Drowsiness detection with OpenCV

Project Guide :Ms.Lino Murali

Anakha K Babu 12160012
Bhagya C 12160027
Jithinya K 12160043

Department of Computer Science
School of Engineering,CUSAT

B.Tech (2015-2019) Project Design
14-Nov 2018

Outline

1. Introduction
   - Why Drowsiness detection?
   - Why OpenCv?
   - Introduction to Drowsiness detection with OpenCv

Outline

1. Introduction
   - Why Drowsiness detection?
   - Why OpenCv?
   - Introduction to Drowsiness detection with OpenCv
2. Problem Statement

Outline

1. Introduction
   - Why Drowsiness detection?
   - Why OpenCv?
   - Introduction to Drowsiness detection with OpenCv
2. Problem Statement
3. Literature Review
   - Literature Review 1
   - Literature Review 2
   - LIterature Review 3

Outline

1. Introduction
   - Why Drowsiness detection?
   - Why OpenCv?
   - Introduction to Drowsiness detection with OpenCv
2. Problem Statement
3. Literature Review
   - Literature Review 1
   - Literature Review 2
   - LIterature Review 3
4. Modular Design
   - Rigging the car or table with a drowsiness detector
   - The drowsiness detector algorithm
   - Building the drowsiness detector with OpenCV
   - Testing the OpenCV drowsiness detector

Outline

1. Introduction
   - Why Drowsiness detection?
   - Why OpenCv?
   - Introduction to Drowsiness detection with OpenCv
2. Problem Statement
3. Literature Review
   - Literature Review 1
   - Literature Review 2
   - LIterature Review 3
4. Modular Design
   - Rigging the car or table with a drowsiness detector
   - The drowsiness detector algorithm
   - Building the drowsiness detector with OpenCV
   - Testing the OpenCV drowsiness detector
5. Tools Used
   - Hardware Tools
   - Software Tools

Outline

1. Introduction
   - Why Drowsiness detection?
   - Why OpenCv?
   - Introduction to Drowsiness detection with OpenCv
2. Problem Statement
3. Literature Review
   - Literature Review 1
   - Literature Review 2
   - LIterature Review 3
4. Modular Design
   - Rigging the car or table with a drowsiness detector
   - The drowsiness detector algorithm
   - Building the drowsiness detector with OpenCV
   - Testing the OpenCV drowsiness detector
5. Tools Used
   - Hardware Tools
   - Software Tools

Outline

1. Introduction
   - Why Drowsiness detection?
   - Why OpenCv?
   - Introduction to Drowsiness detection with OpenCv
2. Problem Statement
3. Literature Review
   - Literature Review 1
   - Literature Review 2
   - LIterature Review 3
4. Modular Design
   - Rigging the car or table with a drowsiness detector
   - The drowsiness detector algorithm
   - Building the drowsiness detector with OpenCV
   - Testing the OpenCV drowsiness detector
5. Tools Used
   - Hardware Tools
   - Software Tools

Outline

1. Introduction
   - Why Drowsiness detection?
   - Why OpenCv?
   - Introduction to Drowsiness detection with OpenCv
2. Problem Statement
3. Literature Review
   - Literature Review 1
   - Literature Review 2
   - LIterature Review 3
4. Modular Design
   - Rigging the car or table with a drowsiness detector
   - The drowsiness detector algorithm
   - Building the drowsiness detector with OpenCV
   - Testing the OpenCV drowsiness detector
5. Tools Used
   - Hardware Tools
   - Software Tools

Why Drowsiness detection?

- According to available statistical data, over 1.3 million people die each year on the road and 20 to 50 million people suffer non-fatal injuries due to road accidents.
- In order to prevent these devastating accidents,the state of drowsiness of the driver should be monitored.
- These statistics suggest that driver drowsiness is one of the main causes of road accidents.

