Driver Drowsiness Detection System

COURSE:
Human Computer Interaction

Submitted To:
Dr. Muhammad Firoz Mridha

Submitted By:
Sadir Ahmed Zidan, 20-43263-1
Sajjad Hossain Shakib, 20-43842-2
MD. Sahilur Rahman, 20-43257-1
S.M. Adnan Mahmud, 20-43507-1

Department of Computer Science
Faculty of Science & Technology
American International University, Bangladesh

Summer, 2022-2023

1. Abstract

Drowsiness is a state of strong inclination to sleep and poses a significant risk to safety, in activities such as driving or operating heavy machinery. It can significantly affect cognitive abilities such as attention, concentration, and decision-making. It can lead to decreased alertness, slower reaction times, and reduced ability to process information effectively. Drowsy driving accidents have serious repercussions including the possibility of fatal injuries and substantial economic losses, so to prevent mishaps and secure the health of individuals it's crucial to identify and alleviate sleepiness in real-time. Machine learning techniques like transfer learning and deep neural networks are facilitating new prospects for detecting drowsiness in recent times and by using Mobile Net architecture as a base model and incorporating transfer learning method we are aiming to construct an efficient and accurate drowsiness detection system. We will use eye images for dataset as two states for open and close eyes of individuals and using Mobile Net as a base model and some custom layers to adapt the features. After training and validating data and after getting a good accuracy in test data or new data we can go for real time detection using webcam and Arduino. While detecting the state of the eyes, the output will show in an LCD Display, when the predicted result will continuously show the eye is close for several frames then it will show the driver is drowsy on the display then a buzzer will ring, and a danger alert led light will blink. Results of each frame will continuously interact with the Arduino Board which shows the output and gives the alert.

2. Introduction

Traffic accidents are one of the top 10 causes of death worldwide, according to reports from the World Health Organization (WHO) [1]. A report of the National Safety Council of USA shows that each year, drowsy driving accounts for about 100,000 crashes, 71,000 injuries and 1,550 fatalities [2]. Therefore, identifying drowsy drivers could be a useful means of avoiding accidents. Additionally, it enhances the functionality of the Driver Monitoring System (DMS) and Advanced Driver Assistance Systems (ADAS), enhancing road safety. Drowsiness detectors can be divided into three primary categories: vehicle-based [3], signal-based [4], and facial feature-based [5]. Vehicle-based systems track changes in steering wheel angle, acceleration, lateral position, and other factors to infer tiredness from driving conditions. For real-time work, these methods are inefficient. Drowsiness is inferred from psychophysiological characteristics via signal-based approaches. Based on these techniques, several research have been conducted in recent years [6]. The most important physiological signals that have been employed and studied include electroencephalograms (EEG) [7], electrooculograms (EOG) [8], electrocardiograms (ECG) [9], skin temperature (ST), galvanic skin response (GSR), and intramuscular activity as measured by electromyograms (EMG). These strategies must consider invasive captors, which may have a negative impact on driving. Facial feature-based approaches are less expensive and more widely available than other approaches, and they can analyze the target in real time without using invasive devices.

3. Methodology

The main goal of this project is to develop an automatic driver drowsiness detection and warning system with better accuracy and minimal cost. The system we design will correctly track the driver's eyes' open and closed states in real time, and this will be our focus. This system can detect drowsy drivers by utilizing a sequence of images of closed eyes. First, we need to use image processing and computer vision to extract features from capture frames of real time videos. After that for predicting close and open eyes, we will use machine learning which reduces the cost of different sensors. To implement machine learning on image data, Convolutional Neural Network is used which is mainly deep learning. We will use one of the well performed and widely popular CNN model MobileNet as transfer learning. We will use python to build our model and the main program. Then a dataset was chosen to train the model. After the model is trained another CNN model Haar cascade classifier will be used to detect human facial features from images with any background. It will detect the human eye and the MobileNet model will predict the eye state. We will use serial communication library to send the result to our Arduino Program, which is written in C language, the Arduino program will check the result and gives output according to result. In the Arduino program we will mainly use different instructions for LCD display, buzzer and led light. Then the Arduino setup will be built and connected to the computer. Then the Arduino program will be exported to the Arduino board. When the python program run, the camera will start automatically with the help of OpenCV which is a computer vision library and it will keep capturing frames, the machine learning model will implement on each frame and calculate the result while the results will continuously pass to the Arduino board via serial communication, and it will show the output. After the whole system starts working properly, we can have a system that can detect driver drowsiness in real time through camera and give alert through display, led and buzzer.

