are added together. Several Haar classifiers compose a stage. A stage accumulator sums all the Haar classifier results in a stage and a stage comparator compares this summation with a stage threshold. The threshold is also a constant obtained from the AdaBoost algorithm. Each stage does not have a set number of Haar features. Depending on the parameters of the training data individual stages can have a varying number of Haar features

### Cascade

The cascade eliminates candidates by making stricter requirements in each stage with later stages being much more difficult for a candidate to pass. Candidates exit the cascade if they pass all stages or fail any stage. A face is detected if a candidate passes all stages.

### DESIGN

There are two basic steps in this project

### Classifier Training

In this project around 1000 positive images and 5000 negatives images were taken as sample datasets in order to train the Face, Eye Classifiers. The Output of Viola Jones Algorithm are the Classifier files –face.xml, eye.xml. This step is performed only once during the execution of the project and at the beginning. The input to the Drowsy Detector Algorithm is these xml Classifier files. Drowsiness of a person can be measured by the extended period of time for which his/her eyes are in closed state. In this system, primary attention is given to the faster detection and processing of data. The number of frames for which eyes are closed is monitored. If the number of frames exceeds a certain value, then a warning message is generated on the display showing that the driver is feeling drowsy. In this algorithm, first the image is acquired by the cam for processing. Then the Haarcascade file face.xml zface is detected then another frame is acquired. If a face is detected, then a region of interest in marked within the face. This region of interest contains the eyes and mouth. Defining a region of interest significantly reduces the computational requirements of the system. After that the eyes from the region of interest by using eye.xml respectively.

### Image Acquisition

A Webcam is fixed in a rigid location for capturing the face of the driver. The location of the webcam is strategically decided so as the Web cam doesn't interfere with the driver's vision. 3.2.2 Face detection

The face area is detected using the haardetectobjects () function [4] for which the face classifier xml file is passed as a parameter to the action. The number of detected Faces is stored in a vector. Rectangular region of interest are then marked for the detected faces.3.2.3Eyes Detection The eye area o detected using the haardetectobjects () function for which the eye.xml classifier xml files respectively are passed as a parameter to the function. The number of detected eyes and mouth are stored in a vector. Rectangular region of interest is then marked for the detected eyes.

The output of the Viola Jones Algorithm is a trained Haar Classifier that is stored in a XML File which can be loaded using the OpenCV function: cvLoad() and then haardetectobjects()[4] is used to find objects similar to the ones it was trained on to train our own classifiers to detect other objects such as faces , eyes and mouth.This has been done with the OpenCV haar training application, which creates a classifier from a training set of positive and negative samples .Steps are described below:

### Creation of Positive Dataset Directory

Positive dataset means the collection of the object which is required to be detected by the system. In this case the positive dataset will contain the images which have faces.

These may be stored in one or more directories indexed by a text file in the following format

```
<path>/img_1 count_1 x11 y11 w11 h11 x12 y12 . . .
<path>/img_2 count_2 x21
y21 w21 h21 x22 y22 . . .
```

Each of these lines contains the path (if any) and file name of the image containing the object(s). This is followed by the count of how many objects are in that image and then a list of rectangles containing the objects. Here the starting value of the rectangles are stored followed by the width and height pixel values. In order to get the maximum accuracy in detecting the face region, eye region and mouth region, taking into consideration more datasets will be necessary. The output of this process finally creates a directory file in the positive.txt In the case of the positive dataset for faces, the pixel value of rectangles bounding the face will be stored and similar case goes with the eyes and mouth.

#### Creation of Vector File

The utility application create samples is used to build a vector output file of the positive samples. This vector file is used as the input for the training function.

```
For example: create samples - vec face.vec -info
positive.txt -w 30 - h 40
```

This takes the positive.txt file and outputs a formatted training file, face.vec. Then create samples extracts the positive samples from the images before normalizing and resizing them to the specified width and height

#### Creation of Negative Dataset Directory

The Viola-Jones cascade is a binary classifier: It simply decides whether or not the object in an image is similar to the training set. Any image that doesn't contain the object of interest can be turned into a negative sample. It is useful to take the negative images from the same type of data. That is, if one wants to learn faces in online videos, for best results one should take their negative samples from comparable frames. However, comparable results can still be obtained using negative samples taken from any other source. Again, the images are put into one or more directories and then an index file is made consisting of a list of image filenames, one per line. For example, an image index file called negative.txt

#### Training of the Classifier

With the creation of all the datasets one can train the classifier. The OpenCv [4] documentation provides set of haar training commands and functions which can be used for

training the classifier. The following haar training command is used.

```
Haartraining /
-
data face /
-
vec face.vec
-
w 30
-
h 40 /
-
bg negative.txt
/
-
nstages 20 /
-
nsplits 1 /
[
-
nonsym] /
-
minhitrate 0.998 /
-
maxfalsealarm 0.5
```

The face.vec file contains the information regarding the positive dataset. The negative.txt gives the information regarding the negative dataset.

The cascade goes through a total of 20 stages in the training process.

In the case of the project the classifier was executed thrice in order to train the following three classifiers- face.xml, eye.xml, mouth.xml

## IV. RESULTS AND DISCUSSION

We monitor the number of frames in which the eyes are closed. When the number of frames is above a certain threshold, the driver is informed about his drowsy state by a warning message by an alarm. A detected blink is equivalent of a closed eye, if the closing state continues for more than 10 consecutive frames then drowsiness condition is displayed.