Outline

1. Introduction
   - Why Drowsiness detection?
   - Why OpenCv?
   - Introduction to Drowsiness detection with OpenCv
2. Problem Statement
3. Literature Review
   - Literature Review 1
   - Literature Review 2
   - LIterature Review 3
4. Modular Design
   - Rigging the car or table with a drowsiness detector
   - The drowsiness detector algorithm
   - Building the drowsiness detector with OpenCV
   - Testing the OpenCV drowsiness detector
5. Tools Used
   - Hardware Tools
   - Software Tools

Why OpenCv?

Some Important Points

- OpenCV (Open Source Computer Vision) is a library of programming functions mainly aimed at real-time computer vision

Why OpenCv?

Some Important Points

- OpenCV (Open Source Computer Vision) is a library of programming functions mainly aimed at real-time computer vision
- In simple language it is library used for Image Processing.

Why OpenCv?

Some Important Points

- OpenCV (Open Source Computer Vision) is a library of programming functions mainly aimed at real-time computer vision
- In simple language it is library used for Image Processing.
- Detection of faces and its features.

Why OpenCv?

Some Important Points

- OpenCV (Open Source Computer Vision) is a library of programming functions mainly aimed at real-time computer vision
- In simple language it is library used for Image Processing.
- Detection of faces and its features.
- It provides hundreds of functions for the capture, analysis, and manipulation of visual data

Why OpenCv?

Some Important Points

- OpenCV (Open Source Computer Vision) is a library of programming functions mainly aimed at real-time computer vision
- In simple language it is library used for Image Processing.
- Detection of faces and its features.
- It provides hundreds of functions for the capture, analysis, and manipulation of visual data
- Primary interface of OpenCV is in C++.There are also C, Python and JAVA full interfaces

Outline

1. Introduction
   - Why Drowsiness detection?
   - Why OpenCv?
   - Introduction to Drowsiness detection with OpenCv
2. Problem Statement
3. Literature Review
   - Literature Review 1
   - Literature Review 2
   - LIterature Review 3
4. Modular Design
   - Rigging the car or table with a drowsiness detector
   - The drowsiness detector algorithm
   - Building the drowsiness detector with OpenCV
   - Testing the OpenCV drowsiness detector
5. Tools Used
   - Hardware Tools
   - Software Tools

Introduction to Drowsiness detection with OpenCv

Some Important Points

- We introduce, a computer vision system that can automatically detect driver drowsiness in a real-time video stream and then play an alarm if the driver appears to be drowsy

Introduction to Drowsiness detection with OpenCv

Some Important Points

- We introduce, a computer vision system that can automatically detect driver drowsiness in a real-time video stream and then play an alarm if the driver appears to be drowsy
- Compute the eye aspect ratio to determine if the eyes are closed.

Introduction to Drowsiness detection with OpenCv

Some Important Points

- We introduce, a computer vision system that can automatically detect driver drowsiness in a real-time video stream and then play an alarm if the driver appears to be drowsy
- Compute the eye aspect ratio to determine if the eyes are closed.
- well setup a camera that monitors a stream for faces and apply facial landmark localization to extract the eye regions from the face

Introduction to Drowsiness detection with OpenCv

Some Important Points

- We introduce, a computer vision system that can automatically detect driver drowsiness in a real-time video stream and then play an alarm if the driver appears to be drowsy
- Compute the eye aspect ratio to determine if the eyes are closed.
- well setup a camera that monitors a stream for faces and apply facial landmark localization to extract the eye regions from the face
- Sound an alarm if the eyes have been closed for a sufficiently long enough time.

Problem Statement

In the 21st century, driver drowsiness has continued to be a major challenge contributing to a large number of accidents on our roads.

In India, driver drowsiness especially among long distance truck drivers, public service vehicles drivers and private vehicle drivers is a major concern. This continues despite the government putting in place several measures to address the problem;

A few systems are available in the market however; they are expensive making them a reserve for a few who can afford the cost of the current vehicles fitted with search technologies.