3.1. Components

Hardware

- Arduino Uno
- Camera
- Wires
- Buzzer
- Breadboard
- Display
- LED

Software & Tools

- Arduino IDE
- VS Code
- Microsoft Visio
- Jupyter Notebook

3.2. Description of components

- Arduino Uno: The Arduino Uno is an open-source microcontroller board created by Arduino.cc and was first released publicly in 2010. It is based on the Microchip ATmega328P microprocessor. [10] [11] A selection of expansion boards (shields) and other circuits can be interfaced with the board's sets of digital and analog input/output (I/O) pins. The board contains six analog I/O pins and 14 digital I/O pins, six of which can be used for PWM output. It can be programmed using the Arduino IDE (Integrated Development Environment) with a type B USB connector. It accepts voltages between 7 and 20 volts, but an external 9-volt battery or USB cable can also power it. It mimics Arduino Nano and Mega in specific ways.

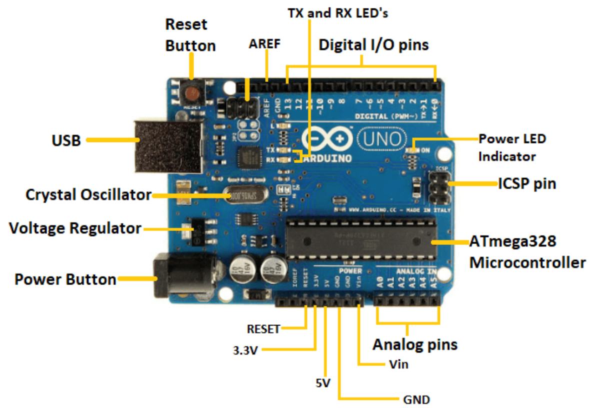

Fig: Arduino Uno

- Camera: An optical device designed to seize static visuals or document dynamic visuals is known as a camera. These visuals can be stored within a physical storage medium, either in a digital setup or on traditional photographic film. Comprising of an essential lens that concentrates light from the subject, as well as a camera body that houses the image-capturing mechanism, cameras play a crucial role in capturing moments. For instance, in the realm of electronics, an Arduino-compatible camera module can be incorporated to enable image capture within innovative projects.

Fig: Camera

- Buzzer: A beeper or buzzer, for example, could be electromechanical, piezoelectric, or mechanical in design. The signal's primary function is to convert audio to sound. It is often powered by DC voltage and used in timers, alarm clocks, printers, computers, and other electronic devices. It can produce a variety of sounds, including an alarm, music, a bell, and a siren, depending on the varied designs. [12]

Fig: Buzzer

- LED: When an electric current passes through a semiconductor device called a light-emitting diode (LED), the LED emits light. When current flows through an LED, the electrons and holes recombine and produce light. LEDs only let current flow in one direction—forward—and stop it from going the other way. [13]

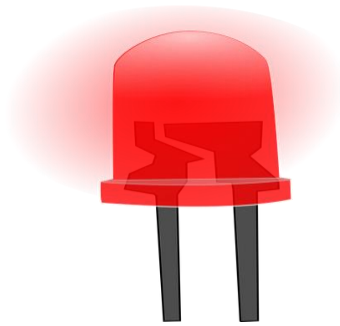

Fig: LED

- LCD Display: Liquid Crystal Display, or LCD in LCD 162, is a display technology that is used in screens for computer monitors, TVs, smartphones, tablets, and other mobile devices. Although LCD and CRT displays have similar appearances, they operate differently. A liquid crystal display contains a backlight that illuminates each pixel that is placed in a rectangular network, as opposed to electrons diffracting at a glass panel. [14]

Fig: LCD Display

- Arduino IDE: Arduino IDE will be used to write Arduino code and connect it with Arduino Uno.
- VS Code: To build the model and write coded in python, VS code will be used.
- Microsoft Visio: Circuit Diagram, Flow chart and Block Diagram is drawn using Visio.
- Jupyter Notebook: Along with VS code, it will also be used for writing code in python.