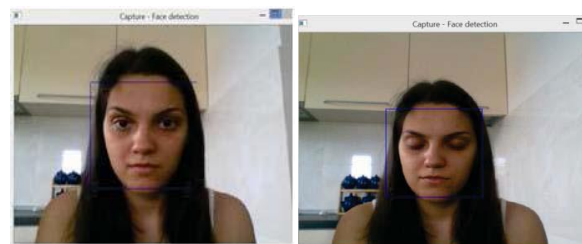
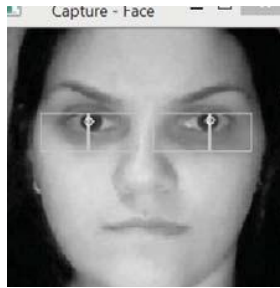
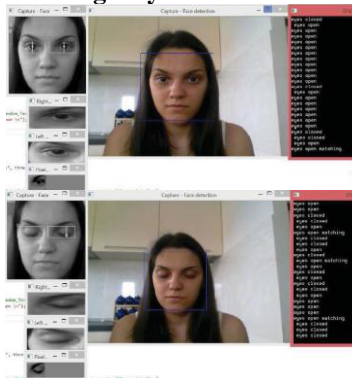


Fig2: Face detection results





**Fig3: eyes detection**



## V. CONCLUSION AND FUTURE SCOPE

**Conclusion:** This paper has discussed a system for assisting driver which is very effective for preventing major accidents caused due to driver drowsiness and alcohol consumption. Algorithms related to image processing have been used to identify the state of driver. A buzzer is used to alert the driver if he/she is drowsy. With reference to the centre of gravity the position of driver's head is determined and accordingly the current state of driver is identified. A camera of appropriate resolution is used to sense the movement of eyes. A system gives extra feature of yawning detection. If a driver yawns more frequently then also an alarm is generated. A sensor is used to detect whether the driver is drunken or not. There should be proper distance between the sensor and the driver for accuracy. The alarm generated can be in the form of audio in order to inspire the driver to reach his/her destination safely. Although there is need for more research, the proposed system can contribute effectively in detecting the driver's state and highly decrease the frequency of road accidents.

**Future work:** There are many improvisations that can be brought to this model such as implementing , NLP and so on. Therefore it can be made interactive with the user and keep monitor of every parameters of the car for the safer drive.

## ACKNOWLEDGMENT

Akash gowda BR, Dharmesh B chavda and Md ismail zabi Ulla would like to thank **my guide Mrs. Swetha N Assistant Professor, CSE Dept., East West Institute of Technology for**

providing the facilities. Finally, I would like to thank all the Teaching, Technical faculty and supporting staff members of Department of Computer Science and Engineering, East West Institute of Technology, Bengaluru, for their support

## REFERENCES

- [1]. 1.Yuichi Saito, Makoto Itoh, Toshiyuki Inagaki, "Driver Assistance System With a Dual Control Scheme: Effectiveness of Identifying Driver Drowsiness and Preventing Lane Departure Accidents" IEEE Transactions on Human-Machine Systems March 21, 2016.
- [2]. 2. D. Tran, E. Tadesse and W. Sheng, Y. Sun, M. Liu and S. Zhang, "A Driver Assistance Framework Based on Driver Drowsiness Detection" The 6<sup>th</sup> Annual IEEE International Conference on Cyber Technology in Automation, Control and Intelligent Systems June 19-22, 2016.
- [3]. 3. A. Mittal, K. Kumar, S. Dhamija, M. Kaur "Head Movement-Based Driver Drowsiness Detection: A Review of State-of-Art Techniques" 2
- [4]. Nd IEEE International Conference on Engineering and Technology (ICETECH) March 17-18, 2016.
- [5]. 4. Alesandar, Oge Marques and Borko Furht "Design and Implementation of a Driver Drowsiness Detection System A Practical Approach".
- [6]. 5. Anjali K U, Athiramol K Thampi, Athira Vijayaraman, Franiya Francis M, Jeffy James N, Bindhu K Rajan " Real-Time Nonintrusive Monitoring and Detection of Eye Blinking in View of Accident Prevention Due to Drowsiness" 2016 International Conference on Circuit , Power and Computing Technologies[ICCPCT].
- [7]. 6. J. Ahmed, Jain-Ping Li, S. Ahmed Khan, R.Ahmed Shaikh "Eye Behavior Based Drowsiness Detection System".
- [8]. 7. A. Rahman, M. Sirshar, A. Khan "Real Time Drowsiness Detection Using Eye Blink Monitoring" 2015 National Software Engineering Conference(NSEC 2015).

## Authors Profile

*Mr. AKASH GOWDA B R* is pursuing her 8 semester BE in Computer Science & Engineering at East West Institute of Technology, Bengaluru, India. His area of interest include Image processing

*Mr. DHARMESH B CHAVDA* is pursuing her 8 semester BE in Computer Science & Engineering at East West Institute of Technology, Bengaluru, India. His area of interest include Image processing

*Mr. MD ISMAIL ZABI* is pursuing her 8 semester BE in Computer Science & Engineering at East West Institute of Technology, Bengaluru, India. His area of interest include Image processing

*Mrs. Swetha N got M.Tech degree in Computer Science, Bengaluru, India. she is currently working as Assistant Professor in the Department of CSE, EWIT. Her area of interest include Image processing , machine learning.*