There is hence great need to provide drowsiness detection system that are affordable to the many who are low income earners and also public service vehicles to help address the many accidents associated with drowsiness.

Outline

1. Introduction
   - Why Drowsiness detection?
   - Why OpenCv?
   - Introduction to Drowsiness detection with OpenCv
2. Problem Statement
3. Literature Review
   - Literature Review 1
   - Literature Review 2
   - LIterature Review 3
4. Modular Design
   - Rigging the car or table with a drowsiness detector
   - The drowsiness detector algorithm
   - Building the drowsiness detector with OpenCV
   - Testing the OpenCV drowsiness detector
5. Tools Used
   - Hardware Tools
   - Software Tools

Literature Review 1

Tereza Soukupov and Jan ech,Introduced A real-time algorithm to detect eye blinks in a video sequence from a standard camera is proposed

The proposed algorithm therefore estimates the landmark positions, extracts a single scalar quantity eye aspect ratio (EAR) characterizing the eye opening in each frame

Finally, an SVM classifier detects eye blinks as a pattern of EAR values in a short temporal window. The simple algorithm outperforms the state-of-the-art results on two standard datasets.

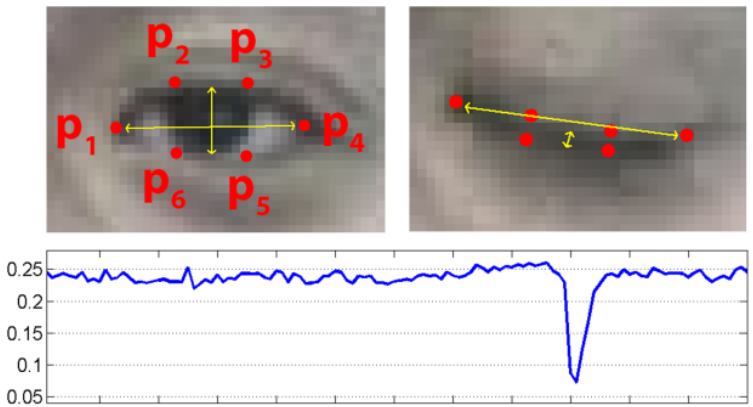

Figure 1: Open and closed eyes with landmarks p_i automatically detected by [1]. The eye aspect ratio EAR in Eq. (1) plotted for several frames of a video

Outline

1. Introduction
   - Why Drowsiness detection?
   - Why OpenCv?
   - Introduction to Drowsiness detection with OpenCv
2. Problem Statement
3. Literature Review
   - Literature Review 1
   - Literature Review 2
   - LIterature Review 3
4. Modular Design
   - Rigging the car or table with a drowsiness detector
   - The drowsiness detector algorithm
   - Building the drowsiness detector with OpenCV
   - Testing the OpenCV drowsiness detector
5. Tools Used
   - Hardware Tools
   - Software Tools

Literature Review 2

- **Vahid Kazemi and Josephine Sullivan**proposed One Millisecond Face Alignment with an Ensemble of Regression Trees.
- how an ensemble of regression trees can be used to estimate the faces landmark positions directly from a sparse subset of pixel intensities, achieving super-realtime performance with high quality predictions.
- a general framework based on gradient boosting for learning an ensemble of regression trees that optimizes the sum of square error loss and naturally handles missing or partially labelled data.