3.3. Diagrams

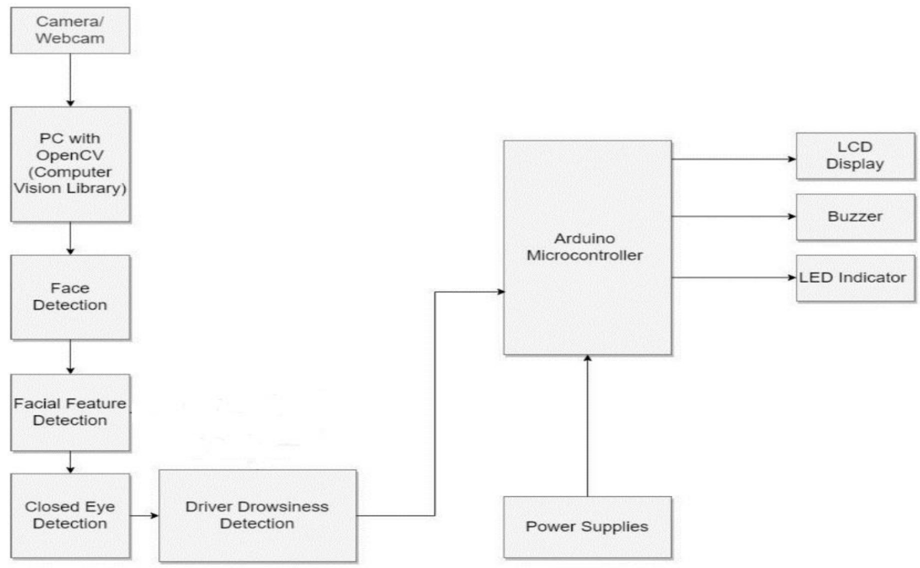

Fig: Block Diagram

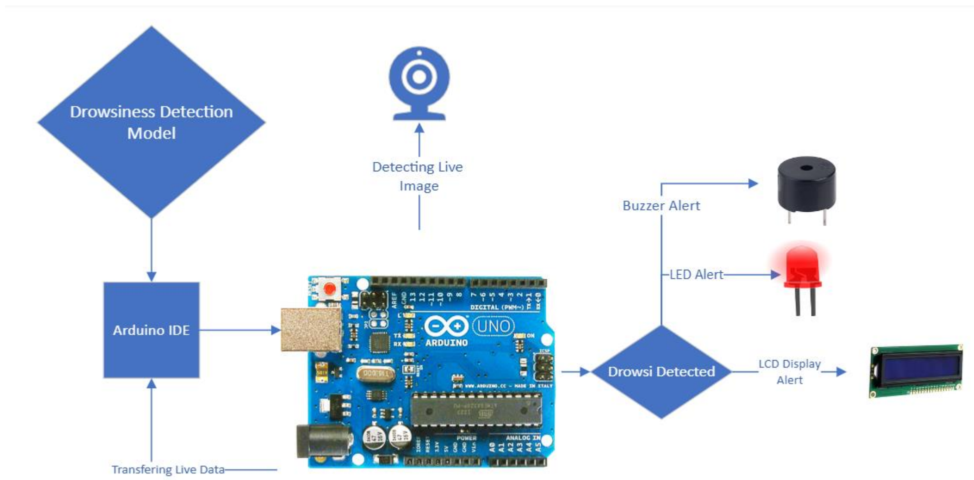

Fig: Circuit Diagram

3.4. Software Installation

The required software packages will be installed, including the Arduino IDE, Python, OpenCV, and VS code and Jupyter Notebook.

3.5. Data Collection:

In order to train our machine learning CNN model, we need to choose a dataset with enough numbers of data. In this project, MRL eye dataset will be used [15]. This dataset has an extensive collection of images of human eyes. This dataset includes low and high-resolution infrared photos that were all taken under varied lightning situations and with various equipment. Testing various features or trainable classifiers is appropriate for this dataset. The photographs are separated into numerous categories to make it easier to compare methods, and this also makes them excellent for developing and testing classifiers. This dataset is consisting images of 37 candidates, among them 33 men and 4 women. All the images are open eyes or closed eyes image. There are total number of 84,898 images of close and open eyes. We will split the data 80% for training and validation other 20% for testing and evaluating the model.

