**GE 401 – INNOVATIVE PRODUCT DESIGN AND DEVELOPMENT I**

**Software Progress Report II**

**Version 1**

7.12.2012

**TEAM 1**



Ömer Faruk Karaayvaz 20701059 EE

Batuhan Kılıç 20802213 EE

Neris Şan Bage 20802583 IE

Özlem Karakaş 20800668 IE

Berk Eşgüner 20800297 EE

**Table of Contents**

Abstract..........................................................................2

Introduction..................................................................2

Software Design Decisions.........................................3

Algorithm Simulations and Related Algorithms.....4

Software Development Summary.............................6

Technology Demonstration Test Plan.....................6

Conclusion....................................................................6

References.....................................................................7

.

**List of Figures and Tables**

Figure 1 :Eye blink time detection algorithm.......................4

Figure 2 :Information fusing algorithm................................5

**I.Abstract**

The document “Software Progress Report II” will try to give information on what is achieved so far, and how the rest of the development of IDS software will be handled.

**II. Introduction**

Our product, Intelligent Drowsiness Sensor, is a precaution system that will detect if a driver is sleepy and alert the driver if necessary. The detection will be made trough EEG signals acquired from the driver’s brain and measuring the eye blinking time of the driver. The IDS software is responsible of two main tasks: Detecting the eye blinking time, and combining the information coming from the two sensors specified above to alert the driver if needed.

III. Software Design Decisions

The application will be running on Android devices. Android operating system is chosen because its mainstream usage. Android operating system dominates the smartphone market with its 68% worldwide share.

Next step was choosing the lowest version of Android our application will be running on. Android version 2.3.3 (API 10) is chosen, because of the availability of functions, such as face recognition in its API. Another reason behind the version chosen is to eliminate the probability of older phones with lower processing power not being able to run our app as required. Note that the Android version restriction will not affect our target market much, almost all Android phones being used are either capable or running API 10, or already running API 10 or higher.

The resolution for the images to be captured and processed with the Android device was another decision to be made. After getting familiar with Android programming, the resolution is chosen as 640x480. This resolution is both good enough to detect eye (and subsequently the blinking time) and low enough not to restrict the variety of the phones our app will be running on. The selection of the low image quality will provide our app a lower response time and will consume less of the phone’s processing power.

The application will not have a user interface other than a simple welcome screen. The alarm levels and thresholds could be made customizable, but the risk of overcomplicating the application for the users is not taken at this stage. The application and the UI might be improved at later stages.

IV. Algorithm Simulations and Related Algorithms

***I. Algorithm Simulations***

No algorithm simulations were possible at this point, since our product requires a working application first. After we manage to get the other parts working, simulations/experiments will be carried out to optimize the performance of our product.

*II. Related Algorithms*

The software of our product requires two main algorithms: The eye-blink time detection algorithm and the information-fusing algorithm.

The eye blink time detection algorithm will check the drivers eye constantly, and decide that the driver is sleepy if the blinking time exceeds 500ms.



Figure 1. Eye blink time detection algorithm

We are implementing a product consisting of two sensors, therefore the evaluation of the data from coming from both sensors and a decision making process is necessary. The information-fusing algorithm will gather and evaluate the data coming from the two sensors and decide if an alarm is necessary.



Figure 2. Information fusing algorithm

V. Software Development Summary

So far, regarding the software part of our product, we have managed to develop a simple android application that displays images coming from the camera. We have decided on certain specifications, and algorithms as well as conducting research on means to measure the eye-blinking time, eye detection and face detection.

VI. Technology Demonstration Test Plan

At the end of the semester we are planning to present an Android application that can detect face and therefore the eyes. The application will alert the user if no face is present, so the user can reallocate the Android device to a position, at which the camera of the phone will be able to see the users face.

Getting face detection to work on an Android phone running 2.3.3 will be enough to validate the software decisions made. The tests will done on a Samsung Galaxy SIII running Android version 4.1.1 (high-end device) and a HTC Explorer running Android version 2.3.3 (lower bound).

VII. Conclusion

Through the documentation we have prepared, we managed to define the working environment of our product, as well as the specifications to satisfy the customer expectations. We decided on some crucial algorithms needed for the operation of our software.

The main risk remaining in the implementation of our project is the possibility of long image processing times. To handle this problem, image resolution can be lowered as well as t improving the algorithm that processes the images.

The main difficulty in the implementation of our project was the lack of knowledge we had about Android programming. Now that we managed to overcome the main part related with the Android (getting the images) the rest of the project is relatively straightforward to implement.

VIII. References

- http://usatoday30.usatoday.com/tech/news/story/2012-08-08/android-worldwide-smartphone-market/56876294/1

- http://developer.android.com/guide/components/index.html

- http://developer.android.com/reference/packages.html