Outline

1. Introduction
   - Why Drowsiness detection?
   - Why OpenCv?
   - Introduction to Drowsiness detection with OpenCv
2. Problem Statement
3. Literature Review
   - Literature Review 1
   - Literature Review 2
   - LIterature Review 3
4. Modular Design
   - Rigging the car or table with a drowsiness detector
   - The drowsiness detector algorithm
   - Building the drowsiness detector with OpenCV
   - Testing the OpenCV drowsiness detector
5. Tools Used
   - Hardware Tools
   - Software Tools

Literature Review 3

- **Tom D RUTAROVSK** proposed Eye-blink Detection Using Gradient Orientations.This paper presents present a feature descriptor.
- Gradient orientations and magnitudes are used to build the feature descriptor.
- It use the difference between samples of open and closed eyes.
- Gradient descriptors are further used to train the SVM.
- SVM can predict the open or closed eye state from the given input data.
- Due to the knowledge of the state of the eye (open or closed), eye-blink frequency and duration can be computed.
- These parameters are used to establish the level of sleepiness

Outline

1. Introduction
   - Why Drowsiness detection?
   - Why OpenCv?
   - Introduction to Drowsiness detection with OpenCv
2. Problem Statement
3. Literature Review
   - Literature Review 1
   - Literature Review 2
   - LIterature Review 3
4. Modular Design
   - Rigging the car or table with a drowsiness detector
   - The drowsiness detector algorithm
   - Building the drowsiness detector with OpenCV
   - Testing the OpenCV drowsiness detector
5. Tools Used
   - Hardware Tools
   - Software Tools

Rigging the car or table with a drowsiness detector

- The camera used for this project was a Logitech C920 having features like:
  i) Is relatively affordable.
  ii)Can shoot in full 1080p.
  iii)Is plug-and-play compatible with nearly every device tried it with.
  Here we are planning to use Raspberry Pi 3 due to
  the real-world implications of building a driver drowsiness detector using very affordable hardware,the Raspberry Pi isnt quite fast enough for real-time facial landmark detection, for the time being, well simply use a standard laptop computer. With all hardware setup, ready to move on to building the actual drowsiness detector using computer vision techniques.

Outline

1. Introduction
   - Why Drowsiness detection?
   - Why OpenCv?
   - Introduction to Drowsiness detection with OpenCv
2. Problem Statement
3. Literature Review
   - Literature Review 1
   - Literature Review 2
   - LIterature Review 3
4. Modular Design
   - Rigging the car or table with a drowsiness detector
   - The drowsiness detector algorithm
   - Building the drowsiness detector with OpenCV
   - Testing the OpenCV drowsiness detector
5. Tools Used
   - Hardware Tools
   - Software Tools

The drowsiness detector algorithm

- (i)First, setup a camera that monitors a stream for faces.
  (ii)If a face is found, apply facial landmark detection and extract the eye regions
  (iii) Then compute the eye aspect ratio to determine if the eyes are closed:
  (iv)f the eye aspect ratio indicates that the eyes have been closed for a sufficiently long enough amount of time, Then sound an alarm to wake up the driver

Outline

1. Introduction
   - Why Drowsiness detection?
   - Why OpenCv?
   - Introduction to Drowsiness detection with OpenCv
2. Problem Statement
3. Literature Review
   - Literature Review 1
   - Literature Review 2
   - LIterature Review 3
4. Modular Design
   - Rigging the car or table with a drowsiness detector
   - The drowsiness detector algorithm
   - Building the drowsiness detector with OpenCV
   - Testing the OpenCV drowsiness detector
5. Tools Used
   - Hardware Tools
   - Software Tools

Building the drowsiness detector with OpenCV

- (i)SciPy package so we can compute the Euclidean distance between facial landmarks points in the eye aspect ratio calculation
  (ii)If the eye is closed, the eye aspect ratio will again remain approximately constant, but will be much smaller than the ratio when the eye is open.
  (iii)The dlib library ships with a Histogram of Oriented Gradients-based face detector along with a facial landmark predictor.
  (iv)Therefore, to extract the eye regions from a set of facial landmarks, we simply need to know the correct array slice indexes.