3.6. Model:

MobileNet Model will be used for classification of closed eyes and open eyes. This model is widely used for image classification as it performed well on the Image dataset. Here we can use binary classification which is only closed eye and open eyes. We can also modify the classification layers. We can use Haar Cascade frontal face and Haar Cascade eye to detect face and eyes from images or frames from camera data. First, Haar cascade frontal face detect only face portion and then the Haar cascade eye detect the eyes from the face portion. The whole process is done mainly in two steps.

In the 1st step: The datasets will be prepared then the CNN Model will be built based on the problem. After that, the CNN model will be trained on the prepared Dataset.

In the 2nd step: Realtime detection will be done using OpenCV computer vision library. Haar Cascade classifier detects the face and eyes After the eyes detected it cropped the eye image and convert the image from RGB to Gray Scale. Then the Modified MobileNet trained model detects the eye state, if eye is close the calculated value is near 0 and if it is open then the value is near 1. The calculated result will be sent to the Arduino Program by using pyserial library which is mainly used for serial communication between python program and Arduino. Based on the result the Arduino program gives different instructions to the Arduino board.

The pyserial library sent the results to the Arduino board using serial communication and the exported Arduino program in the Arduino board shows the output based on the calculated results. If the result shows close eyes, it will calculate a score where score=score+1, if it is open eyes the score=0. Once the score crosses the threshold, it will alert drowsiness. The threshold is basically the serial number of frames where the eyes are closed, the threshold can be changed based on the fps rate of the device. If the device captures frames slowly then the threshold will be less.