Outline

1. Introduction
   - Why Drowsiness detection?
   - Why OpenCv?
   - Introduction to Drowsiness detection with OpenCv
2. Problem Statement
3. Literature Review
   - Literature Review 1
   - Literature Review 2
   - LIterature Review 3
4. Modular Design
   - Rigging the car or table with a drowsiness detector
   - The drowsiness detector algorithm
   - Building the drowsiness detector with OpenCV
   - Testing the OpenCV drowsiness detector
5. Tools Used
   - Hardware Tools
   - Software Tools

Testing the OpenCV drowsiness detector

- (i)once the video stream was up and running, carefully start testing the drowsiness detector in the parking garage or open ground to ensure it was indeed working properly.
  (ii)hen moved on to some back roads and parking lots were there was very little traffic
  (iii)driving with your eyes closed, even for a second, is dangerous, so I took extra special precautions to ensure that the only person who could be harmed during the experiment was myself.

Outline

1. Introduction
   - Why Drowsiness detection?
   - Why OpenCv?
   - Introduction to Drowsiness detection with OpenCv
2. Problem Statement
3. Literature Review
   - Literature Review 1
   - Literature Review 2
   - LIterature Review 3
4. Modular Design
   - Rigging the car or table with a drowsiness detector
   - The drowsiness detector algorithm
   - Building the drowsiness detector with OpenCV
   - Testing the OpenCV drowsiness detector
5. Tools Used
   - Hardware Tools
   - Software Tools

Hardware Tools

- (i)Camera-Logitech C920
  (ii)Laptop-
  (iii)Raspberry Pi-

Outline

1. Introduction
   - Why Drowsiness detection?
   - Why OpenCv?
   - Introduction to Drowsiness detection with OpenCv
2. Problem Statement
3. Literature Review
   - Literature Review 1
   - Literature Review 2
   - LIterature Review 3
4. Modular Design
   - Rigging the car or table with a drowsiness detector
   - The drowsiness detector algorithm
   - Building the drowsiness detector with OpenCV
   - Testing the OpenCV drowsiness detector
5. Tools Used
   - Hardware Tools
   - Software Tools

software Tools

- (i)Python3 Interpreter.
  (ii)OpenCV and Dlib libraries.

Schedule Design

JAN 1,2 WEEK	Rigging the car or table with a drowsiness detector
JAN 3,4 WEEK	Selecting drowsiness detector algorithm and compare
FEB 1,2,3 WEEK	Implementing algorithm in laptop
FEB 4 ,MAR 1 WEEK	Implementing algorithm in Raspberry Pi
MAR 2,3 WEEK	Testing and Optimization

Conclusion

- Building the drowsiness detector using OpenCV, dlib, and Python. Our drowsiness detector hinged on two important computer vision techniques:
  (i) Facial landmark detection
  (ii) Eye aspect ratio
- The proposed system in this analysis provides accurate detection of driver fatigue.
- The proposed system is used to avoid various road accidents caused by drowsy driving ,the system is able to decide if the eyes are opened or closed.
- Image processing achieves highly accurate and reliable detection of drowsiness.
- The calculation speed, accuracy and robustness will be influenced by using combined algorithm.

For Further Reading I

- A. Asthana, S. Zafeoriou, S. Cheng, and M. Pantic. *Incremental face alignment in the wild. In Confer- ence on Computer Vision and Pattern Recognition, 2014*.
- L. M. Bergasa, J. Nuevo, M. A. Sotelo, and M. Vazquez. *Real-time system for monitoring driver vigilance. In IEEE Intelligent Vehicles Symposium, 2004.*.
- J. Cech, V. Franc, and J. Matas. *A 3D approach to facial landmarks: Detection, refinement, and track- ing. In Proc. International Conference on Pattern Recognition, 2014.*.
- M. Chau and M. Betke. *Real time eye tracking and blink detection with USB cameras. Technical Report 2005-12, Boston University Computer Science, May 2005.*.

For Further Reading II

 H. Dinh, E. Jovanov, and R. Adhami *Eye blink detection using intensity vertical projection. In In- ternational Multi-Conference on Engineering and Technological Innovation, IMETI 2012..*