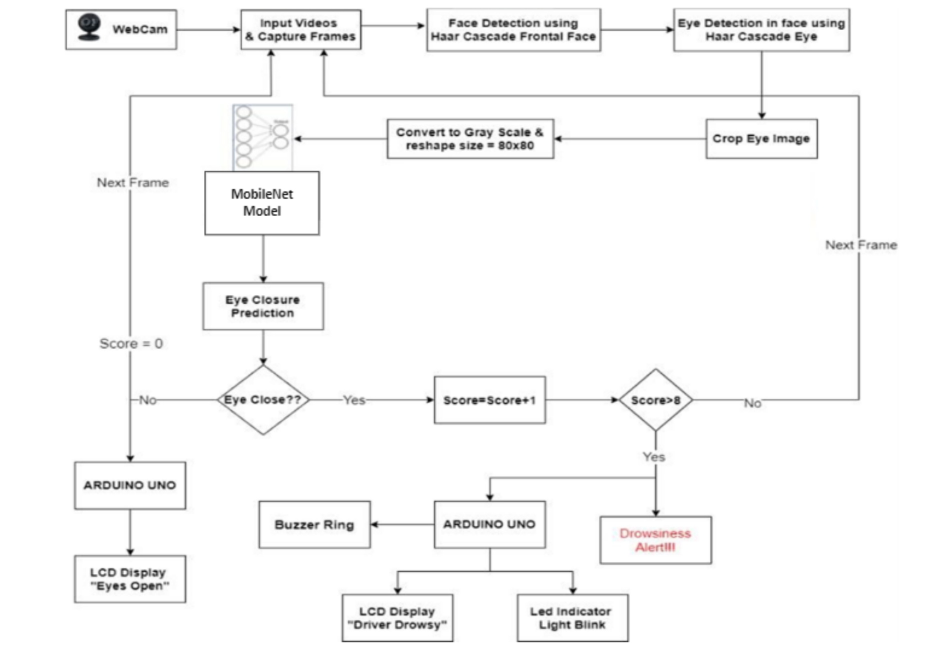

Fig: Flowchart Of the proposed method

3.7. Final Model

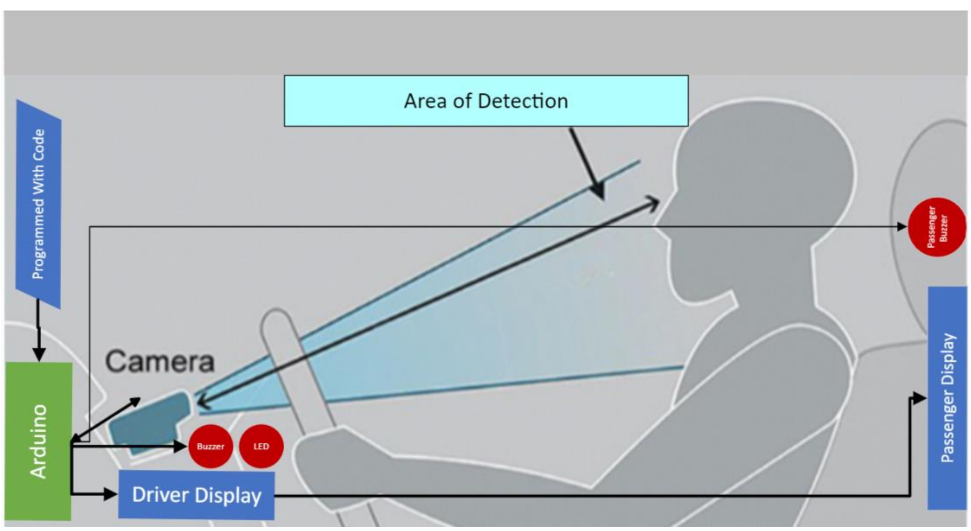

Fig: Final Model

4.Features

The core idea behind Arduino-based driver drowsiness detection system is quite straight forward. We will need an LCD display, a camera, a buzzer, led light and Arduino Uno Board which will be connected with the computer or code. And after completion, the features will be:

- Driver Drowsiness detection system using facial feature-based approach.

- Drowsiness detection in the most accurate and less expensive way.
- Getting alert on live data.
- Alert through buzzer.
- Alert through LED.
- Alert through LCD display.

And these alerts are not only for drivers, but also the passengers can monitor and know about the condition of driver through LCD display and buzzer.

5. Advantage:

Drowsiness can hinder performance in various domains, including academic, work, and sports. It can impact productivity, learning, and skill execution. Tasks that require focus, precision, and quick thinking may be compromised, leading to suboptimal outcomes. Here are some of the advantages:

- **Enhanced Road Safety**: The primary advantage of the proposed system is the enhancement of road safety. By detecting driver drowsiness in real-time, the system aims to prevent accidents caused by drowsy driving. This is particularly important given that drowsy driving contributes to a significant number of accidents and fatalities.

- **Accurate Drowsiness Detection**: The system utilizes image processing and machine learning techniques to accurately detect driver drowsiness. It employs the Haar Cascade Classifier algorithm for face and eye detection, as well as the MobileNet model for eye blink detection. These advanced techniques contribute to higher accuracy in identifying drowsiness.

- **Low-Cost Implementation**: The system is designed to be cost-effective. By utilizing readily available components such as an Arduino UNO board, a camera module, an LCD display, a buzzer, and LEDs. So, the drowsiness detection system is affordable and accessible.

- **Real-time Monitoring**: The proposed system provides real-time monitoring of the driver's eye state. It continuously captures frames from a camera, processes the images, and predicts the driver's eye status (open or closed). This instantaneous monitoring ensures that any signs of drowsiness are promptly identified.

- **Automatic Alerts**: Once the system detects that the driver's eyes have been closed for a certain duration (crossing a predefined threshold), it triggers automatic alerts. These alerts are displayed on the LCD display, and both a buzzer and an LED light will be activated to alert the driver and draw their attention back to the road.

- **Non-Invasive Approach**: Unlike some other drowsiness detection methods that require invasive devices or physiological sensors, this system relies on non-intrusive methods like image analysis. It analyzes the driver's eye behavior without requiring additional sensors attached to the driver's body.

6. Conclusion & Future Works:

In conclusion, the proposed drowsiness detection system offers a promising solution to a critical safety issue – drowsy driving. By combining image processing, computer vision, machine learning, and hardware components like the Arduino Uno board, camera, buzzer, and LED, the system can accurately and promptly detect driver drowsiness in real-time. The use of the MobileNet CNN model for eye state prediction, along with the Haar Cascade Classifier for facial feature detection, showcases the system's advanced technology. The main advantage of the system lies in its ability to enhance road safety by preventing accidents caused by drowsy driving. The real-time monitoring of driver eye states, automatic alerts through various mechanisms, and the non-invasive nature of the approach make it a valuable addition to modern vehicles. The system's cost-effectiveness and adaptability contribute to its practicality and potential for wider implementation.

Future Works:

While the proposed drowsiness detection system presents a robust foundation, there are several avenues for future work and improvements:

1. **Enhanced Model Training:** Continuous improvement of the CNN model's accuracy can be pursued by collecting more diverse and extensive datasets for training. This may involve variations in lighting conditions, facial appearances, and different ethnicities to ensure the model's generalization.

2. **Personalization:** Developing the system to recognize individual drivers' unique eye behaviors could improve accuracy. Personalized models could be trained for specific drivers, considering factors like blinking patterns and variability.

3. **Multimodal Approaches:** Combining facial feature-based detection with other physiological signals like heart rate, EEG, or steering behavior could enhance the system's accuracy and reliability.

4. **Cloud Connectivity:** Leveraging cloud computing could allow for centralized monitoring of multiple vehicles, providing real-time drowsiness alerts to fleet managers or family members.

5. **Data Privacy and Ethics:** Ensuring data privacy and addressing ethical concerns related to driver monitoring systems is crucial. Clear consent and data protection mechanisms should be in place.

6. **Sensitivity to Different Races and Genders:** Ensuring the system is equally accurate across different races, genders, and age groups requires careful consideration and diverse training data.

In conclusion, the proposed drowsiness detection system has the potential to significantly enhance road safety and prevent accidents caused by drowsy driving. Continued research, development, and testing will refine the system's capabilities and contribute to its successful integration into vehicles, ultimately saving lives and reducing the impact of drowsy driving accidents.

References:

[1] World Health Organization The top ten causes of death. [Online] https://www.who.int/news-room/fact-sheets/detail/the-top-10-causes-of-death

[2] Drivers are falling asleep behind the wheels. [Online] https://www.nsc.org/road/safety-topics/fatigueddriver#:~:text=A%20study%20by%20the%20AAA,and%2
0about%206%2C400%20were%20fatal.

[3] Yu J, Chen Z, Zhu Y, Chen YJ, Kong L, Li M. Fine-grained abnormal driving behaviors detection and identification with smartphones. IEEE Trans Mobile Comput.
2016;16(8):2198–212.

[4] Budak U, Bajaj V, Akbulut Y, Atila O, Sengur A. An effective hybrid model for EEG-based drowsiness detection. IEEE Sens J. 2019;19(17):7624–31.

[5] Ngxande M, Tapamo JR, Burke M. Driver drowsiness detection using behavioral measures and machine learning techniques :a review of state-of-art techniques. In: 2017 Pattern recognition association of South Africa and robotics and mechatronics. 2017;pp. 156-161.

[6] Sahayadhas A, Sundaraj K, Murugappan M. Detecting driver drowsiness based on sensors: a review. Sensors. 2012;12(12):16937–53.

[7] Kim JY, Jeong CH, Jung MJ, Park JH, Jung DH. Highly reliable driving workload analysis using driver electroencephalogram (EEG) activities during driving. Int J Automotive Technol. 2013;14(6):965–70.

[8] Zheng WL, Gao K, Li G, Liu W, Liu C, Liu J, Q., & Lu, B. L. . Vigilance estimation using a wearable EOG device in real driving environment. In: IEEE Transactions on Intelligent Transportation Systems. 2019;.

[9] Lee H, Lee J, Shin M. Using wearable ECG/PPG sensors for driver drowsiness detection based on distinguishable pattern of recurrence plots. Electronics. 2019;8(2):192. [10] Somasundaram, M. R., Manoj, S., Ali, H. M. A., Mohanakarthikeyan, M., & Ranjithkumar, M. (2021).

[10] Arduino Uno For Beginners - Projects, Programming, and Parts (Tutorial). (2017). Retrieved 22 December 2022. from https://www.makerspaces.com/arduino-uno-tutorialbeginners/

[11] Medea (2022). Retrieved 22 December 2022,from http://medea.mah.se/2013/04/arduino-faq/

[12] Agarwal, T. (2021). Buzzer: Working, Types, Circuit, Advantages & Disadvantages. Retrieved 22 December 2022, from https://www.elprocus.com/buzzer-working-applications/

[13] What is LED? - Definition, Working, Properties, Uses, Advantages. (2022). Retrieved 22 December 2022, from https://byjus.com/physics/light-emitting-diode

[14] Robotics Bangladesh (RoboticsBD). (2022). Retrieved 22 December 2022, from https://store.roboticsbd.com/

[15] MRL Eye Dataset. [Online] http://mrl.cs.vsb.cz/eyedataset